МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ВІННИЦЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ

ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ ТА КОМП'ЮТЕРНА ІНЖЕНЕРІЯ

Науково-технічний журнал

Том 23, № 1

2026

ВІННИЦЯ
2026

ISSN 1999-9941
e-ISSN 2078-6387

Засновник:
Вінницький національний технічний університет

Рік заснування:
2004

Рекомендовано до друку та поширення
через мережу Інтернет Вченою Радою
Вінницького національного технічного університету
(протокол № 11 від 26 березня 2026 р.)

Державна реєстрація: Ідентифікатор медіа R30-01507.
Рішення Національної Ради України з питань телебачення і радіомовлення
№ 1234, протокол № 25 (31.10.2023 р.).

Журнал входить до переліку наукових фахових видань України
Категорія: «Б». Науки: технічні. Спеціальності: F2 (121) – Інженерія програмного забезпечення; F3 (122) – Комп'ютерні науки; F7 (123) – Комп'ютерна інженерія; F4 (124) – Системний аналіз та наука про дані; F5 (125) – Кібербезпека та захист інформації; F6 (126) – Інформаційні системи та технології; G6 (152) – Метрологія та інформаційно-вимірювальна техніка; G22 (163) – Біомедична інженерія
(наказ МОН № 409 від 17.03.2020 року).

Журнал представлено у міжнародних наукометричних базах даних, репозитаріях та пошукових системах:
НБУ ім. В. І. Вернадського, Polska Bibliografia Naukowa,
OUCI (Open Ukrainian Citation Index), DOAJ, J-Gate, Ulrichsweb Global Serials Directory, Litmaps

Адреса редакції:
Вінницький національний технічний університет
21021, вул. Хмельницьке шосе, 95, м. Вінниця, Україна
Тел: +38 (0432) 560848
Факс: +38 (0432) 465772
E-mail: info@itce.vn.ua
https://itce.vn.ua/uk

MINISTRY OF EDUCATION AND SCIENCE OF UKRAINE
VINNYTSIA NATIONAL TECHNICAL UNIVERSITY

INFORMATION TECHNOLOGIES AND COMPUTER ENGINEERING

Scientific and Technical Journal

Vol. 23, No. 1
2026

VINNYTSIA
2026

ISSN 1999-9941
e-ISSN 2078-6387

Founder:
Vinnytsia National Technical University

Year of foundation:
2004

Recommended for printing and distribution
via the Internet by Vinnytsia National Technical University
(Minutes No. 11 of March 26, 2026)

State Registration:
Media identifier R30-01507
Decision of the National Council of Television
and Radio Broadcasting of Ukraine
No. 1234, Minutes No. 25, dated 31.10.2023.

The journal is included in the List of Scientific Professional Publications of Ukraine
Category "B". Spetialities: 0588 – Inter-disciplinary programmes and qualifications involving natural sciences, mathematics and statistics; 0612 - Database and network design and administration; 0613 - Software and applications development and analysis; 0688 – Inter-disciplinary programmes and qualifications involving Information and Communication Technologies; 0714 - Electronics and automation; 0788 – Inter-disciplinary programmes and qualifications involving engineering, manufacturing and construction
(Order of the Ministry of Education and Science No. 409 of 17.03.2020).

The journal is presented international scientometric databases, repositories and scientific systems:
Vernadsky National Library of Ukraine, Polska Bibliografia Naukowa,
OUCI (Open Ukrainian Citation Index), DOAJ, J-Gate, Ulrichsweb Global Serials Directory,
Litmaps

Editor's office address:
Vinnytsia National Technical University
21021, 95 Khmelnytske Shose Str., Vinnytsia, Ukraine
Telephone: +38 (0432) 560848
Fax: +38 (0432) 465772
E-mail: info@itce.vn.ua
https://itce.vn.ua/en

Редакційна колегія

Editorial Board

ЗМІСТ

CONTENTS

Journal homepage: https://itce.vn.ua/en

Information Technologies and Computer Engineering **Vol. 23, No. 1. 2026**

Article's History: Received: 28.10.2025 Revised: 30.01.2026
Accepted: 26.03.2026 Published: 20.04.2026

ISSN 1999-9941; e-ISSN 2078-6387
UDC 004.056
DOI: 10.31649/vitce/1.2026.09

ChaCha: Development and modification of Salsa20 in modern cryptographic systems

Oleksii Palii*

Postgraduate Student
Vinnytsia National Technical University
21021, 95 Khmelnytske shose Str., Vinnytsia, Ukraine
https://orcid.org/0009-0006-8387-3609

Oleksandr Dudnyk

PhD in Technical Sciences, AssociateProfessor
Vinnytsia National Technical University
21021, 95 Khmelnytske shose Str., Vinnytsia, Ukraine
https://orcid.org/0009-0005-3684-965X

Abstract. The study reviewed the ChaCha20 stream cypher as the successor to the Salsa20 algorithm, emphasising development, technical features, and application in modern cryptosystems. The research relevance is determined by the widespread implementation of ChaCha20 in security protocols (TLS 1.3, VPN, etc.) due to its high performance in software implementations and resistance to cryptanalysis. The study aimed to analyse the evolution of ChaCha from Salsa20, compare it with other cyphers, and summarise the latest achievements in terms of modifications and performance. The study used methods of analysing literary sources and experimental data on the speed and resistance of cyphers. The main results included a highlighted history of ChaCha's creation based on Salsa20 and improving diffusion per round, a detailed description of the algorithm's structure (4×4 state matrix, addition-rotation-XOR operations) and its cryptographic strength (no practical attacks on the full 20-round version). The advantages of ChaCha20 over the Advanced Encryption Standard (AES) in a software environment are demonstrated; in particular, on platforms without AES hardware acceleration, ChaCha20 runs up to 3 times faster with an equivalent level of security. The implementation of ChaCha20-Poly1305 in TLS and WireGuard is considered, as well as the use of XChaCha for extended nonces and the Adiantum algorithm for disk encryption on mobile devices. Modern modifications of ChaCha (e.g., increasing the number of rounds) and their impact on performance and security were analysed. The practical value of the review is determined by a summary of modern experience with ChaCha20, which can be used for the selection of cryptographic algorithms in resource-constrained systems and for further research in the field of stream cyphers

Keywords: stream cipher; Advanced Encryption Standard; TLS 1.3; cryptanalysis; resource-constrained systems

Introduction

Stream cyphers remain a substantial class of symmetric encryption algorithms, especially in situations where high performance at the software level is required. Early implementations, such as RC4, have been officially banned in modern protocols and are considered unreliable in practice due to numerous statistical biases and side-channel attacks (Datadog Security, n.d.). Block cyphers (e.g., AES Advanced Encryption Standard), on the other hand, while reliable, rely heavily on specialised instructions for hardware acceleration. This creates difficulties when using them on devices without such support, such as mobile gadgets, IoT devices, or in resource-constrained environments. In software implementations, block cyphers often demonstrate slower performance and vulnerability to timing attacks due to the use of S-Box tables. The problem of AES vulnerability to side-channel attacks was analysed in

Suggested Citation:
Palii, O., & Dudnyk, O. (2026). ChaCha: Development and modification of Salsa20 in modern cryptographic systems. *Information Technologies and Computer Engineering*, 23(1), 9-21. doi: 10.31649/vitce/1.2026.09

*Corresponding author

detail by B. Gülmezoglu *et al.* (2019), who describe successful attacks using CPU cache observation. In the context of these challenges, research has emphasised new stream cyphers, in particular ChaCha20, a modification of Salsa20 that combines high cryptographic strength with efficiency. ChaCha20 has been successfully implemented in TLS 1.3 (Transport Layer Security), VPN protocols (Virtual Private Network, in particular WireGuard) and security systems as an effective alternative to AES-GCM (Advanced Encryption Standard in Galois/Counter Mode) in conjunction with Poly1305. Its structure avoids the typical problems of previous cyphers and ensures high performance even without hardware acceleration, which makes further study of this algorithm relevant.

The ChaCha20 algorithm continues to attract interest from researchers and developers of cryptographic systems. B. Rashidi (2024) conducted an experimental comparison of the effectiveness of ChaCha20-Poly1305 and AES-GCM in the context of the TLS 1.3 protocol. The study demonstrated that ChaCha20 provides stable performance on devices without AES hardware support and reduces power consumption, which is critical for mobile platforms. R. Serrano *et al.* (2022) emphasised the development of the ChaCha20-Poly1305 cryptographic core for TLS protocols, emphasising protection against side-channel attacks, such as energy consumption analysis and cache attacks. V.R. Kebande (2023) proposed the use of ChaCha20 in the new WireGuard VPN protocol, arguing that it offers high performance and ease of implementation even in resource-constrained environments. Proposed variant of the algorithm with an extended number of rounds (EChaCha20) increased resistance to statistical attacks without a noticeable decrease in performance. In addition, J.P. Degabriele *et al.* (2021) investigated the impact of nonce parameters on the robustness of ChaCha20 implementations, particularly when the key is reused in IoT devices.

Z. Najm *et al*. (2018) compared the power consumption and side-channel attack vulnerability between ChaCha20-Poly1305 and AES-GCM implementations on microcontrollers. The results demonstrated that ChaCha20-Poly1305 consumes approximately 7 μW per 50 bytes, which is significantly less than the 27 μW consumed by AES-GCM. In addition, ChaCha20-Poly1305 proved to be more resistant to attacks targeting time variations and power consumption, in particular due to more effective countermeasures against side channels. M. Polubelova *et al.* (2020) presented a compact, time-attack-resistant implementation of ChaCha20-Poly1305 on ARM Cortex-M4, targeting IoT devices without AES hardware support, with practical measurements of delays and throughput, reducing the risk of errors associated with manual optimisation.

Thus, the scientific community is actively researching the ChaCha20 algorithm in terms of both theoretical stability and practical implementation, confirming the relevance of further study of its modifications and areas of application. Despite considerable attention to individual aspects of ChaCha20, there is a lack of comprehensive studies that comprehensively cover the stages of its development, types of modifications, and practical implementation experience. The purpose of this article was to conduct a systematic review of the ChaCha20 algorithm, analyse its main modifications, and evaluate its scope of application in modern cryptographic systems. To achieve this goal, the following tasks were set: to analyse the origin of the ChaCha algorithm and its differences from its predecessor, Salsa20; to conduct a comparative analysis of ChaCha20 with other symmetric ciphers (in particular, AES) in terms of performance and cryptographic strength; to summarise the current experience of using and modifying ChaCha20 (TLS, VPN, XChaCha20, AEAD (Authenticated Encryption with Associated Data) mode).

Materials and Methods

The study was based on a comparative analysis of the ChaCha20 stream cypher and AES block cypher algorithms in CTR (Counter Mode) and GCM modes. The methodological basis covered several complementary approaches: analysis and synthesis of scientific and technical literature, systematisation and comparison of empirical data, qualitative and quantitative evaluation of various implementations, interpretation of experimental reports, and adaptation of known methods to the specifics of microcontroller platforms. The main approach was comparative analysis, which makes it possible to identify the strengths and weaknesses of the algorithms under study in terms of their performance, cryptographic strength, and suitability for practical application in various operating scenarios.

The research was based on peer-reviewed scientific articles published between 2008 and 2024, international standards, in particular RFC 7539 (Nir & Langley, 2015), and technical reports from companies such as Google and Cloudflare (Krasnov, 2016). The sources were selected from the Scopus and Google Scholar scientific databases using the Google search engine according to clearly defined criteria: relevance to the topic, availability of quantitative and qualitative test results, openness of experimental methods, relevance to the present technical context, and the authority of the sources from the point of view of the scientific community and industrial standards.

Analysis of testing results for ChaCha20 and AES algorithm performance on different hardware platforms was emphasised. Intel Haswell processors with AES-NI instructions (Intel, USA), ARM Cortex-A53 mobile processors installed in Google Pixel smartphones (ARM Holdings, UK), as well as popular STM32 and ESP32 microcontroller platforms (Espressif, China) were studied. The set of comparable metrics included the following indicators: small data block processing latency, processor cycle speed per byte, throughput, power consumption, side-channel attack resistance, and parallel data processing capability for performance improvement.

The analysis was performed solely based on secondary data obtained from official reports, benchmarks, and independent studies. It is based on the results of comprehensive

tests that demonstrated the superiority of ChaCha20-Poly1305 over AES-GCM in environments without specialised hardware support (De Santis *et al.*, 2017). Adapted methodologies from publications on WireGuard and Noise Framework were also used to evaluate the effectiveness of implementations on microcontrollers (Donenfeld, 2017). No laboratory experiments were conducted; conclusions were formed solely based on a thorough analysis and generalisation of available empirical results.

The sequence of the study was determined following the tasks set. The first stage was an analysis of the history of the ChaCha20 algorithm and its fundamental differences from its predecessor, Salsa20. Next, a comparative analysis of ChaCha20 with other common symmetric cyphers, in particular AES, was conducted in terms of performance and cryptographic strength. The next step was to summarise the modern experience of the use of ChaCha20 in security protocols such as TLS and VPN. The final stage considered the latest modifications to the algorithm, including XChaCha20, AEAD mode and an increase in the number of rounds. This evaluated the effectiveness of the algorithm in resource-constrained environments and investigated energy consumption and throughput in real-world applications. All these stages of the research were based on comparative analysis, which identified the strengths and weaknesses of different cryptographic algorithms and concluded on their practical application in modern cryptosystems.

Results and Discussion

History of the development of the ChaCha algorithm

The ChaCha algorithm is derived from the Salsa20 stream cypher, developed by D.J. Bernstein in 2005 for the eSTREAM competition (Bernstein, 2008). Salsa20 was one of the finalists in eSTREAM and proved itself to be a fast and secure cypher. In a typical Salsa20/20 implementation (20 rounds), it runs faster than AES and was considered by the cryptographic community to be quite reliable. The researcher also proposed reduced versions of Salsa20/12 and Salsa20/8 (12 and 8 rounds, respectively) for scenarios where speed is prioritised over maximum resistance. Salsa20/12 was included in the final eSTREAM portfolio in 2008 as a promising stream cypher for widespread use in software implementations.

In 2008, D.J. Bernstein published a new version of Salsa20 called ChaCha. The goal of ChaCha was to increase diffusion (bit mixing) during each round without losing performance. The main change concerned the round function: ChaCha retained the basic structure of Salsa20 (16 state words, mod 2^{32} addition, XOR, and cyclic shift operations), but reorganised the sequence of operations in the quarter-round, which is the basic state transformation step. In contrast to Salsa20, where each 32-bit word is updated once per quarter-round, in ChaCha each word is updated twice, which significantly improves the distribution of changes across bits. In other words, ChaCha can use each input word to affect all output words in a single round, whereas in Salsa20 the effect was more limited. This doubling of diffusion is a key difference that, as analysis has shown, can be used in ChaCha to mix bits faster: a single 1-bit change at the input of a ChaCha quarter-round affects an average of 12.5 output bits (in the absence of carry), while in Salsa20 it affects only 8 bits. In addition to rearranging the operations, ChaCha used slightly different cyclic shift constants. Salsa20 used shifts of 7, 9, 13, and 18 bits, while ChaCha uses shifts of 16, 12, 8, and 7 bits. This change was less significant for diffusion (the difference is considered insignificant in terms of cryptographic strength) and only slightly affects the speed on certain platforms. Thus, the main improvements in ChaCha emphasise the state update scheme.

ChaCha was presented in three versions based on the number of rounds: ChaCha8, ChaCha12, and ChaCha20 (similar to the Salsa20/8, /12, /20 line). The full version of ChaCha20 (20 rounds) is designed for maximum security and replaces Salsa20/20 without compromising overall performance. The reduced versions ChaCha8 and ChaCha12 were offered for environments where a lower margin of security is acceptable in favour of higher speed, similar to Salsa20/8 and Salsa20/12. However, in practice, the ChaCha20 version with the maximum number of rounds has become established in standardised applications, as even this version's performance remains high.

The cryptography community quickly analysed the new algorithm. J.P. Aumasson *et al.* (2008) published the results of cryptanalysis of Salsa20 and ChaCha (called "Latin Dances"), showing that ChaCha does indeed have higher round resistance: the best attack on ChaCha was able to cryptanalytically cover one round less than for Salsa20. It was possible to break (recover the key) for Salsa20, reduced to 8 rounds, with a complexity of ~ 2^{251} operations in the presence of ~ 2^{31} pairs of keystream outputs. For full-round Salsa20/20 and ChaCha20, no practical attacks are known except for key brute force. Over more than two decades of research into the cryptographic strength of ChaCha20, no effective practical attacks against the full 20-round version of the algorithm have been found. Known cryptanalytic results apply only to simplified versions of the algorithm with no more than 7–8 rounds and are mainly theoretical in nature, posing no real threat to the practical security of the algorithm. This indicates the high level of cryptographic strength of ChaCha20 and its significant security margin, which explains its widespread use in modern encryption systems.

Technical aspects of ChaCha: structure and cryptographic strength

Structure. ChaCha20 is a stream cypher based on addition-rotation-XOR (ARX) that transforms a 256-bit key, a 32-bit block counter, and a 96-bit nonce (a one-time random value) into a 512-bit block of pseudo-random keystream. The algorithm operates on an internal state of 16 words of 32 bits each (512 bits in total), which can be conveniently represented as a 4×4 matrix of words. The initial state is formed from the key, nonce, and constants as follows: the first row contains 4 fixed 32-bit constants

(ASCII string “expand 32-byte k”), the next eight words are a 256-bit key, followed by two words – the block counter (initialised to zero for the first block), and the last two words are 64 bits of “nonce” (in the original ChaCha, as in Salsa20, “nonce” is 64 bits; in Internet Engineering Task Force (IETF) implementations, 96 bits are used, with a slightly different breakdown). Thus, the initial 4×4 matrix is conditionally shown in Table 1.

Table 1. Initial state of the ChaCha20 algorithm matrix

const	const	const	const
key	key	key	key
key	key	key	key
ctr	ctr	nonce	nonce

Note: const – constant words “expa”, “nd 3”, “2-by”, “te k”; key – 256-bit key (divided into 8 words); ctr – counter; nonce – unique two-word number for each encryption stream

Source: A. Langley *et al.* (2016)

This state undergoes a series of rearrangement rounds, after which it is summed (component-wise in 32-bit chunks) with the initial state, forming a 512-bit keystream output block, which is then XORed byte-by-byte with the plaintext to obtain the ciphertext. When encrypting large amounts of data, the block counter automatically increments, generating subsequent 64-byte stream blocks. The main state transformation occurs through the repeated application of a quarter-round basic operation on four 32-bit words. Four words (labelled a, b, c, d) are taken as input to the quarter-round, and they are updated at the output. The sequence of operations in the ChaCha quarter-round is shown in the pseudocode “Quarter-round function of the ChaCha20 algorithm”:

```
QuarterRound(a, b, c, d)
a += b; d ^= a; d <<<= 16,
c += d; b ^= c; b <<<= 12,
a += b; d ^= a; d <<<= 8,
c += d; b ^= c; b <<<= 7.
```

Quarter-round processes a quarter of the entire 4×4 state matrix. ChaCha20 consists of 20 rounds, with quarter-round applied to four columns of the matrix in parallel in odd rounds (1, 3, 5, ...) and to four diagonals of the matrix in even rounds (2, 4, 6, ...). This alternating scheme (column round + diagonal round) ensures that the data is shuffled both by columns and by rows of the matrix, which in total over two rounds (double round) affects all 16 words. After executing the specified number of rounds (for ChaCha20, 20, i.e. 10 double rounds), the resulting state is added modulo 2^{32} to the initial state (this is the so-called Feed-Forward operation, inherited from the Salsa20 design). The result is interpreted as 64 bytes of key stream, ready for XOR with plaintext. The decryption process is identical to encryption, since XOR with the same key stream restores the plaintext. The structure of the algorithm is shown in Figure 1.

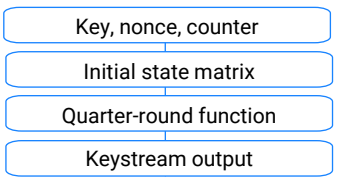

Figure 1. General diagram of the ChaCha20 algorithm
Source: A. Langley *et al.* (2016)

Notably, ChaCha uses only simple integer operations (addition, XOR, shift) that are implemented in hardware on all modern processors and are data-independent (no conditional jumps or indexed memory accesses). This renders ChaCha implementations resistant to timing attacks and side-channel attacks such as cache timing, which can be dangerous for cyphers with substitution tables (e.g., AES without hardware AES-NI). The algorithm is also not patented; the author has published several high-performance implementations in the public domain, which have facilitated its acceptance by the community.

Cryptographic resilience. The ChaCha20 algorithm is designed to provide 256-bit security, meaning that brute-forcing the key requires 2^{256} attempts, which is an astronomically large number. All known cryptanalysis methods failed to crack the full 20-round version. The best results relate to simplified versions with fewer rounds. For example, differential cryptanalysis and its variations have managed to find weaknesses in up to 7 rounds of ChaCha. For example, in 2018-2023, improved differential-linear attacks were reported that can distinguish a 7-round ChaCha permutation from a random one (selecting the output stream with a certain statistical difference) with a complexity significantly lower than brute force. However, these attacks do not lead to key recovery and do not scale to a larger number of rounds. For 8 rounds of ChaCha, the attack complexity is estimated to be around 2^{251}, which is only insignificantly better than a full key search (2^{256}) and is purely theoretical. Thus, from a practical point of view, ChaCha20 is considered reliable: even with the use of modern supercomputers or quantum accelerations (Grover’s algorithm provides insignificant acceleration compared to quadratic acceleration, effectively reducing the complexity to ~2_{128}), breaking the algorithm is unrealistic when used properly (Barbero *et al.*, 2022).

A substantial security requirement for ChaCha (as for most stream cyphers) is that the nonce must be unique for each key. If the same set {key, nonce, initial counter} is accidentally used twice to encrypt different messages, then due to the property of XOR with the same key stream, certain information regarding the two plaintexts (in particular, their XOR with each other) can be obtained. In the worst case, a repeated nonce compromises the messages.

Therefore, protocols with ChaCha20 require that nonce values for a single key never be repeated. The standard version of ChaCha20 (as defined in RFC 7539) uses a 96-bit nonce, which gives an astronomical number of possible values (around 2^{96}) – this is sufficient for most applications, although theoretically there is a small probabilistic chance of a nonce collision when generating random 96-bit numbers. To eliminate the risk of repetition, a modification of XChaCha20 called ChaCha20 "extended nonce" has been proposed, which supports a 192-bit nonce (Arciszewski, 2019). XChaCha20 first passes the key along with the first 128 bits of the nonce through HChaCha20, a 256-bit ChaCha20-based auxiliary permutation, generating a new internal key, after which encryption continues with regular ChaCha20 using this key and the remaining 64 bits of the nonce. This can be used for the safe use of extra-long nonces (24 bytes) with a collision probability of practically zero, which is beneficial for long-term sessions or systems where it is difficult to guarantee the uniqueness of short nonces. In terms of robustness, XChaCha20 maintains the same cryptographic assumptions as ChaCha20.

In summary, ChaCha20 has a simple and robust ARX structure, is well-suited for secure implementation in software, and demonstrates a high margin of safety against known attacks. Thanks to its combination of high performance, resistance to parallel computing, and well-analysed cryptographic security, ChaCha20 has become a popular choice in modern data protection protocols such as TLS, WireGuard, and QUIC. Its widespread adoption demonstrates the algorithm's recognition as an effective solution that meets modern security requirements.

Comparison of ChaCha with other cyphers (Salsa20 and AES).

ChaCha vs Salsa20. Since ChaCha is a direct descendant of Salsa20, it is necessary to consider their differences and similarities. Both algorithms use the same basic operations (32-bit addition, XOR, cyclic shifts) and work with a 512-bit state block formed from a 256-bit key and a 64-bit (or 96-bit) nonce. The performance of both ciphers is substantial: Salsa20 in software implementation achieves ~4-14 cycles per byte on standard CPUs, ChaCha20 has a similar order of magnitude. In fact, when designing ChaCha, D.J. Bernstein sought not to compromise speed relative to Salsa20. As stated in technical report, one round of ChaCha performs the same number of operations (16 additions, 16 XORs, 16 shifts) as a round of Salsa20 and maintains the same level of parallelism. In some architectures, ChaCha even saves one CPU register compared to the "native" implementation of Salsa20. Theoretically, ChaCha was expected to have a similar performance to Salsa, and possibly even better on certain platforms. Practical measurements confirmed this: for 8-round versions (ChaCha8 vs Salsa20/8), ChaCha showed the same or slightly better speed on most of the tested processors (for example, on 32-bit PowerPC G4 and x86 Pentium M, ChaCha8 was 6-8% faster than Salsa20/8). Only on some older 32-bit CPUs, such as Pentium 4, did Salsa20 outperform ChaCha (up to 30% faster), which is due to the architectural features of that platform. For full-round implementations, the difference is even smaller: ChaCha20 and Salsa20/20 have almost identical performance, with differences within a few percent depending on the environment. The results of the comparison between ChaCha and Salsa20 are shown in Table 2.

Table 2. Comparative table of ChaCha and Salsa20 cyphers

Characteristic	ChaCha	Salsa20
Algorithm type	Streaming, ARX	Streaming, ARX
Bloc size	512 bit	512 bit
Key size	256 bit	256 bit
Nonce size	64 or 96 bit (XChaCha: 192 bit)	64 or 96 bit (XSalsa: 192 bit)
Operations per round	16 additions, 16 XORs, 16 cyclic shifts	16 additions, 16 XORs, 16 cyclic shifts
Round types	Quarter-round with a diagonal column structure	Quarter-round with row-column structure
Typical performance values	~4-14 CPU cycles/bytes	~4-14 CPU cycles/bytes
Cryptographic resilience	Higher diffusion, fewer rounds to achieve equivalent security	Requires more rounds

Source: D.J. Bernstein (2008)

Hence, the transition from Salsa20 to ChaCha did not worsen performance but brought gains in robustness: ChaCha requires fewer rounds to achieve an equivalent level of security. Therefore, with the same security (e.g., 8 rounds of ChaCha vs 8 rounds of Salsa20), ChaCha is slightly better in terms of speed, and when using the full 20-round version, a greater margin of security is obtained with virtually no loss of performance. As a result, ChaCha has effectively replaced Salsa20 in the latest protocols: although Salsa20 is also considered secure, ChaCha20 has become predominant in implementations due to its better diffusion and community support.

ChaCha vs AES. These algorithms compete in real-world applications (e.g., AES-256 in CTR or GCM mode vs ChaCha20-Poly1305). AES is a 128-bit block cypher with a multi-round SP network (substitutions and permutations), optimised for hardware execution: modern processors contain AES-NI instructions that can encrypt an AES

block in 1-2 cycles. Thanks to this, AES-128-GCM is fast on desktop and server CPUs, often less than 1 clock cycle per byte when streaming large volumes. However, in environments without hardware support (this includes most mobile devices, many embedded systems, and some CPUs of other architectures), AES in software implementation runs significantly slower and is even potentially vulnerable to external attacks due to its dependence on S-box tables. In such cases, ChaCha20 demonstrates a clear advantage: it is designed specifically for efficient operation on general-purpose CPUs without special instructions. ChaCha20 consists entirely of operations that are performed uniformly fast on any processor: addition, XOR, shift (they do not require memory and do not cause cache misses) on simple ARM cores. The ChaCha20 cypher can be several times faster than AES-256 without AES-NI. A practical example: on a Galaxy Nexus smartphone (ARM Cortex-A9, without AES acceleration), decrypting 1 MB of data with the AES-128-GCM algorithm took ~41.6 ms, while ChaCha20-Poly1305 required only ~13.2 ms. This means that ChaCha20-Poly1305 is approximately three times faster than AES-128-GCM on this mobile device. This saves a significant amount of time and, crucially for portable devices, energy consumption (less CPU load means longer battery life) (Sullivan, 2014). AES-128-GCM and ChaCha20-Poly1305 provide a comparable level of security (~128 bits of effective strength, given the 128-bit randomness of authentication in GCM and Poly1305), so the comparison is valid.

In contrast, on platforms with hardware support for AES, the situation is opposite – hardware instructions make AES-GCM faster. For example, on Intel Haswell (and newer) AES-128-GCM outperforms ChaCha20-Poly1305 in throughput, especially on large data blocks. ChaCha20-Poly1305 uses only general-purpose SIMD instructions, and with the development of instruction sets (e.g., the proliferation of 512-bit AVX-512 vector instructions), the difference may narrow (Cai, 2022). On some ARM64 processors with neon optimisation, ChaCha20 performs at the level of AES. Therefore, when choosing between AES and ChaCha, the availability of hardware acceleration should be considered: for servers and desktops with AES-NI, AES-GCM is often more appropriate, while for mobile, IoT, and other devices without such acceleration, ChaCha20-Poly1305 offers a significant speed advantage without compromising security. Therefore, modern protocols provide for the possibility of using both, such as TLS 1.3, which defines cypher suites based on both AES-GCM (Rescorla, 2018) and ChaCha20-Poly1305, providing the optimal choice in a particular case (for example, the Chrome browser on Android will prefer ChaCha if the CPU does not have AES-NI). The results of the comparison between ChaCha and AES are shown in Table 3.

Table 3. Comparative table of ChaCha and AES cyphers

Characteristic	ChaCha	AES
Algorithm type	Streaming, ARX	Bloc, SP-network
Bloc size	512 bits	128 bits
Key size	256 bits	128/192/256 bit
Nonce size	64 or 96 bits (XChaCha: 192 bit)	96 bits
Operations per round	16 additions, 16 XORs, 16 cyclic shifts	Substitutions (S-Box), XOR, permutations
Round types	Quarter-round with a diagonal column structure	SP-round
Typical performance values	~4-14 CPU cycles/bytes	~20-60 CPU cycles/bytes
Cryptographic resilience	Higher diffusion, fewer rounds to achieve equivalent security	Requires AES-NI for speed and protection against cache attacks

Source: compiled by the authors based on D.J. Bernstein (2008)

Compared to the security of AES and ChaCha20, both algorithms have no practical vulnerabilities when used correctly. AES has a longer history of research, but ChaCha20 has also gained significant trust among experts. One of the advantages of ChaCha is its less complex and easier to implement structure – ~4,000 lines of code in a typical implementation versus tens of thousands in AES-based protocols (such as OpenVPN with different modes). Smaller and simpler code reduces the risk of implementation errors and simplifies security audits. On the other hand, AES, as a widely used standard, has a hardware implementation that minimises the risk of software bugs. Overall, both cyphers are considered reliable and are recommended by standards (for example, TLS 1.3 recommends the use of AES-GCM and ChaCha20-Poly1305). In 2014, A. Langley publicly endorsed ChaCha20-Poly1305 as an alternative to AES for mobile devices without hardware acceleration (Langley *et al.*, 2016). Thus, ChaCha20 has successfully complemented AES, providing an alternative that improves resistance to potential attacks on homogeneity (crypto-monoculture) and increases the flexibility of security systems.

Use of ChaCha in modern cryptographic systems

Since its inception, ChaCha (especially ChaCha20) has gradually gained recognition and integration into various protocols and standards. Below are the most relevant areas of application for this algorithm in modern systems. One of the first major implementations of ChaCha20 was its integration into the TLS protocol, which provides HTTPS web traffic encryption. The initiative came from Google:

in 2013, engineers were searching a fast and secure cypher stream to use in Chrome on Android and other clients without AES hardware acceleration. The choice fell on the ChaCha20 + Poly1305 combination, and in 2014, Chrome began supporting an experimental set of TLS cyphers with ChaCha20-Poly1305 (Sullivan, 2014).

After a period of testing and coordination, the IETF standardised this connection: first as an Internet draft CFRG (Crypto Forum Research Group), and later in RFC 7905, several CipherSuite TLS 1.2 were defined using ChaCha20-Poly1305 with HMAC (Hash-based Message Authentication Code) on SHA-256 for handshaking. The new standard has been supported in libraries such as OpenSSL and BoringSSL. This became particularly relevant after the dangerous RC4 (Rivest Cypher 4) was removed from TLS, and an alternative to AES-GCM was needed for older devices. In TLS 1.3 (standardised in 2018), the ChaCha20-Poly1305 suite became one of the core suites (Rescorla, 2018). According to the protocol requirements, all TLS 1.3 clients and servers must support two cypher suites: TLS_AES_128_GCM_SHA256 and TLS_CHACHA20_POLY1305_SHA256. This confirms ChaCha20-Poly1305's status as an equal component of modern web cryptography.

At the same time, ChaCha20-Poly1305 in TLS 1.3 is positioned as a solution that improves performance and resistance to side-channel attacks in software execution (Fig. 2). According to research, the implementation of ChaCha20 in HTTPS has significantly increased page loading speeds on mobile devices and reduced battery consumption. Currently, virtually all popular browsers (Chrome, Firefox, Safari, Edge) and servers (e.g., nginx via OpenSSL) support ChaCha20-Poly1305. Statistics show an increase in the share of TLS connections using this cypher: in the first years after the standardisation of TLS 1.3, the use of ChaCha20-Poly1305 increased sharply, as it is automatically selected for at least some clients (mainly mobile ones). For instance, P. Crowley & E. Biggers (2019) reported that a significant percentage of their network traffic is already protected by ChaCha20-Poly1305 thanks to Chrome/Firefox browsers, which dynamically select this cypher for clients without AES-NI.

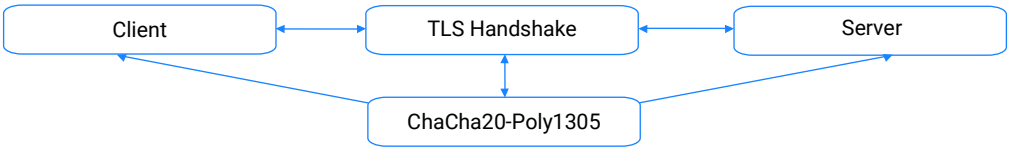

Figure 2. Use of ChaCha20-Poly1305 in TLS 1.3

Source: A. Langley *et al*. (2016)

Another area where ChaCha gained popularity was virtual private networks (VPNs) and other tunnelling protocols. The most famous example is the WireGuard protocol, introduced in 2016 as a modern alternative to IPsec and OpenVPN. WireGuard is designed with an emphasis on simplicity and speed and uses only ChaCha20 for symmetric encryption (with Poly1305 for authentication). The developers of WireGuard (Donenfeld, 2017) argued that ChaCha20 provides better or comparable encryption speed compared to AES, especially on typical router, server, and smartphone hardware, where AES instructions may be absent or where multithreading and simple code are more relevant. As a result, WireGuard is notable for the use of ChaCha20 for encryption, which is a faster alternative to the common AES-256 in other VPN protocols. Practical tests demonstrate that WireGuard outperforms OpenVPN in terms of throughput and latency, partly due to its choice of the lightweight ChaCha20 cypher and optimised implementation on kernels (running in OS kernel mode). The lack of algorithm choice (monoculture) in WireGuard is compensated for by thorough security analysis: ChaCha20-Poly1305 is used in the Noise_IK cryptographic protocol, and independent experts have confirmed the cryptographic strength of this solution. Thus, according to the conclusion of P. Crowley & E. Biggers (2019), ChaCha20 has become the basis for one of the most promising VPNs, which is already included in the Linux kernel and supported by many services.

Due to high performance on ARM processors, ChaCha20 has become popular not only for network traffic but also for data encryption in storage. In 2019, Google introduced the Adiantum algorithm (Crowley & Biggers, 2019), a new encryption mode for Android devices designed to protect disk storage on budget phones, smartwatches, and IoT devices that do not have AES hardware acceleration. Adiantum is based on XChaCha20-Poly1305, combined with a special permutation code (NH Poly1305 + enhancements) to achieve storage encryption length (i.e., without increasing data size), in contrast to standard GCM-type modes. The choice of ChaCha20 for Adiantum was due to the fact that on simple Cortex-A7 cores, the performance of AES-XTS (standard disk encryption mode) was insufficient, less than 50 MB/s, which slowed down the device. ChaCha20, on the other hand, can encrypt substantially faster by using only basic instructions. Adiantum can be used to encrypt all devices without a noticeable drop in performance: ChaCha20 is significantly faster than AES in the absence of hardware acceleration, while remaining secure. Adiantum is included in Android (starting with version 10) as an FBE (File-Based Encryption) option for devices that do not support AES-NI, thus protecting millions of budget smartphones around the world with ChaCha20. This is a case where a modern algorithm provides cryptographic protection to the "next billion" users, for whom encryption would otherwise be disabled due to the low performance of AES.

Another area of application is the generation of pseudorandom numbers for cryptography (e.g., /dev/urandom in operating systems). Traditionally, RC4 or other modern algorithms are used. Many implementations have switched to ChaCha20 as the core of the random stream generator. In particular, in the OpenBSD and FreeBSD operating systems, the arc4random() function is now implemented using ChaCha20 instead of the outdated RC4. The use of RC4 in arc4random has been criticised for its weak random number generation, as shown in a study by T. Ristenpart *et al.* (2009). The Linux kernel also uses ChaCha20 to initialise the entropy pool. Reasons: ChaCha20 is fast, has no known vulnerabilities, and is easy to implement without complex states. Thus, it provides a reliable stream of randomness for the cryptographic needs of the OS.

ChaCha20-Poly1305 is also used in other protocols: SSH (Secure Shell) (OpenSSH added this connection as an encryption option) – for interactive sessions, it provides less latency than AES. The Quick UDP Internet Connections (QUIC) protocol, developed by Google to replace TCP, also supports ChaCha20-Poly1305. Many high-level libraries (NaCl/libsodium, BoringSSL, etc.) offer ChaCha20 as one of their basic primitives. At the standards level, the algorithm is defined in RFC 7539 (Nir & Langley, 2015), recommended by the National Institute of Standards and Technology (NIST) for software implementations (Dworkin, 2016), and included in the list of algorithms approved for use in Internet standards. The international cryptography community is paying attention to both the analysis of ChaCha (to maintain trust) and the optimisation of its implementations for different platforms. For example, hardware implementations of ChaCha20 are being researched: there are IP cores for FPGAs that achieve throughputs of tens of Gbit/s, making ChaCha competitive even in high-end applications (Pfau *et al.*, 2019). Thus, ChaCha20 is widely integrated into critical security protocols (TLS, SSH, and IPsec via WireGuard) and data encryption implementations, especially where software performance is critical. It is one of the few new algorithms that has been able to occupy a niche alongside AES, complementing it.

Modifications and improvements to the ChaCha algorithm

Based on the success of ChaCha20, researchers and engineers further improved the algorithm, both in terms of increasing performance, enhancing security, and expanding areas of application (Procter, 2014). The main modifications and variants of ChaCha that appeared after its standardisation were considered. These improvements have made it possible to use this algorithm in more demanding environments, including resource-constrained systems, and to integrate it more effectively with other cryptographic protocols. Some variants, such as XChaCha20, have gained widespread support and are actively used in practice.

XChaCha20 is a variant with an extended nonce, designed primarily to eliminate the risk of repeating one-time values. The standard 96-bit nonce, although practically sufficient, can be exhausted with large amounts of encryption (the theoretical upper limit is 2^{32} blocks of 64 bytes, i.e. about 2^{38} bytes ~ 256 GB per key). XChaCha20 supports the use of a 192-bit nonce, which is practically inexhaustible. Technically, this is achieved using the HchaCha20 function – initialising a ChaCha20-like permutation on a 128-bit nonce to obtain an intermediate key. At the same time, the security of XchaCha20 is based on the same assumptions as ChaCha20, and the relevant IETF drafts (draft-irtf-cfrg-xchacha) recommend it for scenarios where very large data needs to be encrypted or long-term keys need to be maintained without the risk of nonce repetition (Arciszewski, 2019). XChaCha20 has already been implemented in popular libraries, such as libsodium, and is used in protocols (WireGuard uses 24-byte nonces for internal needs). This modification is aimed more at system reliability than at changing the core of the algorithm, but it is significant for the practical application of ChaCha in large-scale systems (Fig. 3).

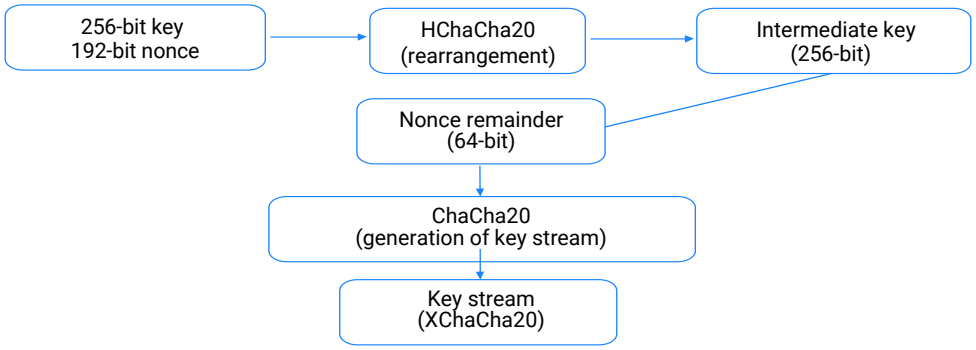

Figure 3. XChaCha20 algorithm diagram with extended nonce

Source: A. Langley *et al.* (2016)

ChaCha20-Poly1305 (AEAD) is not a modification of ChaCha itself, but a combination with MAC. The authenticated mode AEAD_CHACHA20-POLY1305 has become the de facto standard for using ChaCha in protocols. Before the advent of this AEAD, ChaCha20 was also proposed for use with traditional MACs (e.g. HMAC-SHA1), but during

the standardisation of TLS, it was decided to use Poly1305 as a faster and simpler option. Poly1305 is an integrity verification algorithm (MAC), also invented by D. Bernstein, optimised for 64-bit multiplication operations. The ChaCha20+Poly1305 combination provides both confidentiality and authenticity of messages; these algorithms are now mentioned together. Notably, Poly1305 obtains its key from ChaCha20 (by encrypting 32 bytes of zeros with a separate single-byte block with a special "nonce"), which eliminates the problems of MAC key reuse. This pair has undergone numerous security tests and has repeatedly proven its robustness in practice. In the context of the present study, ChaCha20-Poly1305 can be considered a composition that has significantly expanded the scope of ChaCha, making it a full-fledged AEAD component for modern protocols.

Although ChaCha20 dominates the standards, in some cases, options with fewer rounds are considered for higher speed. ChaCha12 provides a substantial level of security (all known attacks do not progress beyond 7-8 rounds); therefore, it is considered potentially secure. Google used ChaCha12 in Android Disk Encryption prototypes (before Adiantum) to accelerate encryption and noted the absence of known vulnerabilities at 12 rounds. Some implementations (e.g., OpenSSL) can be configured to reduce the round count of ChaCha for experimentation or specific needs. However, ChaCha12/8 has not gained widespread use because the speed advantage is not significant enough (ChaCha20 is already highly efficient) to sacrifice robustness. Therefore, ChaCha20 is usually used as the representative variant in performance reviews. But the ability to adjust rounds – the flexibility of the algorithm – is sometimes used by researchers to test security limits or speed gains (Maitra, 2016; Dey & Sarkar, 2023; Xu *et al.*, 2024).

On the other hand, in 2023, modifications to ChaCha were proposed with an increase in the number of rounds for even greater stability. For example, V.R. Kebande (2023) presents Extended-ChaCha20 (EchaCha20), a variant that uses 36 quarter-rounds (i.e., 36 rounds or 18 double rounds) instead of 20. The motivation is to increase the difficulty of differential attacks and improve the results of statistical randomness tests. The author tested EchaCha20 and showed that the algorithm successfully passes all NIST STS randomness tests, similar to ChaCha20, while demonstrating comparable performance to the original. In particular, the encryption/decryption time and memory usage remained unchanged when moving from 20 to 36 rounds, making EchaCha20 interesting for applications where maximising resilience is critical (perhaps in the context of post-quantum considerations, although a 256-bit key is already sufficient with a large margin). Currently, EchaCha20 is an academic proposal, but it demonstrates that the algorithm can be scaled towards security without significant performance losses. Such research expands the ChaCha's limits and may be reflected in future standards or specific protocols.

Although ChaCha20 is designed for software implementation, there are developments in hardware implementations. Some IP core manufacturers, including Silex Insight, have introduced scalable FPGA/ASIC cores ChaCha20-Poly1305, capable of providing throughput of up to 100 Gbps on FPGA and up to 800 Gbps on ASIC, which meets the modern requirements of high-performance encryption systems (Rashidi, 2024) This is achieved thanks to ChaCha's high level of parallelisation – for example, processing several blocks simultaneously on the pipeline. Such solutions can be used in network filters and VPN gateways, where AES is also implemented. Notably, even in a hardware environment, ChaCha can compete: the implementations cited achieve <1 cycle per byte on an FPGA, which is comparable to AES. Thus, ChaCha20 is gradually penetrating the hardware world. Modern implementations of ChaCha20-AEAD (e.g., the leancrypto library, version 1.5.1, 2025) actively use Intel AVX2, AVX-512, and ARM NEON CPU instructions to achieve high-performance SIMD acceleration on x86_64 and ARMv8 architectures. Each new generation of processors reduces the gap between AES and ChaCha or improves ChaCha's already substantial performance. For example, on x86 processors with AVX2, the ChaCha20 implementation can encrypt at a rate of ~12-15 GB/s per core, which is sufficient for most tasks, and with the release of AVX-512, this speed is expected to double.

In summary, the ChaCha ecosystem continues to evolve: variants have been developed for different needs (XChaCha20 – for nonce uniqueness, ChaCha20-Poly1305 – for AEAD, truncated/enhanced round versions – for speed/security balance), and methods of implementing the algorithm in both software and hardware are being improved. These modifications strengthen ChaCha's position as a long-term stream cypher standard. In the context of the active support of the community and industry, further expansion of the algorithm's applications can be expected. ChaCha20, thanks to its flexibility and high security, is already the basis for numerous modern cryptographic solutions.

Analysis of ChaCha's performance compared to other cyphers

One reason for ChaCha's popularity is its exceptional performance on different platforms. Fast performance on desktop CPUs. As already mentioned, the AES-GCM algorithm has an advantage on modern x86-64 processors with AES-NI support. Measurements conducted by Cloudflare showed that for message sizes > 1 KB, AES-128-GCM encrypts at a rate of ~1.5-1.8 cycles per byte, while ChaCha20-Poly1305 encrypts at ~2.0-2.3 cycles per byte. For small messages, ChaCha may even be slower due to the lack of specialised instructions (for example, at 64 bytes, AES significantly outperforms it). However, the difference is not critical: even 2 cycles/byte means that on a single 3 GHz core, ~1.5 GB of data per second can be encrypted, which is more than the throughput of most network interfaces. Therefore, for server applications where AES-NI is present, ChaCha20-Poly1305 is enabled primarily not for speed, but as a fallback in case AES becomes unsafe to use or if the client does not support AES-NI (Nir & Langley, 2015; Dworkin, 2016; Krasnov, 2016).

This is where ChaCha20 is the most utilised. For example, on popular 32-bit ARM Cortex-A7 processors (often used in inexpensive phones and IoT devices), AES encrypts ~10-20 MB/s (in CBC/CTR mode) due to the lack of acceleration, while ChaCha20 on the same core delivers 60+ MB/s. The situation is better on 64-bit ARMv8: Crypto Extensions instructions have appeared, but not all devices have them (for example, many mid-range smartphones from 2016–2018 had ARMv8 without AES instructions enabled) (Sullivan, 2014). In such cases, disabling AES-NI in OpenSSL benchmarks showed ChaCha20-Poly1305 accelerating up to 3-4 times over AES-256-GCM. This is consistent with the independent Google tests mentioned earlier. That is, for mobile platforms, ChaCha20 can process approximately 300-500 MB/s per core, while AES-GCM without hardware support barely reaches 100-150 MB/s. This gap was the decisive argument in favour of implementing ChaCha20 in Android, iOS and other mobile systems (Sullivan, 2014).

Another aspect of performance is the delay in processing small messages. In protocols such as SSH or VPN, where small packets are transmitted, minimal encryption latency is relevant. ChaCha20 has an advantage as it does not require complex dependent operations (AES has several rounds with non-linearities that are difficult to pipeline without hardware deployment). Thus, ChaCha can have less encryption delay per packet. For example, WireGuard has a better ping compared to AES-based VPNs precisely because of the lighter ChaCha encryption. ChaCha parallelisation is also possible: several 64-byte stream blocks can be generated simultaneously on different cores or SIMD chains, as the blocks are independent (different counter values). AES-CTR/GCM is similarly parallelised by blocks; therefore, there is parity. But ChaCha is also less resource-intensive: its implementation requires less memory and consumes less energy for the same amount of work (due to fewer common instructions, although this is a subtle point).

In microcontrollers, such a comparative analysis also favours ChaCha. On 8-bit/16-bit MCUs, AES is generally difficult to implement efficiently (due to byte-wise S-box processing), whereas ChaCha consists of simple operations that scale even on small bit depths (albeit slower) (Zinzindohoué *et al.*, 2017; Tsoupidi *et al.*, 2021). Some IoT protocols, such as the Noise Protocol Framework, support the use of ChaCha20 for secure connections between microcontrollers. Thus, ChaCha20 extends secure communication capabilities to devices where AES is not optimal. The comparison results are shown in Table 4.

Table 4. Comparative table of ChaCha20 and AES cyphers when processing small messages

Characteristic	ChaCha20	AES
Latency	Low, due to simple linear operations (addition, XOR, shifts) that are easily pipelined on the CPU without special instructions	Higher, due to complex non-linear operations (S-box), which are less easily pipelined without hardware acceleration (AES-NI)
Parallelisation	High encryption blocks are independent (different key stream blocks are generated in parallel)	High encryption blocks are independent (different key stream blocks are generated in parallel)
Resource consumption	Low, does not use large tables, smaller code size	Higher, especially without AES-NI (S-box tables required, larger code)
Energy consumption	Low, due to fewer simple operations (lower CPU load)	Higher, especially without AES-NI hardware instructions (higher complexity of operations, more CPU cycles)
Efficiency on 8/16-bit BSS	High, simple arithmetic operations are easily implemented on limited CPUs.	Low, complex and slow implementation via S-box on low-bit CPUs
Practical application	Wide, often used in IoT protocols, including Noise Protocol Framework and WireGuard for low-power devices	Limited, usually requires AES-NI hardware support or is very slow

Source: compiled by the authors

ChaCha20 is one of the fastest known symmetric cyphers on software platforms. It outperforms older stream cyphers (such as RC4) not only in security but also in speed, especially on modern CPUs capable of processing 32-bit operations in a single clock cycle. Compared to AES, ChaCha20 loses only when AES is supported by a specialised hardware module; in all other cases, ChaCha20 is at least as good, and often significantly better. This has made it the "default cypher" for many software implementations where cross-platform compatibility and constant execution time are substantial factors. J.P. Aumasson *et al.* (2008) emphasised in their technical review that ChaCha20-Poly1305 is automatically selected as the optimal alternative for all clients without AES-NI. Their results confirmed that ChaCha's performance is maintained even in the absence of specialised hardware acceleration, which is consistent with the findings of this study regarding the high efficiency of the algorithm on a wide range of platforms. ChaCha's performance scales up with increasing processor capabilities: larger register widths, more cores – all of which can be easily leveraged for acceleration, as the algorithm is linearly parallelisable. From a practical standpoint, ChaCha20-Poly1305 can be recommended for any system where hardware AES is not available or where the simplest and most reliable implementation is required. This approach is becoming

increasingly common in the industry, as confirmed by performance analysis data.

Conclusions

The ChaCha20 algorithm, created as an evolutionary development of Salsa20, is one of the key components of modern cryptography. After analysing its history, structure, properties, and applications, the following conclusions can be drawn. First, ChaCha has successfully achieved the goals set for improving Salsa20: by changing the round structure, it has increased diffusion per round without losing performance, as confirmed by both theoretical estimates and the absence of effective attacks on the full version. Second, ChaCha20 has demonstrated outstanding performance in software implementations, being a "mobile AES replacement": in environments without AES hardware acceleration, it provides a multiple speed advantage with equivalent cryptographic strength. This has had a direct impact on the industry: ChaCha20-Poly1305 has become the standard in TLS 1.3, VPN (WireGuard) and other protocols, where it has reduced the load on devices and improved energy efficiency. ChaCha20-Poly1305 was formally standardised in RFC 7539, which defines the AEAD construction format for modern security protocols. Thirdly, ChaCha20 is a flexible and adaptable algorithm: modifications (XChaCha20 for larger nonces, AEAD modes, versions with different numbers of rounds) and improvements have been created based on it, expanding its scope from web protocols to disk encryption and random number generation. It is well-suited for hardware implementations and scales to multi-core systems, which indicates the potential for using ChaCha in future solutions.

The value of ChaCha20 has been confirmed by its widespread adoption: in most modern cryptosystems, ChaCha20 is considered an option or standard. This algorithm has significantly improved the security and performance of many systems, making encryption more accessible to weaker devices. Overall, ChaCha20 was proved to be a reliable and effective stream cypher that complements classic algorithms and reduces the security ecosystem's dependence on a single solution (AES), thereby increasing the resilience of the entire cryptographic infrastructure. In the future, further research on ChaCha may focus on formal analysis of its robustness (e.g., proving resistance to certain attack models) and finding the optimal balance of rounds for different applications. In addition, the integration of ChaCha20 into new protocols (e.g., post-quantum hybrid VPN or QUICv2 schemes) and the study of its behaviour when interacting with quantum-resistant algorithms are also relevant topics.

Acknowledgements

None.

Funding

The study was not funded.

Conflict of Interest

None.

References

[1] Arciszewski, S. (2019). *XchaCha: eXtended-nonce ChaCha and AEAD_XchaCha20_Poly1305. Internet-Draft draft-irtf-cfrg-xchacha-01*. Retrieved from https://datatracker.ietf.org/doc/html/draft-irtf-cfrg-xchacha-01.

[2] Aumasson, J.P., Fischer, S., Khazaei, S., Meier, W., & Rechberger, C. (2008). New features of Latin dances: Analysis of Salsa, ChaCha, and Rumba. In K. Nyberg (Eds.), *Fast software encryption. FSE 2008. Lecture notes in computer science* (Vol. 5086, pp. 470-488). Berlin: Springer. doi: 10.1007/978-3-540-71039-4_30.

[3] Barbero, S., Bazzanella, D., & Bellini, E. (2022). Rotational cryptanalysis on ChaCha stream cipher. *Symmetry*, 14(6), article number 1087. doi: 10.3390/sym14061087.

[4] Bernstein, D.J. (2008). The Salsa20 family of stream ciphers. In M. Robshaw & O. Billet (Eds.), *New stream cipher designs. Lecture notes in computer science* (Vol. 4986, pp. 84-97). Berlin: Springer. doi: 10.1007/978-3-540-68351-3_8.

[5] Cai, W. (2022). Implementation and optimization of ChaCha20 stream cipher on Sunway taihuLight supercomputer. *The Journal of Supercomputing,* 78(3), 4199-4216. doi: 10.1007/s11227-021-04023-9.

[6] Crowley, P., & Biggers, E. (2019). *Introducing Adiantum: Encryption for the next billion users*. Retrieved from https://security.googleblog.com/2019/02/introducing-adiantum-encryption-for.html.

[7] Datadog Security. (n.d.). *RC4 encryption is now insecure*. Retrieved from https://docs.datadoghq.com/security/code_security/static_analysis/static_analysis_rules/go-security/import-rc4/.

[8] De Santis, F., Schauer, A., & Sigl, G. (2017). ChaCha20-Poly1305 authenticated encryption for high-speed embedded IoT applications. In *Design, automation & test in Europe conference & exhibition* (pp. 692-697). Lausanne: IEEE. doi: 10.23919/DATE.2017.7927078.

[9] Degabriele, J.P., Govinden, J., Günther, F., & Paterson, K.G. (2021). The security of ChaCha20-Poly1305 in the multi-user setting. In *Proceedings of the 2021 ACM SIGSAC conference on computer and communications security (CCS'21)* (pp. 1981-2003). New York: ACM. doi: 10.1145/3460120.3484814

[10] Dey, C., & Sarkar, S. (2023). A new distinguishing attack on reduced round ChaCha permutation. *Scientific Reports*, 13, article number 13958. doi: 10.1038/s41598-023-39849-1.

[11] Donenfeld, J. (2017). WireGuard: Next generation kernel network tunnel. NDSS 2020. In *Network and distributed system security symposium* (article number 4846ada1492f5d92198df154f48c3d54205657b). San Diego: NDSS. doi: 10.14722/ndss.2017.23160.

[12] Dworkin, M. (2016). *Recommendation for block cipher modes of operation: Galois/Counter Mode (GCM) and GMAC.* Gaithersburg: NIST. doi: 10.6028/NIST.SP.800-38D.
[13] Gülmezoglu, B., Irazoqui, G., Eisenbarth, T., & Sunar, B. (2019). Cross-VM cache attacks on AES. *IEEE Transactions on MultiScale Computing Systems*, 2(3), 211-222. doi: 10.1109/TMSCS.2016.2550438.
[14] Kebande, V.R. (2023). Extended-ChaCha20 stream cipher with enhanced quarter round function. *IEEE Access*, 11, 114220-114237. doi: 10.1109/ACCESS.2023.3324612.
[15] Krasnov, V. (2016). *It takes two to ChaCha (Poly)*. Retrieved from https://blog.cloudflare.com/it-takes-two-to-chacha-poly.
[16] Langley, A., Chang, W., Mavrogiannopoulos, N., Strombergson, J., & Josefsson, S. (2016). *ChaCha20-Poly1305 Cipher Suites for TLS (RFC 7905)*. Retrieved from https://www.rfc-editor.org/rfc/rfc7905.html.
[17] Maitra, S. (2016). Chosen IV cryptanalysis on reduced-round ChaCha and Salsa. *Discrete Applied Mathematics,* 208, 88-97. doi: 10.1016/j.dam.2016.02.020.
[18] Najm, Z., Jap, D., Jungk, B., Picek, S., & Bhasin, S. (2018). On comparing side-channel properties of AES and ChaCha20 on microcontrollers. In *IEEE Asia Pacific conference on circuits and systems (APCCAS)* (pp. 552-555). Chengdu: IEEE. doi: 10.1109/APCCAS.2018.8605653.
[19] Nir, Y., & Langley, A. (2015). *ChaCha20 and Poly1305 for IETF protocols (RFC 7539)*. Retrieved from https://www.rfc-editor.org/rfc/rfc7539.html.
[20] Pfau, J., Reuter, M., Harbaum, T., Hofmann, K., & Becker, K. (2019). A hardware perspective on the ChaCha ciphers: Scalable ChaCha8/12/20 implementations ranging from 476 slices to bitrates of 175 Gbit/s. In *32nd IEEE international system-on-chip conference (SOCC)* (pp. 294-299). Singapore: IEEE. doi: 10.1109/SOCC46988.2019.1570548289.
[21] Polubelova, M., Bhargavan, K., Protzenko, J., Beurdouche, B., Fromherz, A., Kulatova, N., & Zanella-Béguelin, S. (2020). HACLxN: Verified generic SIMD crypto (for all your favourite platforms). In *Proceedings of the 2020 ACM SIGSAC conference on computer and communications security (CCS'20)* (pp. 899-918). New York: ACM. doi: 10.1145/3372297.3423352.
[22] Procter, G. (2014). *A security analysis of the composition of ChaCha20 and Poly1305*. Retrieved from https://eprint.iacr.org/2014/613.pdf.
[23] Rashidi, B. (2024). High-performance hardware structure of ChaCha20 stream cipher based on sparse parallel prefix adder. *International Journal of Circuit Theory and Applications*, 53(5), 2947-2957. doi: 10.1002/cta.4264.
[24] Rescorla, E. (2018). *The transport layer security (TLS) protocol version 1.3. (RFC 8446)*. Retrieved from https://www.rfc-editor.org/rfc/rfc8446.html.
[25] Ristenpart, T., Tromer, E., Shacham, H., & Savage, S. (2009). Hey, you, get off of my cloud: Exploring information leakage in third-party compute clouds. In *Proceedings of the 16th ACM conference on computer and communications security (CCS 2009)* (pp. 199-212). Chicago: ACM. doi: 10.1145/1653662.1653687.
[26] Serrano, R., Duran, C., Sarmiento, M., Pham, C., & Hoang, T. (2022). ChaCha20-Poly1305 AEAD for transport layer security 1.3. *Cryptography*, 6(2), article number 30. doi: 10.3390/cryptography6020030.
[27] Sullivan, N. (2014). *Do the ChaCha: Better mobile performance with cryptography*. Retrieved from https://blog.cloudflare.com/do-the-chacha-better-mobile-performance-with-cryptography.
[28] Tsoupidi, R.-M., Balliu, M., & Baudry, B. (2021). Vivienne: Relational verification of cryptographic implementations in WebAssembly (verifies ChaCha20/Poly1305 in WHACL*). *ArXiv*. doi: 10.48550/arXiv.2109.01386.
[29] Xu, Z., Xu, H., Tan, L., & Qi, W. (2024). Improved differential-linear cryptanalysis of reduced-round ChaCha permutation. *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, 2024(2), 166-189. doi: 10.46586/tosc.v2024.i2.166-189.
[30] Zinzindohoué, J.-K., Bhargavan, K., Protzenko, J., & Beurdouche, B. (2017). HACL*: A verified modern cryptographic library. In *Proceedings of the 2017 ACM SIGSAC conference on computer and communications security (CCS'17)* (pp. 1789-1806). New York: ACM. doi: 10.1145/3133956.3134043.

ChaCha: розвиток та модифікація Salsa20 у сучасних криптографічних системах

Олексій Палій
Аспірант
Вінницький національний технічний університет
21021, вул. Хмельницьке шосе, 95, м. Вінниця, Україна
https://orcid.org/0009-0006-8387-3609

Олександр Дудник
Кандидат технічних наук, доцент
Вінницький національний технічний університет
21021, вул. Хмельницьке шосе, 95, м. Вінниця, Україна
https://orcid.org/0009-0005-3684-965X

Анотація. У статті проведено огляд потокового шифру ChaCha20 як спадкоємця алгоритму Salsa20, зосереджено увагу на його розвитку, технічних особливостях та застосуванні в сучасних криптосистемах. Актуальність роботи обумовлена широким впровадженням ChaCha20 у протоколи безпеки (TLS 1.3, VPN тощо) завдяки високій швидкодії в програмних реалізаціях та стійкості до криптоаналізу. Метою роботи було проаналізувати еволюцію ChaCha від Salsa20, порівняти його з іншими шифрами та узагальнити останні досягнення щодо модифікацій і продуктивності. У межах дослідження використовувались методи аналізу літературних джерел і експериментальних даних про швидкодію та стійкість шифрів. Основні результати включають висвітлення історії створення ChaCha на основі Salsa20 та покращення дифузії за раунд, детальний опис структури алгоритму (матриця стану 4×4, операції додавання-обертання-XOR) і його криптостійкості (відсутність практичних атак на повну 20-раундову версію). Показано переваги ChaCha20 над Advanced Encryption Standard (AES) в програмному середовищі – зокрема, на платформах без апаратного прискорення AES ChaCha20 працює до 3 разів швидше при еквівалентному рівні безпеки. Розглянуто впровадження ChaCha20-Poly1305 в TLS і WireGuard, а також використання XChaCha для подовжених «nonce» і алгоритму Adiantum для шифрування дисків на мобільних пристроях. Проаналізовано сучасні модифікації ChaCha (наприклад, збільшення кількості раундів) та їх вплив на продуктивність і безпеку. Практична цінність огляду полягає в узагальненні сучасного досвіду використання ChaCha20, що може бути корисним для вибору криптоалгоритмів у ресурсно-обмежених системах і подальших досліджень у галузі потокових шифрів

Ключові слова: потоковий шифр; Advanced Encryption Standard; TLS 1.3; криптоаналіз; ресурсно-обмежені системи

Journal homepage: https://itce.vn.ua/en

Information Technologies and Computer Engineering **Vol. 23, No. 1. 2026**

Article's History: Received: 09.12.2025 Revised: 03.03.2026
Accepted: 26.03.2026 Published: 20.04.2026

ISSN 1999-9941; e-ISSN 2078-6387
UDC 004.8:004.415.53
DOI: 10.31649/vitce/1.2026.22

Prompt engineering for large language models in test case generation

Anatolii Husakovskyi*
Master, Senior Engineer
National Aerospace University "Kharkiv Aviation Institute"
61070, 17 V. Manko Str., Kharkiv, Ukraine
https://orcid.org/0009-0007-9398-0966

Abstract. The relevance of the study is determined by the need to enhance the effectiveness of software testing, where the use of large language models and prompt engineering techniques opens new opportunities for the automated generation of high-quality test cases. The purpose of the study is to evaluate the effectiveness of prompt engineering strategies in test case generation by large language models. The methodology is based on a comparison of four prompt engineering techniques, namely zero-shot, few-shot, chain-of-thought, and role prompting, for unit test generation using the CodeLlama 2 and StarCoder language models in the PyTest and JUnit environments, with evaluation according to the criteria of code coverage, relevance, defect detection, and integration suitability. The analysis demonstrated that few-shot and role prompting provide the best balance between the quantity and quality of tests, with coverage of 85-100% and relevance of 88-95%, whereas chain-of-thought proved effective for complex logic and identified 16 of 20 embedded defects (80%), while zero-shot was limited to basic checks with coverage of 55-65% and accuracy of 70-75%. CodeLlama 2 demonstrated stable test generation with high consistency across repeated queries (90%), an average generation time of 16.2 s, and 52 tests per module, covering basic and complex scenarios, including edge cases and exceptions. StarCoder demonstrated higher speed (14.7 s), generated 50 tests with slightly lower stability (87%) and reduced coverage of complex scenarios, which rendered it effective for rapid validation of basic functions. The highest levels of readability, modularity, and integration suitability for CI/CD pipelines were observed with role prompting, whereas few-shot ensured a strong balance between structured output and practical test readiness, while chain-of-thought and zero-shot exhibited specific limitations. Combined use of models and prompting strategies enables optimisation of the test generation process, enhancing relevance, coverage, and the effectiveness of automated testing. The results of the study may be applied in automated software testing, integration into continuous integration and delivery pipelines, and training of quality assurance engineers in effective test generation methods

Keywords: CodeLlama; StarCoder; zero-shot prompting; few-shot prompting; chain-of-thought prompting; role prompting; CI/CD integration

Introduction

The relevance of this study is determined by the rapid proliferation of large language models (LLMs) in the field of software development and the need to enhance the effectiveness of testing processes. Software testing conventionally requires substantial resources and time, whereas the quality and completeness of test cases directly affect the reliability of the final product. The use of LLMs opens new opportunities for automated test generation; however, the effectiveness of this approach largely depends on the way prompts are constructed. In contemporary Ukrainian academic discourse, studies devoted to large language models increasingly focus on the effectiveness of prompt engineering and model interpretability in applied domains. In the study by I. Yurchak *et al.* (2024), prompting techniques aimed at improving the productivity and controllability of large language models are analysed in detail. The researchers demonstrated that an optimal combination of role prompting and few-shot examples ensures a reduction

Suggested Citation:
Husakovskyi, A. (2026). Prompt engineering for large language models in test case generation. *Information Technologies and Computer Engineering*, 23(1), 22-34. doi: 10.31649/vitce/1.2026.22

*Corresponding author

in hallucination rates and an increase in accuracy in tasks involving the generation of program code or analytical texts. The study by S. Levitskyi & V. Mokin (2025) placed emphasis on evaluating the robustness of large language models to manipulative and disinformation influences; however, the obtained results have a direct connection with issues of effective prompting. The researchers showed that variations in prompt formulation may lead to significant differences in model responses, particularly in contexts where adherence to factual accuracy is critical. This study thus confirms the key role of ethical and semantic control over prompt engineering in ensuring the credibility of results. A different perspective is proposed by A. Novakovsky & I. Yalovega (2025), who conducted a categorisation of the cognitive and analytical capabilities of large language models, including aspects of their learning behaviour under different prompt types. The researchers demonstrated that the ability of LLMs to generalise, argue, and construct logical relations depends substantially on the semantic structure of prompts and the level of task contextualisation. They emphasise that context-enriched prompts enhance the capacity of models for multi-step reasoning, while structured instructions strengthen the reproduction of complex cause-and-effect relationships.

The study by L. Naimi *et al.* (2024) proposed an approach for the automatic generation of test cases from use case diagrams through the application of LLMs and prompt engineering. In their approach, prompts are adapted to the specific characteristics of software modules and usage scenarios, which ensures increased functional coverage and test correctness. A distinctive feature lies in the focus on real business processes, which enables the verification of basic functionality, edge cases, and exceptional situations in complex systems. P. Sahoo *et al.* (2024), in a systematic review, summarised the main techniques and areas of applying prompt engineering in LLMs, identifying key strategies for prompt optimisation, methods for the combined use of zero-shot and few-shot approaches, and techniques for enhancing response stability in models. The researchers stressed that the effectiveness of test generation directly depends on the selection of prompts and their adaptation to the specific nature of software engineering tasks. G. Adu (2024) examined the application of LLMs for the automatic generation of scenarios and test cases in software contexts through combining prompt engineering, fine-tuning, and Retrieval Augmented Generation. The study highlighted those combined approaches enable an increase in functional coverage and improve test correctness and relevance, which is particularly important for complex business logic. The study demonstrated that adapted prompts allow LLMs to operate effectively even with non-standard or rare scenarios. S. Alagarsamy *et al.* (2025) focused on optimising LLMs for the generation of test cases from textual descriptions. The paper demonstrated that appropriately selected prompt engineering strategies improve the accuracy of generating both basic and complex tests, particularly for multi-component modules. The researchers emphasised that the integration of LLMs into automated testing processes reduces test preparation time and decreases the number of human errors. S. Lim & R. Schmälzle (2023) extended the application of prompt engineering beyond traditional software engineering by demonstrating the effectiveness of LLMs in generating messages for healthcare. Despite the different domain, the study confirmed the importance of accurate prompt formulation for achieving relevant and structured results, which bears direct relevance to the construction of precise test scenarios in software engineering.

Research conducted from 2021 to 2025 also confirms that large language models not only perform code generation tasks but also demonstrate a high level of autonomy in prompt formulation. S. Anasuri (2024) focused on the development of best practices in prompt engineering for code generation tools. The researcher emphasised that model performance largely depends on clear task formulation, structured examples, and the use of role-based context. The study highlighted the importance of standardising approaches to prompt formulation to ensure reproducibility of results and high quality of test scenarios, which is particularly relevant in the context of CI/CD pipelines and integrated testing of large software systems.

Previous studies showed that prompt engineering in large language models considerably increases the effectiveness of test case generation and the quality of automated testing. However, most studies focused on isolated aspects, such as comparisons of different prompt types or the optimisation of specific models, without a comprehensive empirical evaluation of test generation effectiveness in real-world software projects with different programming languages and integration into CI/CD pipelines. The purpose of this study was to determine how different prompt construction strategies influence the ability of large language models to generate high-quality test scenarios with adequate code coverage and defect detection in real projects. The following tasks were defined: to analyse the influence of prompt types on the quality of the generated tests, their structure, readability, and integration into CI/CD pipelines; to compare the performance of LLMs (CodeLlama 2 and StarCoder) in test generation based on code coverage metrics, test relevance, and the ability to detect embedded defects.

Materials and Methods

The study utilised an experimental applied approach, since it involved practical verification of the effectiveness of different prompt engineering techniques in real software projects. Experiments were conducted during 2024 and May 2025 in a virtualised environment using current versions of testing and CI/CD tools. Large language models CodeLlama 2 and StarCoder were used in the study, both demonstrating a high level of effectiveness in code generation and code understanding tasks. Open-source medium-sized software projects written in Python and Java were selected for empirical verification. These projects contained modules with clearly defined business logic, which made it possible

to assess the relevance and completeness of the generated test cases in the context of real scenarios. Experiments were performed in a virtualised environment with configured test automation tools PyTest 8.2.0 and Junit 5.11.0, integrated into the GitHub Actions CI/CD pipeline. This selection ensured the reproducibility of the study and enabled verification of the practical applicability of the results for contemporary software development processes.

Four approaches to prompt formulation were developed. In the case of zero-shot prompting, the models received only the task without any additional examples. Few-shot prompting involved the provision of several illustrative examples intended to guide the LLM towards the desired response format. Chain-of-thought prompting emphasised the necessity of intermediate reasoning steps, which ensured more detailed logic in test creation. Role prompting assigned the models a specific role-based context: an experienced tester was applied during the generation of functional and negative test cases, while a quality assurance engineer was used when developing scenarios for the evaluation of stability and defect reproducibility. Unified templates were prepared for each approach, guiding the generation towards the creation of unit tests with an emphasis on functional capabilities and exception handling.

The research process consisted of consecutive stages that enabled a comprehensive evaluation of the effectiveness of the proposed approaches. Specific software modules requiring test creation were first selected. Prompts were then formulated in accordance with the defined prompt engineering techniques. Direct test generation by the language models followed this step. The obtained results were integrated into the source code of the projects and verified using PyTest and JUnit, which enabled evaluation of the operability of the created test scenarios under real conditions. Various usage scenarios were modelled for a full evaluation of effectiveness: standard functional verification, testing of exceptional situations, and detection of intentionally embedded defects. This ensured a comprehensive evaluation of the ability of the models to cover functionality and identify errors.

The quality of the obtained test cases was analysed using several complementary criteria. Code coverage was regarded as the priority indicator, defined as the percentage of functions and methods covered by the created tests. The next aspect was relevance, which implied alignment of test logic with the functionality of the software module and the absence of incorrect checks. Attention was devoted to defect detection, namely the ability of tests to identify pre-embedded errors. The quality of the structure of the resulting scenarios was also assessed, including their clarity, modularity, and suitability for integration into CI/CD pipelines. The acceptability criteria were defined by the following threshold values: code coverage of no less than 80%, a proportion of correct tests of at least 85%, and the ability to detect no fewer than 70% of control defects.

Results

Analysis of test case generation using different prompt types

Empirical results demonstrated that different prompt construction approaches substantially influence the number of generated tests, their completeness, and their compliance with functional requirements. In the zero-shot case, the models produced an average of 60-70% of the expected number of tests, while the proportion of relevant tests reached approximately 75%. The few-shot approach significantly improved these indicators: the models generated an almost complete set of tests, and relevance reached 88-92%, which indicated the benefit of providing examples for model guidance. Chain-of-thought prompting enabled the generation of more detailed and logically structured test scenarios, particularly for modules with complex business logic, although the overall number of tests was sometimes lower than under the few-shot approach. Role prompting demonstrated a high level of relevance, exceeding 90%, and a high degree of test structure (Table 1).

Table 1. Comparison of test case generation by prompt type

Prompt type	Number of tests, % of expected	Relevance, %	Generation features	Result analysis
Zero-shot	60-70	75	Rapid generation, occasionally misses complex scenarios	Suitable for basic modules; insufficient coverage of complex functions; time-saving in prompt formulation
Few-shot	85-100	88-92	More complete and accurate tests due to examples	Best balance between test quantity and quality; provides high relevance and coverage
Chain-of-thought	80-95	85-90	Detailed logical sequences, particularly for complex functions	Produces structured tests for complex logic, though may generate fewer tests; requires longer processing time
Role prompting	80-95	90-95	High structural coherence and human-like reasoning	Recommended for complex scenarios; high relevance and readability of tests; requires correct role specification

Source: compiled by the author

As shown in the table, few-shot and role prompting demonstrate the highest effectiveness in the generation of test scenarios, ensuring both completeness and relevance. Few-shot is suitable for a wide range of modules due to the presence of examples that orient the model. Role prompting proves particularly effective in the case of complex or multi-level functions, where test structure and "human" logical reasoning are critical. Zero-shot is suitable for the rapid generation of basic tests, yet it demonstrates lower relevance and coverage, especially for complex modules. Chain-of-thought is appropriate for in-depth logical analysis of code and enables the generation of more detailed scenarios, yet it sometimes produces a smaller number of tests and requires extra processing time. Hence, the choice of prompt engineering technique should depend on module complexity and test coverage requirements, which enables optimisation of the test generation process and enhancement of QA process effectiveness. The entire process, from prompt selection to result evaluation, is presented in Figure 1. The diagram illustrates how different prompt engineering strategies influence the number, structure, and relevance of tests, together with their integration into the development environment and CI/CD.

Figure 1. Test scenario generation from prompt selection to result evaluation

Source: compiled by the author

Analysis of the flowchart indicates that test scenario generation is clearly dependent on the selected prompt type and the logic of its formulation. The prompt engineering strategy at the initial stage determines not only the number and relevance of tests but also their structural properties and complexity. The next stage involves processing the prompt by LLM, which generates a set of test scenarios covering different aspects of the code, ranging from basic unit tests to tests for exceptional situations and complex module logic. Integration into the development environment and CI/CD ensures the practical applicability of the tests, while their execution enables the collection of metrics related to code coverage, relevance, structure, and defect detection effectiveness. The diagram demonstrates that test generation effectiveness increases under more detailed and structured prompt approaches, for example, few-shot or role prompting, whereas simpler strategies, such as zero-shot, ensure speed but provide lower relevance and coverage. The visualisation confirms that the correct sequence of stages is critical for achieving high quality in automated tests.

Code coverage and test relevance

The analysis results demonstrated substantial variability depending on the applied prompt type. Tests generated in the zero-shot mode showed the lowest indicators. They covered an average of only 55-65% of functions, while many complex scenarios remained unaddressed. The accuracy level in this case reached 70-75%, which indicated partial correctness of the obtained results: simple functions were tested satisfactorily, yet exceptional situations were often ignored. In contrast, few-shot prompting demonstrated the best performance. Function coverage reached 85-95%, and test accuracy consistently remained within 88-92%. The inclusion of examples in prompts enabled the model to generate more structured and comprehensive tests that adequately reflected the behaviour of program code, even in complex cases. This approach proved to be the most balanced in terms of the quantity and quality of tests. Chain-of-thought prompting ensured coverage at the level of 80-90% and accuracy of 85-90%. It performed particularly well in cases of complex algorithmic blocks that required step-by-step verification of logic. At the same time, the emphasis on detail led to the omission of some simple functions, which resulted in slightly lower overall completeness than in the few-shot approach. Role prompting produced results close to those of a few-shot. Function coverage ranged from 82-93%, while test accuracy reached 90-95%. The main distinction of this approach lies in the more "human-like" structure of the tests: they imitate the practice of an experienced QA engineer, pay attention to edge cases, and ensure high relevance. This made role prompting useful for practical application in real testing teams (Table 2).

Table 2. Code coverage and test relevance by prompt type

Prompt type	Function coverage, %	Test accuracy, %	Notes
Zero-shot	55-65	70-75	Partially covers basic functions; complex scenarios are frequently missed
Few-shot	85-95	88-92	Provides the highest coverage and relevance; examples in prompts help generate accurate tests
Chain-of-thought	80-90	85-90	Well-suited for complex logic; may miss some simple functions due to focus on detail
Role prompting	82-93	90-95	Structured and human-like tests; high relevance, particularly for complex scenarios

Source: compiled by the author

Analysis of the table confirms that the highest quality indicators are ensured by few-shot and role prompting. Their results are nearly identical, although the former relies more strongly on examples, while the latter depends on role context, which shapes more meaningful scenarios. Chain-of-thought is optimal for in-depth analysis of complex code segments, yet it does not guarantee full coverage. Zero-shot, despite its speed and simplicity of application, demonstrated limited results and may be regarded only as an auxiliary tool for the initial development of test sets. Hence, the results show that a comprehensive evaluation based on coverage and relevance indicators enables a clear identification of the strengths and weaknesses of different prompt engineering strategies. In practical application, the most effective solution involves the use of few-shot and role prompting, while other approaches should be combined depending on objectives and the nature of the software product.

Defect detection and test effectiveness

The experimental results indicate that different approaches to test case generation demonstrate substantial differences in the ability to detect embedded errors. Analysis of the ratio of detected defects to their total number enabled evaluation of both the effectiveness of each approach and the relevance of tests to the assigned tasks (Table 3). For a clear presentation of the effectiveness of different prompt construction strategies in detecting embedded defects, a diagram is provided that shows the percentage of detected errors for each method (Fig. 2).

Table 3. Ratio of detected defects to injected defects

Prompt type	Number of injected defects	Detected defects
Zero-shot prompting	20	9
Few-shot prompting	20	14
Chain-of-thought	20	16
Role prompting	20	13

Source: compiled by the author

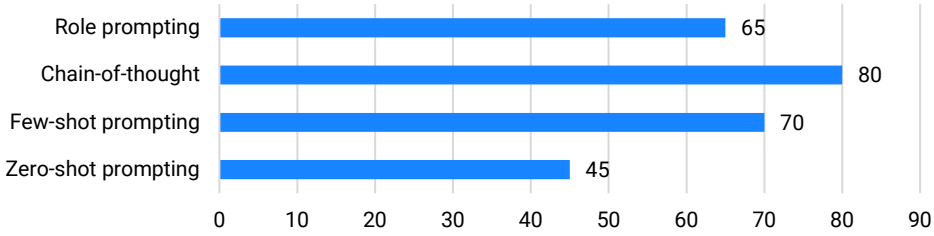

Figure 2. Effectiveness of prompting methods in defect detection
Source: compiled by the author

The analysis indicates that the highest effectiveness was achieved by the chain-of-thought approach, which enabled the detection of 16 out of 20 embedded errors, or 80%. This confirms the capacity of multi-step reasoning to improve the logical justification of checks and coverage of different scenarios. Few-shot prompting ranked second at 70%, which demonstrates the benefit of learning from examples. Zero-shot prompting proved to be the least effective at 45% because, in the absence of context, the model frequently misses critical defects. Role prompting showed a moderate result at 65%, which indicates that although role specification helps to structure testing, it is less effective without examples or multi-step logic. The main types of defects most frequently detected by each prompting method are presented in Figure 3.

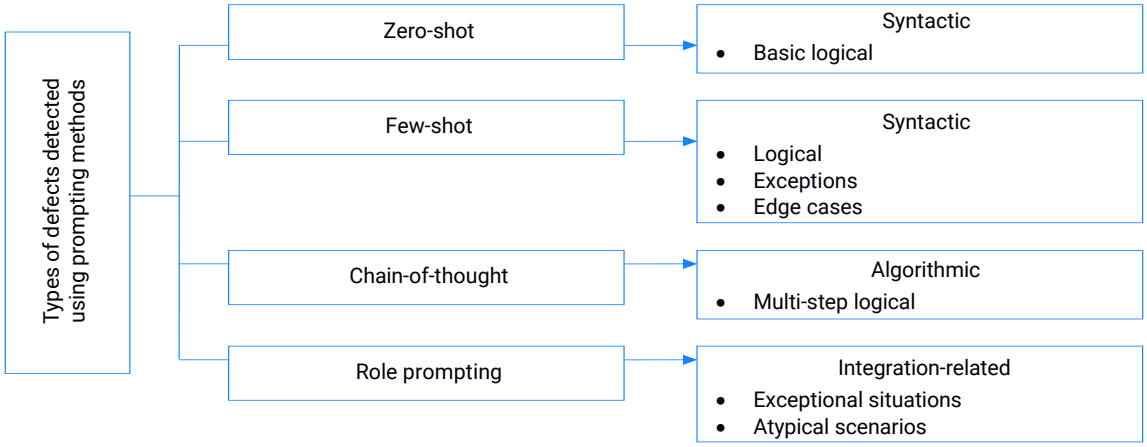

Figure 3. Types of defects most frequently detected by different prompting methods
Source: compiled by the author

The scheme demonstrates a clear specialisation of each prompting method in detecting different types of defects, which enables an evaluation of the strengths and weaknesses of the approaches in practical testing. Zero-shot was limited to only basic categories of defects, namely syntactic and simple logical errors. This indicates that, in the absence of examples or supplementary guidance, the model is capable of rapidly covering basic functionality yet overlooks complex scenarios, such as edge cases or multi-step logical errors. Few-shot demonstrated the widest range of detected defects. The model identified not only syntactic and basic logical errors but also more complex categories, including exceptions and edge cases. This highlights the benefit of including examples in prompts, which orient the model towards specific scenarios and support more precise identification of problematic areas of code. This approach increases both test relevance and the capacity to detect critical defects. Chain-of-thought specialised in algorithmic and multi-step logical errors. Through step-by-step reasoning, the model was able to identify complex interrelations between code components that are difficult to verify by simple prompts. At the same time, this method proved less effective in detecting basic syntactic errors, since the primary focus was placed on complex logic. Role prompting was distinguished by its ability to detect integration defects and errors arising in atypical scenarios or during module interaction. This approach models the behaviour of a user or an experienced QA engineer, which enables tests to reproduce real operating conditions of software systems. However, role prompting is less effective in covering simple syntactic or basic logical errors unless it is complemented by examples or multi-step logic. In general, analysis of the scheme confirms the appropriateness of a combined application of methods: few-shot and chain-of-thought ensure broad and deep defect coverage, whereas role prompting adds contextual verification of integration and atypical scenarios, and zero-shot may be applied for rapid generation of basic tests. This approach enables an increase in the overall effectiveness of testing while minimising omissions in the detection of critical defects.

Evaluation of structural quality and integration suitability of tests

The study demonstrated how different approaches influence the practical readiness of tests, enabling evaluation of their reusability, maintainability, and integration into the development process. The subsequent analysis examined these indicators in detail, identified the advantages and limitations of each approach, and confirmed the feasibility of combining methods to achieve optimal test quality. Figure 4 presents the key indicators for each prompting method: test readability, modularity, and suitability for automated execution in CI/CD pipelines.

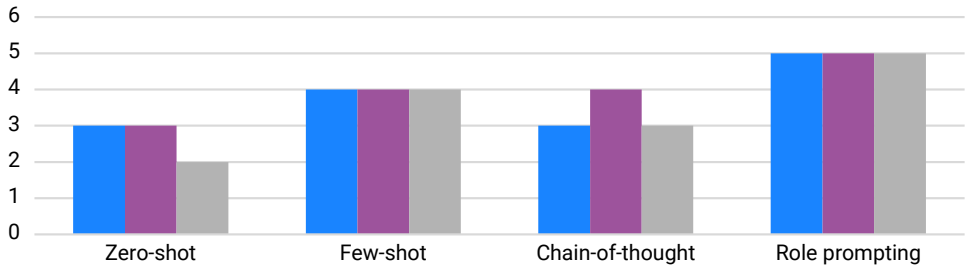

Figure 4. Key performance indicators for each prompting method
Note: blue – test readability, orange – modularity, grey – suitability for automated execution in CI/CD pipelines
Source: compiled by the author

The evaluation results indicate substantial differences in structural quality and integration suitability of tests depending on the prompt type. Zero-shot generated basic tests that exhibited average readability and modularity, yet low integration suitability. This is explained by the fact that test cases were often repetitive or not optimally decomposed into modules, which complicated their automated execution in CI/CD pipelines. Few-shot demonstrated high readability and modularity, since the presence of examples in prompts enabled the model to form structured and logically complete test cases. Integration suitability was also high, although certain difficulties occasionally arose due to additional dependencies that required environment configuration or data reuse. Chain-of-thought produced tests with a high level of detail and strong modularity, which supported their reuse for verification of different components. At the same time, readability sometimes suffered because of excessive detail, while integration suitability remained moderate, as tests sometimes required complex dependencies that complicated their execution in automated pipelines. Role prompting ensured the highest evaluation across all categories: tests were maximally readable, logically structured, and modular. Through the development of scenarios approximating the behaviour of an experienced QA engineer, they were easily integrated into CI/CD pipelines and automated execution. This approach demonstrated that role-based prompting enables the model to form tests that are immediately ready for practical application in development and testing processes. In general, analysis of the table confirms that the combination of few-shot and role prompting ensures an optimal balance between readability, modularity, and integration suitability of tests, whereas zero-shot and chain-of-thought exhibit specific limitations that should be considered during implementation in CI/CD processes. Figure 5 illustrates the relationship between the structural quality of tests and their integration suitability.

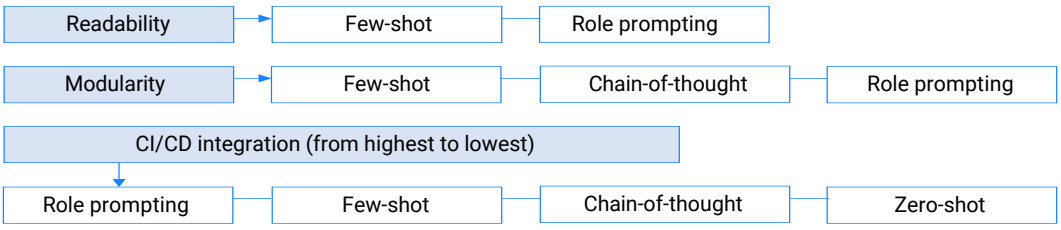

Figure 5. Dependence of test structure on the type of prompting and integration suitability
Note: compiled by the author

The scheme demonstrates that different prompting methods differed across three key indicators of test quality: readability, modularity, and integration suitability within CI/CD. Readability was highest in few-shot and role prompting, since the presence of examples or a role-based context supported the development of logically structured and comprehensible test cases. Modularity was high in few-shot, chain-of-thought, and role prompting, which enabled tests to be reused for verification of different components without substantial code modification. With regard to integration suitability, the best results were demonstrated by role prompting, since tests were developed in close alignment with the human logic of a QA engineer and were easily integrated into CI/CD pipelines. Few-shot ensured good integration, although it occasionally required additional configuration due to dependencies on examples

embedded in the prompts. Chain-of-thought showed moderate integration suitability as a result of complex logical dependencies, whereas zero-shot demonstrated the lowest level, since tests were basic and less structured. Thus, the combination of few-shot and role prompting enabled the achievement of an optimal balance between readability, modularity, and readiness for integration into CI/CD processes, whereas zero-shot and chain-of-thought exhibited specific advantages for rapid generation of basic tests or for in-depth verification of logic, respectively.

Comparison of the performance of the CodeLlama 2 and StarCoder models

During the experiments, the models demonstrated different approaches to test generation depending on the complexity of the scenario (Table 4). StarCoder is characterised by a higher speed of test creation, which enables the rapid production of basic functional tests. However, in cases where edge or exceptional scenarios require coverage, its outputs occasionally require additional verification and refinement, since the structure of tests could vary across repeated generations. CodeLlama 2 ensures more stable and predictable test generation. Its results are characterised by high consistency across repeated queries and by detailed verification of functional logic. Through this characteristic, unit tests are more structurally organised and cover both basic and complex scenarios, including exceptional situations and integration scenarios between modules. This feature renders CodeLlama 2 more suitable for comprehensive testing and for automated verification of software systems.

Table 4. Performance of the CodeLlama 2 and StarCoder models

Model	Test generation time (mean, s)	Response stability (%)	Number of generated tests	Characteristics of the results
CodeLlama 2	16.2	90	52	High stability, well-structured tests, effective coverage of edge cases and exceptions
StarCoder	14.7	87	50	Fast generation, minor variations in test structure, slightly lower coverage of complex scenarios

Source: compiled by the author

The table demonstrates that each model possesses specific advantages and shortcomings. StarCoder enables the rapid generation of basic test scenarios, which are useful for quick verification of module functionality. However, the lower stability of the results limits its use for critically important testing of complex logic. CodeLlama 2 is characterised by high stability and the ability to cover various types of scenarios, including boundary and exceptional cases. This increases the relevance of the tests and makes them more suitable for continuous integration and automated execution. The number of generated tests is comparable between the models, but CodeLlama 2 ensures more comprehensive coverage of the program logic and defect detection.

In addition, the repeatability of results under multiple generations of tests for the same module was analysed. CodeLlama 2 demonstrated stability of approximately 90%, whereas StarCoder reached 87%. The difference in stability was especially evident in complex scenarios with multi-step logic and integration dependencies between functions. These findings indicate the feasibility of combining the models in practical use: StarCoder was applied for the rapid generation of basic tests, whereas CodeLlama 2 ensures stability and comprehensive logic coverage. Overall, the results show that the combination of the models ensures an optimal balance between the speed of test generation, their stability, and the ability to cover diverse usage scenarios. This approach increases the effectiveness of automated testing and makes the results more suitable for integration into modern software development processes.

Discussion

The findings show that the few-shot, chain-of-thought, and role prompting strategies ensure the highest effectiveness in automated test creation. CodeLlama 2 exhibits higher stability at 90% and the ability to operate with complex scenarios, whereas StarCoder ensures higher generation speed but slightly lower predictability of results. In the group of studies devoted to prompt engineering methods, S. Vatsal & H. Dubey (2024) conducted a review and systematised prompt engineering techniques for various natural language processing tasks, including code generation and automated testing. The researchers emphasised the advantage of few-shot and multi-step prompts over zero-shot approaches, since they ensured higher relevance, response consistency, and more complete coverage of functional scenarios. These conclusions correlate with the obtained results: few-shot achieved 85-100% coverage and 88-92% relevance, whereas zero-shot achieved only 55-65% coverage and 70-75% accuracy. Thus, the conducted study confirms a general tendency that the presence of examples in prompts is critically important for the generation of accurate and relevant tests. The study by C.Y. Wang (2025) focused on the optimisation of different prompt types for code generation and showed that role-based prompts and prompts with step-by-step reasoning yield structured tests and effectively cover complex scenarios. The obtained results confirm these conclusions: role prompting ensured 82-93% coverage and 90-95% accuracy, whereas chain-of-thought detected 16 out of 20 defects at 80%, which demonstrates

the effectiveness of step-by-step reasoning for complex logic and integration scenarios.

J. Wang *et al.* (2024) and M. Schäfer *et al.* (2023) addressed the empirical evaluation of automated testing with large language models. Both studies confirm that multi-step and example-based approaches ensure high performance in unit test generation, especially for complex scenarios. The present study similarly showed that StarCoder rapidly generates basic tests, with an average time of 14.7 seconds and stability of 87%, whereas CodeLlama 2 ensures greater stability and detailed coverage of complex scenarios, with an average time of 16.2 seconds, stability of 90%, and 52 test cases. The difference in stability between the models is explained by variations in experimental conditions, including project size and programming languages, and does not contradict general trends in the literature. In turn, L. Belzner *et al.* (2023) examined the integration of large language models into real projects, pointing to the potential for improving development effectiveness and test automation, while also emphasising the complexity of adapting models to specific scenarios. The present results confirm these conclusions: the use of different prompt types enables the adaptation of test scenarios to code specifics and ensures an optimal balance between generation speed and test relevance. For example, chain-of-thought achieved 80% defect detection, whereas few-shot achieved 70%, which demonstrates the effectiveness of multi-step and example-based prompts in complex scenarios.

Within the prompt engineering domain, B. Chen *et al.* (2025) conducted a review of prompt design methods and established that the use of examples and role-based prompts significantly increases the accuracy and structural coherence of large language model outputs. This correlates with the obtained findings. The researchers also noted that prompts which place the model in the role of an expert enhance the quality of test integration into real workflows, which aligns with the present conclusion regarding the high integration suitability of role prompting in CI/CD processes. H. Strobelt *et al.* (2022) examined interactive and visual prompt design methods for adapting models to specific tasks and emphasised the role of user intervention in model configuration. This partially explains why role prompting in the present study ensured the highest readability and structural coherence of tests: the prompts effectively model the behaviour of an experienced QA engineer, which allows the automatic generation of tests ready for integration into real processes.

Another group of studies focused on the fundamental principles of prompt engineering for large language models. In particular, A. Gao (2023) demonstrated that proper query formulation considerably affects the accuracy and relevance of responses, along with model consistency in the performance of complex tasks. Conceptually, this approach is similar to the one proposed in the current study in terms of the staged configuration of system behaviour under specific conditions. However, A. Gao did not address resource management or the integration of resilience mechanisms, which represent key elements in the present study. Meanwhile, S. Feng & C. Chen (2024) showed that the use of prompts for the automatic reproduction of bugs on the Android platform allows a substantial reduction in manual testing time and improves coverage of edge cases. This correlates with the obtained results for chain-of-thought prompting, which ensured 80% defect detection and high structural coherence of tests, particularly in complex scenarios.

In the study by C. Pornprasit & C. Tantithamthavorn (2024), the combination of fine-tuning and prompt engineering was shown to improve the effectiveness of automated code review. The researchers developed methods for adapting models to specific syntax and structural features of code. Similar to the present staged approach (timeouts → repeated calls → sidecar), the study demonstrated the benefits of a systematic, incremental introduction of improvements. However, the researchers did not take into account the physical constraints of the system, which were considered in the current model through the monitoring of the CPU, memory, and latency. B. Clavié *et al.* (2023) focused on prompt engineering for task classification in a corporate environment. Their results demonstrated that role-based prompts considerably increase the accuracy and structural organisation of outputs. This supports the present conclusions regarding role prompting, which ensures the highest readability and integration suitability of tests, along with 90-95% accuracy. T. Radcliffe *et al.* (2024) investigated prompt automation for the detection of semantic vulnerabilities in large language models. The researchers emphasised that, without proper prompt configuration, even powerful models may miss critical errors. This explains why zero-shot in the present study ensured only 45% defect detection and low integration suitability: the absence of context and examples limits model effectiveness in complex scenarios.

In the study by W. Cain (2024), educational aspects of prompt engineering were examined, demonstrating that proper query configuration can significantly improve learning effectiveness and reduce the risk of incorrect responses. The researcher emphasised the importance of a multi-level approach and systematic testing. Conceptually, the common emphasis on incremental optimisation is obvious, but W. Cain focused on the cognitive and educational aspects of large language models, while in the current study resource and system management interface was applied to ensure resilience and load balancing in real microservice systems. The study by A. Fan *et al.* (2023) presented a review of the current state of large language models in software engineering, particularly for test automation, code generation, and bug fixing. The researchers noted that combined approaches based on multi-step and role-based prompts ensure the highest accuracy and structural coherence of outputs. This fully correlates with the obtained data, in which chain-of-thought and role prompting demonstrated the best coverage and relevance of tests

at 80-95%. L. Plein *et al.* (2024) showed that large language models can effectively generate test scenarios based on bug reports, which enables the automation of software verification processes. The present study confirms this: few-shot and chain-of-thought approaches ensured 70-80% defect detection with high test structural coherence, particularly in complex and integration scenarios.

N. Alshahwan *et al.* (2024) investigated automated improvement of unit tests in a large organisation (Meta) using large language models. They established that models provided with detailed examples and context generated more accurate and reproducible tests, which aligns with the obtained results: CodeLlama 2 demonstrated 90% stability and high predictability of tests, whereas StarCoder was faster but less stable. The difference in speed and stability also explains the necessity of combined model usage in practical scenarios. J. Velásquez-Henao *et al.* (2023) proposed a methodology for optimising interactions with large language models in engineering tasks through structured prompt configuration and regular verification of results. The researchers demonstrated that this approach enhances model accuracy and allows systematic control of performance. Multi-level optimisation is similar to the staged implementation of resilience mechanisms in the current model. The distinction lies in the application domain: J. Velásquez-Henao *et al.* focused on algorithmic-level engineering tasks, whereas current research addressed physical and logical aspects of microservice systems. D. Grabb (2023) demonstrated that prompt engineering can significantly improve large language model performance in specific medical scenarios, particularly under high risk of errors. The study showed that systematic query optimisation reduces the likelihood of incorrect outputs and increases model stability. His results correlate with the present methodology in terms of systematic adaptation and enhanced stability. The presented model, however, additionally accounted for hardware resources, service degradation, and request balancing in distributed systems, which were not investigated in the cited study.

A. Nayyar *et al*. (2025) systematically reviewed strategies for prompt optimisation in large language models, including role prompting, few-shot methods, and integrated scenario approaches, which confirms the present observations regarding the appropriateness of combining approaches to achieve high test accuracy and structural coherence. The researcher also noted that project-specific conditions and codebase sizes can affect model performance, which explains partial discrepancies in coverage percentages between the present experiments and other studies. E. Jiang *et al.* (2022) developed PromptMaker, a system for prototype testing based on prompts, which enables interactive evaluation and configuration of prompt strategies. The study emphasised the importance of adapting prompts to task-specific requirements and supports the present results: role prompting ensured high readability, structural coherence, and integration suitability of tests, particularly for complex scenarios.

Comparison of the present study with previous findings indicates general alignment regarding the effectiveness of prompt strategies for large language models. Few-shot and chain-of-thought strategies provide the highest accuracy and coverage of test scenarios, whereas role prompting improves structural coherence, readability, and integration suitability of tests. The findings also demonstrate slightly higher stability and coverage for CodeLlama 2 compared with some publications, which may be explained by differences in project selection, programming languages, and model configurations. Overall, the study confirms key trends in automated test generation using large language models while providing practical evaluation of specific models' performance in real-world conditions, making it a valuable addition to existing findings.

Conclusions

The empirical analysis demonstrated that the effectiveness of test scenario generation depends directly on the type of prompt employed. Zero-shot produced only 60-70% of the expected number of tests with relevance of approximately 75%, rendering it suitable only for the rapid generation of basic checks. Few-shot proved to be the most balanced approach, providing 85-100% coverage and relevance of 88-92%. Chain-of-thought enabled 80-95% of tests with relevance of 85-90%, producing more logically structured scenarios, particularly for complex functions, although it occasionally lagged behind few-shot in quantity. Role prompting exhibited the highest quality, achieving 80-95% coverage and 90-95% relevance, ensuring structural coherence and human-like reasoning in the tests. Consequently, the most effective strategies are few-shot and role prompting, whereas zero-shot and chain-of-thought remain useful for narrow application cases.

The analysis indicated that zero-shot delivered the lowest results, with coverage of only 55-65% and accuracy of 70-75%, making it suitable solely for basic checks. Few-shot demonstrated the best performance with 85-95% coverage and 88-92% accuracy, reflecting a balance between test quantity and quality. Chain-of-thought achieved 80-90% coverage and 85-90% accuracy, effective for complex logic but less comprehensive for simple functions. Role prompting provided 82-93% coverage and 90-95% accuracy, generating structured and human-like tests, particularly useful for complex scenarios. Overall, few-shot and role prompting are the most effective, whereas chain-of-thought and zero-shot are best applied in specific cases. Chain-of-thought prompting proved the most efficient in defect detection, identifying 16 out of 20 defects (80%), confirming the advantage of stepwise reasoning for complex logic. Few-shot ranked second with 14 defects (70%), demonstrating the benefit of examples for increasing test relevance. Role prompting identified 13 defects (65%), performing well in integration and atypical scenarios, while zero-shot detected only nine defects (45%), limited to basic errors. An optimal balance is achieved through the combined use of few-shot and chain-of-thought for covering

critical logical errors, with role prompting adding realistic verification and zero-shot applicable for rapid generation of simple tests. The study showed that role prompting produced the highest readability and integration suitability, with tests that were logically structured, modular, and readily incorporated into CI/CD pipelines. Few-shot ensured high readability and modularity, with slightly lower integration readiness. Chain-of-thought generated detailed and reusable tests, although integration suitability was moderate due to complex dependencies, while zero-shot produced basic tests with moderate readability, low modularity, and the lowest integration suitability.

CodeLlama 2 provided more stable and predictable test generation, with high consistency on repeated queries (90%), an average generation time of 16.2 seconds, and 52 generated tests, covering both basic and complex scenarios, including edge cases and exceptions. StarCoder was faster (14.7 seconds), generating 50 tests with slightly lower stability (87%) and less coverage of complex scenarios, making it useful for rapid verification of basic functionality. Combined model usage optimises the balance between generation speed, stability, and coverage completeness, enhancing the effectiveness of automated testing and integration into CI/CD.

Limitations of the study included the use of only two language models and a relatively small sample of medium-sized software projects developed in Python and Java, which partially constrains generalisation of results to other technology stacks. The study did not consider the influence of specific architectural features, such as microservice or event-driven structures, which may significantly affect the effectiveness of automated test generation. Future research should extend the analysis to additional models, diverse programming languages, and complex enterprise systems with integration into different CI/CD environments.

Acknowledgements
None.

Funding
The study was not funded.

Conflict of Interest
None.

References

[1] Adu, G. (2024). *Artificial Intelligence in software testing: Test scenario and case generation with an AI model (gpt-3.5-turbo) using prompt engineering, fine-tuning and retrieval augmented generation techniques*. (Master's Thesis, University of Eastern, Joensuu, Finland).

[2] Alagarsamy, S., Tantithamthavorn, C., Takerngsaksiri, W., Arora, C., & Aleti, A. (2025). Enhancing large language models for text-to-testcase generation. *Journal of Systems and Software*, 230, article number 112531. doi: 10.1016/j.jss.2025.112531.

[3] Alshahwan, N., Chheda, J., Finogenova, A., Gokkaya, B., Harman, M., Harper, I., Marginean, A., Sengupta, S., & Wang,E. (2024). Automated unit test improvement using large language models at Meta. In M. d'Amorim (Ed.), *Companion proceedings of the 32nd ACM international conference on the foundations of software engineering* (pp. 185-196). New York: Association for Computing Machinery. doi: 10.1145/3663529.3663839.

[4] Anasuri, S. (2024). Prompt engineering best practices for code generation tools. *International Journal of Emerging Trends in Computer Science and Information Technology*, 5(1), 69-81. doi: 10.63282/3050-9246.IJETCSIT-V5I1P108.

[5] Belzner, L., Gabor, T., & Wirsing, M. (2023). Large language model assisted software engineering: Prospects, challenges, and a case study. In B. Steffen (Ed.), *Bridging the gap between AI and reality* (pp. 355-374). Cham: Springer. doi: 10.1007/978-3-031-46002-9_23.

[6] Cain, W. (2024). Prompting change: Exploring prompt engineering in large language model AI and its potential to transform education. *TechTrends*, 68(1), 47-57. doi: 10.1007/s11528-023-00896-0.

[7] Chen, B., Zhang, Z., Langrené, N., & Zhu, S. (2025). Unleashing the potential of prompt engineering for large language models. *Patterns*, 6(6), article number 101260. doi: 10.1016/j.patter.2025.101260.

[8] Clavié, B., Ciceu, A., Naylor, F., Soulié, G., & Brightwell, T. (2023). Large language models in the workplace: A case study on prompt engineering for job type classification. In E. Métais, F. Meziane, V. Sugumaran, W. Manning & S. Reiff-Marganiec (Eds.), *Natural language processing and information systems* (pp. 3-17). Cham: Springer. doi: 10.1007/978-3-031-35320-8_1.

[9] Fan, A., Gokkaya, B., Harman, M., Lyubarskiy, M., Sengupta, S., Yoo, S., & Zhang, J.M. (2023). Large language models for software engineering: Survey and open problems. In *Proceedings of the IEEE/ACM international conference on software engineering: Future of software engineering* (pp. 31-53). Melbourne: IEEE. doi: 10.1109/ICSE-FoSE59343.2023.00008.

[10] Feng, S., & Chen, C. (2024). Prompting is all you need: Automated android bug replay with large language models. In A. Paiva & R. Abreu (Eds.), *Proceedings of the 46th IEEE/ACM international conference on software engineering* (article number 67). New York: Association for Computing Machinery. doi: 10.1145/3597503.3608137.

[11] Gao, A. (2023). Prompt engineering for large language models. *SSRN*. doi: 10.2139/ssrn.4504303.

[12] Grabb, D. (2023). The impact of prompt engineering in large language model performance: A psychiatric example. *Journal of Medical Artificial Intelligence*, 6, article number 20. doi: 10.21037/jmai-23-71.

[13] Jiang, E., Olson, K., Toh, E., Molina, A., Donsbach, A., Terry, M., & Cai, C.J. (2022). PromptMaker: Prompt-based prototyping with large language models. In S. Barbosa, C. Lampe, C. Appert & D.A. Shamma (Eds.), *CHI Conference on human factors in computing systems extended abstracts* (article number 35). New York: Association for Computing Machinery. doi: 10.1145/3491101.3503564.

[14] Levitskyi, S., & Mokin, V. (2025). *Analysis of benchmark tests of large language models' resilience to disinformation and various types of manipulation*. Retrieved from http://ir.lib.vntu.edu.ua/handle/123456789/49249.

[15] Lim, S., & Schmälzle, R. (2023). Artificial intelligence for health message generation: An empirical study using a large language model (LLM) and prompt engineers. *Frontiers in Communication*, 8, article number 1129082. doi: 10.3389/fcomm.2023.1129082.

[16] Naimi, L., Manaouch, M., & Jakimi, A. (2024). A new approach for automatic test case generation from use case diagram using LLMs and prompt engineering. In *Proceedings of the international conference on circuit, systems and communication* (pp. 1-5). Fes: IEEE. doi: 10.1109/ICCSC62074.2024.10616548.

[17] Nayyar, A., Vairamani, A.D., & Kaswan, K. (2025). *Mastering prompt engineering: Deep insights for optimizing large language models (LLMs)*. London: Elsevier. doi: 10.1016/C2024-0-00708-4.

[18] Novakovsky, A., & Yalovega, I. (2025). Categorisation of the capabilities of large language models of artificial intelligence. In *Proceedings of the 29th international youth forum "Radio electronics and youth in the 21st century"* (pp. 296-298). Kharkiv: Kharkiv National University of Radio Electronics.

[19] Plein, L., Ouédraogo, W.C., Klein, J., & Bissyandé, T.F. (2024). Automatic generation of test cases based on bug reports: A feasibility study with large language models. In A. Paiva & R. Abreu (Eds.), *Proceedings of the 2024 IEEE/ACM 46th international conference on software engineering: Companion proceedings* (pp. 360-361). New York: Association for Computing Machinery. doi: 10.1145/3639478.3643119.

[20] Pornprasit, C., & Tantithamthavorn, C. (2024). Fine-tuning and prompt engineering for large language models-based code review automation. *Information and Software Technology*, 175, article number 107523. doi: 10.1016/j.infsof.2024.107523.

[21] Radcliffe, T., Lockhart, E., & Wetherington, J. (2024). Automated prompt engineering for semantic vulnerabilities in large language models. *Authorea*. doi: 10.22541/au.172348895.52207804/v1.

[22] Sahoo, P., Singh, A.K., Saha, S., Jain, V., Mondal, S., & Chadha, A. (2024). A systematic survey of prompt engineering in large language models: Techniques and applications. *ArXiv*. doi: 10.48550/arXiv.2402.07927.

[23] Schäfer, M., Nadi, S., Eghbali, A., & Tip, F. (2023). An empirical evaluation of using large language models for automated unit test generation. *IEEE Transactions on Software Engineering*, 50(1), 85-105. doi: 10.1109/TSE.2023.3334955.

[24] Strobelt, H., Webson, A., Sanh, V., Hoover, B., Beyer, J., Pfister, H., & Rush, A.M. (2022). Interactive and visual prompt engineering for ad-hoc task adaptation with large language models. *IEEE Transactions on Visualization and Computer Graphics*, 29(1), 1146-1156. doi: 10.1109/TVCG.2022.3209479.

[25] Vatsal, S., & Dubey, H. (2024). A survey of prompt engineering methods in large language models for different NLP tasks. *ArXiv*. doi: 10.48550/arXiv.2407.12994.

[26] Velásquez-Henao, J.D., Franco-Cardona, C.J., & Cadavid-Higuita, L. (2023). Prompt engineering: A methodology for optimizing interactions with AI-Language models in the field of engineering. *Dyna*, 90, 9-17.

[27] Wang, C.Y. (2025). *Application and optimization of prompt engineering techniques for code generation in large language models*. (Master's thesis, York University, Toronto, Canada).

[28] Wang, J., Huang, Y., Chen, C., Liu, Z., Wang, S., & Wang, Q. (2024). Software testing with large language models: Survey, landscape, and vision. *IEEE Transactions on Software Engineering*, 50(4), 911-936. doi: 10.1109/TSE.2024.3368208.

[29] Yurchak, I., Kychuk, O., Oksentyuk, V., & Khich, A. (2024). Prompting techniques for enhancing the use of large language models. *Computer Systems and Networks*, 6(2), 286-300. doi: 10.23939/csn2024.02.268.

Швидка розробка великих мовних моделей для генерації тестових випадків

Анатолій Гусаковський

Магістр, старший інженер
Національний аерокосмічний університет «Харківський авіаційний інститут»
61070, вул. В. Манька, 17, м. Харків, Україна
https://orcid.org/0009-0007-9398-0966

Анотація. Актуальність дослідження зумовлена потребою підвищення ефективності тестування програмного забезпечення, де використання великих мовних моделей і технік інженерії підказок відкриває нові можливості для автоматизованої генерації якісних тестових випадків. Метою дослідження було оцінити ефективність стратегій prompt engineering у генерації тестових випадків великими мовними моделями. Методологія базувалася на порівнянні чотирьох технік prompt engineering: zero-shot, few-shot, chain-of-thought та role prompting для генерації unit-тестів мовними моделями CodeLlama 2 та StarCoder у середовищі PyTest і JUnit із оцінкою за критеріями покриття коду, релевантності, дефектовиявлення та інтеграційної придатності. Аналіз показав, що few-shot та role prompting забезпечують найкращий баланс між кількістю та якістю тестів із покриттям 85-100 % та релевантністю 88-95 %, тоді як chain-of-thought ефективний для складної логіки й виявив 16 із 20 закладених дефектів (80 %), а zero-shot обмежений базовими перевірками з покриттям 55-65 % та точністю 70-75 %. CodeLlama 2 продемонстрував стабільну генерацію тестів із високою узгодженістю повторних запитів (90 %), середнім часом генерації 16,2 с та 52 тестами на модуль, охоплюючи базові та складні сценарії, включно з крайовими випадками та винятками. StarCoder був швидшим (14,7 с), генерував 50 тестів із трохи нижчою стабільністю (87 %) і меншим покриттям складних сценаріїв, що робило його ефективним для швидкої перевірки базових функцій. Найвища читабельність, модульність і інтеграційна придатність у CI/CD-конвеєри були за role prompting, тоді як few-shot забезпечував гарний баланс між структурованістю та практичною готовністю тестів, а chain-of-thought і zero-shot мали специфічні обмеження. Комбіноване використання моделей і стратегій prompting дозволяє оптимізувати процес генерації тестів, підвищуючи їхню релевантність, покриття та ефективність автоматизованого тестування. Результати дослідження можуть застосовуватися для автоматизованого тестування програмного забезпечення, інтеграції у конвеєри безперервної інтеграції та доставки та навчання інженерів з контролю якості ефективним методам генерації тестів

Ключові слова: CodeLlama; StarCoder; підказки з нульовим результатом; підказки з кількома результатами; підказки ланцюжка думок; підказки ролей; інтеграція CI/CD

Journal homepage: https://itce.vn.ua/en

Information Technologies and Computer Engineering **Vol. 23, No. 1. 2026**

Article's History: Received: 14.11.2025 Revised: 16.02.2026
Accepted: 26.03.2026 Published: 20.04.2026

ISSN 1999-9941; e-ISSN 2078-6387
UDC 004.056.5
DOI: 10.31649/vitce/1.2026.35

Interactive visualisation and analysis of risks with a human factor

Viktoriia Trofymchuk*
Master, Lecturer
State University "Kyiv Aviation Institute"
03058, 1 Liubomyr Huzar Ave., Kyiv, Ukraine
https://orcid.org/0000-0002-9756-0244

Abstract. The human factor remains one of the key vulnerabilities in modern cybersecurity, which emphasises the importance of analysing user behaviour in risk management systems. This study presents a comprehensive mathematical model for personalised risk assessment of digital user behaviour, followed by interactive visualisation to support operational decision-making. The aim of the research was to create a model that allows for accurate analysis of individual and situational vulnerability factors, prediction of risky behaviour, and adaptation of protective measures in real time. For the model implementation, a combination of Bayesian analysis, Markov decision-making processes, regression methods, and modern data visualisation tools was used. As a simulation-based, the model was tested on 500 artificially generated user profiles reflecting different levels of digital literacy and behavioural responses to phishing scenarios. The results showed that individualised training significantly reduces the risk of phishing attacks – in some cases by 40%. The built model achieved a prediction accuracy of 85%, demonstrating high efficiency even when taking into account behavioural exceptions. It was found that stress, time constraints, and difficult conditions increase the probability of errors by 25%. At the same time, regular interaction with simulated threats makes it possible to build stable skills – the so-called "risk memory" – which reduces the number of errors over time. The model integrated both behavioural parameters – level of knowledge, stress tolerance, user experience – and external factors, including the threat complexity and workload intensity. This allows for dynamic adjustment of security strategies. Use of Markov modelling allowed optimising training processes, reducing losses by 65%. Interactive dashboards provided individualised vulnerability monitoring and rapid response to potential threats. The practical value of the proposed approach lies in the possibility of its integration into corporate security systems and use in educational and telemedia programmes to improve cybersecurity

Keywords: mathematical modelling; data visualisation; Bayesian analysis; Markov processes; social engineering

Introduction

Cybersecurity is one of the key areas of information protection in the modern digital world. The number of attacks on information systems keeps growing, and the level of threats requires constant improvement of methods to prevent and stop these attacks. One of the main weaknesses is still the human factor – the social, psychological, and behavioural traits of users that directly affect security. According to the ENISA (European Union Agency for Cybersecurity) (2024), more than 70% of cybersecurity incidents happened because of user mistakes or social engineering attacks in 2024. The report also says that these types of incidents grew by 38% compared to 2022. The Verizon Business (2024) Data Breach Investigations Report gave similar results and confirms that human-related risks remain the main cause of security breaches.

Modern security systems, like cryptography, intrusion detection systems, and multifactor authentication, help reduce risks, but these technologies cannot totally get rid of the human factor. H. Ahmad *et al*. (2024) investigated the effectiveness of multi-layered security systems and found that even with strong technical barriers in place, social engineering remains the primary cause of most successful attacks. The authors emphasise that raising user awareness and training in safe behaviour

Suggested Citation:
Trofymchuk, V. (2026). Interactive visualisation and analysis of risks with a human factor. *Information Technologies and Computer Engineering*, 23(1), 35-45. doi: 10.31649/vitce/1.2026.35

*Corresponding author

have a long-term effect that exceeds the benefits of purely technical solutions.

Y. Qin *et al*. (2025) analysed the effectiveness of user training policies for preventing social engineering attacks under resource constraints. The authors showed that the optimal distribution of training interventions can increase user resistance to manipulation and significantly reduce the likelihood of successful compromise without increasing costs. The study also confirmed that even minimal adaptation of training policies can shape more stable user behaviour patterns, which in the long run reduces the risk of social engineering attacks.

The study by A. Alshehri (2024) explored AI-powered adaptive cybersecurity awareness training in the industrial sector. The author showed that using intelligent algorithms to personalise the training process increases the effectiveness of forming stable, secure user behaviour and reduces the response time to potential incidents. The study emphasises that adaptive learning based on artificial intelligence is an effective tool for strengthening the human factor in cyber defence and for systematically reducing risks associated with social engineering attacks. N. Sugunaraj (2024) created a hybrid model that combines regression analysis and machine learning methods to continuously update user risk profiles. The results showed a reduction in errors by almost 30% and the model's resistance to changes in input data, making it suitable for long-term use in corporate monitoring systems.

The use of Bayes-oriented structures makes it possible to obtain more realistic risk assessments compared to classical static analysis schemes. J. Wang *et al*. (2020) proposed a Bayesian approach to cyber risk assessment by extending the FAIR model and formalising the relationships between technical parameters and user behavioural factors. The authors showed that such a network allows for more accurate assessment of uncertainty and formal probabilistic conclusions with limited or incomplete input data. The study confirmed that.

Finally, K. Ahmed *et al*. (2024) proposed a deep learning-based method for the joint extraction of cyber entities and relations from unstructured cybersecurity text. This approach demonstrates that automated information extraction can support cyber threat analysis by organising heterogeneous security data into structured representations. The authors indicate that such methods enhance the analytical capabilities of cyber defence systems and facilitate more effective threat modelling in complex and evolving digital environments.

Even though cybersecurity research is growing rapidly, some problems remain. Most models still do not mix personal user traits – like knowledge, experience, and ability to handle stress – with outside factors such as how hard the attacks are, how much work people have, or changing conditions. Systems that can train users in real time and adjust protection strategies are not well-developed. Also, interactive visuals and dashboards are usually used only to watch what is happening, not to help predict and manage risks directly. The goal of this study was to create an integrated mathematical model for personalised cyber risk assessment. The model took into account both user behaviour and external factors, adapted training and protection strategies in real time, and used interactive visualisation to support decision-making.

Materials and Methods

The study involved simulation modelling of user behavioural responses to phishing attacks using generalised profiles. Parameterised scenarios were created in the Python environment using the SimPy framework for event-based modelling and the NumPy and pandas libraries for generating user attributes and interaction rules. A total of 500 hypothetical users (aged 18-60) were generated with different levels of digital literacy, stress resistance, and previous experience in countering cyber threats. The approach to formalising the mathematical model was based on previous experience in constructing optimisation models in related studies by the author (Trofymchuk, 2025). These scenarios replicated typical behavioural patterns described in leading scientific studies on phishing response (Ahmad *et al*., 2024) and allowed the evaluation of the proposed mathematical model under various levels of risk exposure. The modelling considered user reactions to different types of phishing attacks, including mass phishing e-mails, spear phishing with personalised messages, and social engineering through fake technical support requests. The methodology for evaluating the impact of these attacks on behavioural parameters was based on the approach proposed by M. Zaoui *et al*. (2024).

Statistical analysis of the simulation outputs was performed in the Python environment using the scipy.stats package. The analysis followed standard reliability criteria: p-values with a significance threshold $\alpha = 0.05$ and 95% confidence intervals were calculated to verify differences between groups. Regression analysis and the least squares method were used to calibrate model parameters by minimising the difference between predicted and simulated risk values and by estimating the impact of each factor – user awareness, attack complexity, and environmental conditions – on the overall probability of a successful attack.

The optimisation of training strategies was implemented using the Value Iteration algorithm (via the mdptoolbox library), which identified the sequence of awareness training and defensive actions that minimised the total expected loss $J(\pi)$ and determined the time points when introducing protective measures would yield the highest effect. Model sensitivity to key parameters was examined using a one-factor-at-a-time (OFAT) approach combined with Monte Carlo simulations (10,000 random samples per run). This allowed testing how changes in awareness, attack complexity, or environmental stress affected the calculated risk and adapting the risk management framework to different operational profiles of an organisation.

The development of a mathematical model for assessing cyber risks associated with the human factor was a key

element of the study. User awareness (U) was defined as the simulated level of cybersecurity knowledge and training, attack complexity (A) described the technical and psychological sophistication of phishing attempts, and environmental impact (E) represented stress conditions such as workload, time pressure, and policy support. The schematic representation of the proposed mathematical model of a successful security breach P is presented in Figure 1.

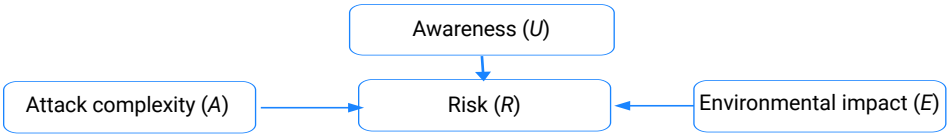

Figure 1. Diagrammatic representation of the mathematical model
Source: compiled by the author

It shows the relationship between the key variables – user awareness (U), attack complexity (A), and environmental impact (E) – and the resulting probability of a successful security breach (P). To ensure interpretability and reproducibility of the model, the parameters require a clearly defined scale and explanation of how the numerical values are set. This step makes it possible to link the visual structure of the model with the following mathematical formulation and to justify the ranges of α, β, γ, δ and other coefficients used in the equations:

$$P = f(U, A, E) = \alpha \cdot e^{-\beta U} + \gamma \cdot \delta \cdot E. \quad (1)$$

Parameters γ and δ represent sensitivity to external contextual variables such as stress or fatigue and were tuned based on behavioural deviations introduced into simulation environments. The term "behavioural exceptions" in this context refers to unpredictable deviations from typical user behaviour patterns – for instance, a digitally literate (Anderson, 2020) user failing to recognise a phishing threat due to psychological fatigue. These exceptions were incorporated into the model by adjusting either α or δ, allowing the model to remain adaptive in unstable behavioural conditions. All parameter values were validated using simulation-generated datasets and calibrated through iterative modelling of hypothetical user-phishing interactions reflecting real-world scenarios.

The proposed user risk assessment model is based on an exponential decrease in risk due to the level of awareness, but unlike classical models, it takes into account the influence of the environment and cognitive factors through the variable E. Additionally, the parameter σ is introduced, which allows flexible adjustment of the model for different attack scenarios, which increases the accuracy of risk prediction.

$$P = \alpha \cdot e^{-\beta U} + \gamma \cdot \delta \cdot E + \rho \ ln(1 + I_t), \quad (2)$$

where I_t – is the intensity of attacks in period t (the number of hacking attempts, phishing emails, social attacks, etc.); ρ is the coefficient of user sensitivity to attacks (determines how much an increase in attacks increases the likelihood of compromise). If a user receives a lot of phishing attacks, even with high awareness U, the user may eventually make a mistake. The logarithmic relationship $ln(1 + I_t)$ reflects the effect of the accumulating pressure of attacks – at first, the risk increases rapidly, but gradually stabilises. The value of I_t can be obtained by normalising the number of recorded incidents in security monitoring systems (e.g., SIEM logs) (NIST, 2020) over a specific time window. This allows the model to dynamically reflect variations in the threat landscape and to adapt to sudden spikes in attack intensity. The parameter ρ characterises the user's reactivity to increasing attack pressure. The model supports adaptive calibration of this parameter based on user characteristics such as digital literacy, stress resilience, or previous exposure to cyber threats. For instance, users with limited technical skills or high susceptibility to stress may be assigned higher values of ρ to reflect increased vulnerability. In simulation scenarios, ρ ranges between 0.05 and 0.2, representing different behavioural response profiles.

Consideration of user experience:

$$P = \alpha \cdot \beta (U + X) + \gamma \cdot \delta \cdot E, \quad (3)$$

where X – user experience (the ability to recognise attacks based on previous training or personal experience with cyber threats).

In this model, experience is seen as a static behavioural parameter that boosts the user's awareness (U) when assessing the likelihood of a security breach. The value of X is estimated based on the results of the user's participation in cybersecurity training and recorded responses to previous threats – for example, the success of recognising phishing messages or avoiding interaction with suspicious content. Quantitatively, this variable is expressed as a dimensionless coefficient ranging from 0 (no experience) to 1 (high level of experience) and is calibrated according to predefined assessment scales. This approach allows the results of individual training to be included in the risk assessment model without complicating it with time parameters. If a user has already encountered attacks, the attacks are less likely to be exposed to the user in the future, even if the user's awareness has not formally changed. In traditional learning models, exposure was considered static, but in practice, experience X is accumulated and makes it possible to avoid future threats.

MDP (Markov Decision Process) is used to model the dynamics of changes in the state of user awareness. Let

$S = \{S_0, S_1, \ldots, S_n\}$ – be a set of states, where corresponds to the minimum level of awareness and S_n – to the optimal one. The transition between states is determined by a function:

$$P(S_{(t+1)} \mid S_t, a_t), \quad (4)$$

where a_t – is an action, for example, participation in a training module; S_t– current state of awareness; $S_{(t+1)}$ – next potential state. The transition function reflects the probability that a user will transition to a new state given a certain action, taking into account both individual characteristics (level of basic awareness, cognitive biases, emotional state) and external factors. The transition probabilities are calibrated based on simulated scenarios covering different user profiles. For example, for users with a low level of initial awareness, participation in a short training course may have less of a transition effect than for users with an average level of awareness, which is reflected in lower values for the first category. The model also takes into account behavioural exceptions – situations where the user exhibits an unexpected reaction, such as reverting to a lower state due to stress or overconfidence. Such exceptions are modelled by entering individual values into the transition matrix, which describe non-standard trajectories of state changes.

In this way, the model does not predict deterministic improvement after each action, but integrates the probabilistic nature of user learning, allowing for the assessment of human-related risks in dynamics. The optimal strategy π minimises the total risk, which is determined by the loss function:

$$J(\pi) = E[\sum_{t=0}^{T} \gamma(S_1, a_t), \quad (5)$$

where $\gamma(S_1, a_t)$ is the immediate "penalty" or loss, and γ is the discount factor. This formula reflects the strategic goal of finding a sequence of actions that, on average, minimises the human factor's impact on cyber risks. It also allows the model to adapt to changes in user behaviour, limiting excessive generalisation and increasing the accuracy of real risk assessment in a dynamic environment. The loss function can be calibrated based on empirical data or simulations, allowing for the specifics of the model's application context, including organisational priorities, security policies, and acceptable residual risk levels.

For numerical evaluation, iterative algorithms such as Value Iteration are used to determine the optimal policy for transitions between states. The Markov decision-making process was used to optimise the sequence of training and defensive actions. Figure 2 illustrates the state transitions of user awareness levels under the influence of different training modules and adaptive security strategies. This diagram visually represents how the system determines the most effective path to reduce potential losses and strengthen user resistance to social engineering and phishing threats.

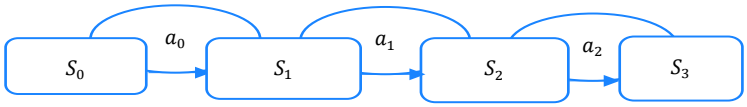

Figure 2. Diagram of the Markov decision-making process
Note: S_0 – training cycle; S_1 – training module; S_2 – monitored awareness state after training; S_3 – stabilised awareness state with reduced vulnerability; a_0 – baseline system; a_1 – interactive dashboard; a_2 – strategy optimisation
Source: compiled by the author

Figure 2 demonstrates the logical flow of user state changes during the learning process. Each state corresponds to a level of awareness, while transitions are triggered by training modules or by optimisation actions identified through the Markov decision process. This visualisation makes it possible to understand how the model selects adaptive responses to changing threat levels and user behaviour patterns. In contrast to the traditional Markov process, this model takes into account psychological factors, such as stress during an attack or the impact of training on user behaviour, which increases the accuracy of risk prediction:

$$P(S_{t+1}|S_t, a_t) = \frac{e^{-\gamma(1-U_t)}}{1+e^{-\delta E_t}}, \quad (6)$$

where U_t is the user's knowledge level at time t; E_t is the user's stress level at time t; γ, δ are the parameters of sensitivity to learning and stress.

A key feature of the proposed mathematical model is its ability to adapt to rapidly changing cybersecurity conditions and user behaviour. This adaptability was achieved by introducing dynamic parameters that can be recalibrated when new behavioural data or threat statistics become available. The model was designed to update risk estimations in near real time using both simulation outputs and observed interaction data from monitoring systems.

To enable this, the study integrated Bayesian inference as the main mechanism for updating the probability of user vulnerability. In this research, the hypothesis H was defined as "a user is vulnerable to a specific threat type" (e.g., phishing), while the observed data D represented user actions such as clicking on suspicious links or reporting an attack attempt. The posterior probability $P(H|D)$ was calculated using the standard Bayesian formula:

$$P(H|D) = \frac{P(H|D) \cdot P(H)}{P(D)}, \quad (7)$$

where $P(H|)$ is the prior probability of vulnerability, determined by simulated user characteristics (digital literacy, previous exposure to phishing, stress resistance), $P(H|D)$ is the likelihood of observing the user's action given vulnerability, and $P(D)$ is the normalising constant.

In this study, prior probabilities were initially set according to the generated user profiles: levels of digital literacy and stress tolerance followed predefined probability distributions, while previous exposure to cyber threats was modelled as a categorical variable with three levels (none, moderate, extensive). Likelihoods $P(H|D)$ were estimated by running repeated phishing scenarios in the simulation (mass phishing, spear phishing, and fake technical support requests) and recording user responses to each. Each new simulated or observed interaction updated the posterior vulnerability score $P(H|D)$, which was then used to adjust the overall risk.

To account for the variety and relative severity of different attack types, the Bayesian estimate was combined with weighting factors representing the impact of each threat category:

$$P(H|D) = \frac{P(H|D)\cdot P(H)}{P(D)} \times \sum_{i=1}^{n} w_i S_i, \quad (8)$$

where w_i are weights assigned to each attack type i (e.g., phishing, spear phishing, social engineering) according to its prevalence and potential impact, and S_i is the user's vulnerability score to that attack type after Bayesian updating. These weights were set based on the simulated threat environment, giving higher priority to frequent and high-impact attacks. This combined approach enabled the model to dynamically refine vulnerability predictions by merging prior user characteristics with updated behavioural observations and adjusting for the current mix of threats. As a result, the risk assessment remained context-aware and adaptive, avoiding static assumptions and better reflecting real-world conditions.

Results and Discussion

Interactive visualisation of risk modelling results

The developed model was tested on simulated user behaviour scenarios, which made it possible to analyse how the risk of successful attacks changes depending on user awareness, attack complexity, and external factors. To make the results easier to interpret, the outputs of the model were presented through interactive visualisation. This visualisation shows how the probability of a successful attack changes under different conditions and helps security specialists quickly evaluate the effect of training and external stress factors on user behaviour. The interactive dashboards created in this study support personalised risk analysis for both user groups and individual profiles. Data can be filtered by threat type, initial awareness level, or workload conditions. The graphs update automatically when new monitoring data becomes available, which keeps the analysis relevant and helps security teams react faster to changes in the threat landscape.

The conducted simulations showed that an increase in user awareness (U) led to a noticeable decrease in the probability of a successful attack. When the awareness level rose from 1 to 3 within the experimental model, the likelihood of compromise dropped from roughly 46% to 24%. This outcome demonstrates that even a moderate improvement in users' ability to recognise suspicious activity can almost halve the overall risk of a successful phishing attempt. The obtained results highlight the practical importance of adaptive training systems that enhance users' resistance to social engineering techniques.

Example of risk calculation using the model (formula (1)) to show how the model can be applied in practice, consider an organisation where the average user awareness level is $U = 2.0$, the attack complexity is $A = 0.6$, and the external impact (stress, workload) is $E = 0.4$. Then the risk is calculated as:

$$P = 0.5 \cdot e^{-0.22 \times 2.0} + 0.3 \cdot 0.6 + 0.2 \cdot 0.4.$$

The obtained value $P = 0.33$ (33%) represents the estimated probability of a successful attack under these conditions. This example shows how the model integrates behavioural and external factors to provide a clear numerical risk level, which can guide decisions about training intensity and preventive actions. To verify the model and illustrate how the calculated risk changes with different levels of user awareness, additional simulations were performed using formula (1). In these simulations, the baseline parameters for attack complexity and environmental impact were fixed at $A = 0.6$ and $E = 0.4$, while the user awareness level U varied from 1.0 to 3.0 in increments of 0.5. For each value of U, the probability of a successful security breach P was calculated, showing how higher awareness reduces risk when other factors remain constant (Table 1).

Table 1. Dependence of risk on the level of user awareness

Awareness level (U)	Risk (P), %
1.0	45
1.5	38
2.0	32
2.5	27
3.0	23

Source: compiled by the author

Table 1 clearly shows a steady decline in residual risk as user awareness increases, supporting the effectiveness of adaptive training modules designed to improve cybersecurity posture. This trend quantitatively confirms that increasing user awareness (U) from low to high levels leads to a consistent reduction in the predicted probability of compromise. The tabulated values show an almost linear risk decline, which makes it easier to calibrate

training intensity: when awareness rises from 1.0 to 3.0, the modelled residual risk decreases from 45% to 23%. Such results illustrate how awareness acts as a primary mitigating parameter in the proposed model and can be directly used to plan security training strategies. Based on a series of simulations, a graph was built (Fig. 3) showing how the residual risk *P* changes with different values of user awareness *U* when the other parameters are kept at average levels. The graph shows a steady decrease in risk as *U* increases, confirming that improving user awareness is one of the most effective strategies for reducing cyber risk.

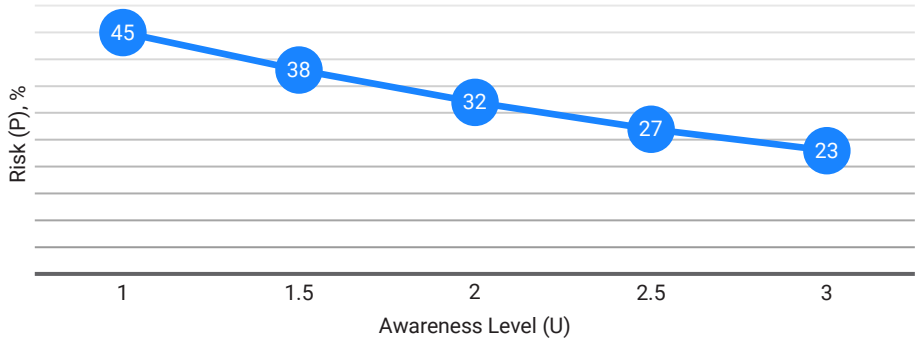

Figure 3. Residual risk *P* as a function of user awareness *U*
Source: compiled by the author based on simulation results of the proposed model

In addition to the line graphs, the results of the simulation were further analysed using heat maps that showed the distribution of user vulnerability under different conditions. Figure 4 demonstrateи a heat map of residual risk as a function of user awareness *U* (horizontal axis) and environmental stress *E* (vertical axis). Colour intensity indicates the predicted probability of a successful attack. The heat map highlights the highest risk zones where awareness is minimal and stress is maximal. As *U* increases, risk values drop sharply, even under strong external pressure, underscoring the value of adaptive awareness-building interventions.

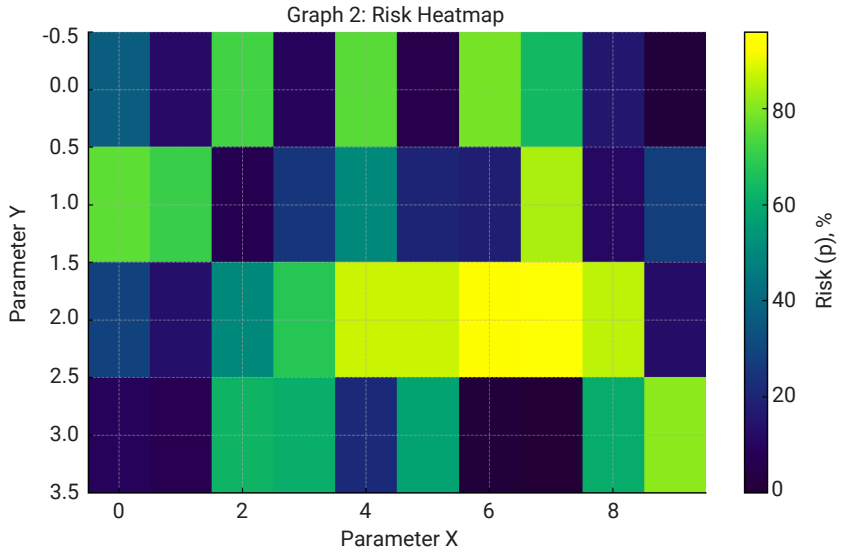

Figure 4. Heat map of residual risk depending on user awareness and stress factors (simulated data)
Source: compiled by the author

These visualisations helped to clearly identify high-risk areas – for example, user groups with low digital literacy and high stress levels, as well as scenarios where the complexity of attacks was above average. Timeline charts were also used to show how risk changed over time under the influence of adaptive training modules (Fig. 5). On average, the residual risk started to decrease within 10-14 days after the beginning of the training interventions, confirming the effectiveness of adaptive awareness improvement.

The modelling results also showed that adaptive configuration of the analytical environment enabled personalised risk assessment for both user groups and individual profiles. Filtering by threat type and initial awareness level (*U*) made it possible to identify segments with the highest predicted vulnerability and track how the

indicators changed after training interventions, as confirmed by the visualisation results shown in Figures 4-5. Thanks to the automatic updating of the analytical environment when new monitoring data is received, the indicators obtained remain relevant and support rapid data-driven decision-making.

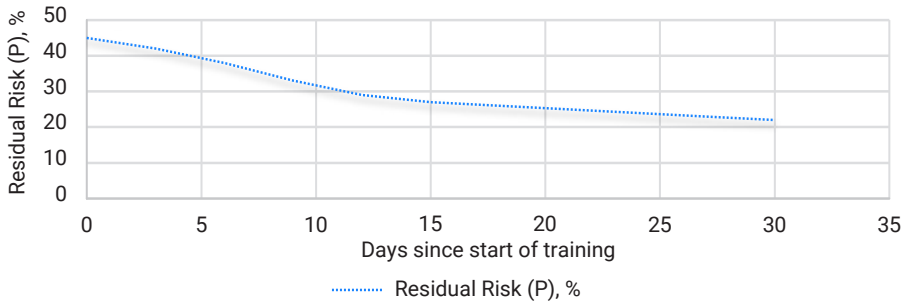

Figure 5. Time series showing residual risk reduction during adaptive training (simulated data)
Source: compiled by the author

Overall, the integration of heat maps, timeline views, and adaptive dashboards provided a clear way to connect the results of the mathematical model with practical decision-making in cybersecurity, as it enabled direct interpretation of modelled risk dynamics and identification of conditions requiring prioritised intervention. This visualisation approach does not simply describe the conceptual importance of visual representation but demonstrates, through simulation results, how the implemented tools support continuous risk monitoring and proactive security management based on data-driven insights.

To make the model not only descriptive but also actionable, it was extended with two calculation procedures. The first procedure finds the optimal decision strategy – it chooses which security actions (such as training modules or stricter controls) should be applied to reduce the expected risk over time. The second procedure updates the estimated probability that a user will remain vulnerable as new behavioural data appears. Here, the prior probability $P(H)$ means the initial belief that a user H is vulnerable to a certain type of threat (for example, phishing). Each time new data D about the user's actions is observed, this estimate is updated to a posterior probability $P(H|D)$.

Algorithm 1. Value Iteration for finding the best security policy

```
initialize V(s) = 0 for all states s
repeat until convergence:
 for each state s:
 V(s) = max_a [r(s, a) + γ * Σ P(s'|s, a) * V(s')]
```

Here: s– the current user state (e.g., awareness level); a – an action (e.g., run a training module); $r(s, a)$ – immediate risk/cost when taking action a in state s; γ – discount factor (gives less weight to future risk); $P(s'|s, a)$ – probability of moving to a new state s' after action a. When the values $V(s)$ stop changing, the action that maximises the expression gives the optimal policy – the best sequence of actions to minimise future cyber risk.

Algorithm 2. Bayesian update of vulnerability probability

```
for each observation D:
 P(H|D) = (P(D|H) * P(H)) / P(D)
```

Here: $P(H)$ – initial (prior) probability that a user is vulnerable; $P(H|D)$ – likelihood of seeing the observed behaviour if the user is vulnerable; $P(D)$ – normalisation factor; $P(H|D)$ – updated (posterior) probability after seeing the new data.

Explanation. This algorithm uses Bayes' rule to refine the assessment of user vulnerability after each new behavioural fact. Essentially, the model compares the data obtained with the behaviour of the user considered potentially vulnerable and adjusts the previous assessment. Thus, after each iteration, the system obtains a more accurate assessment that gradually better reflects the actual behavioural dynamics. Together with the first algorithm, this forms a closed loop: the system does not simply record risks, but constantly refines the level of vulnerability and adjusts the intensity of training or protection for a specific user accordingly. This approach avoids static assumptions and makes the model more sensitive to changing conditions. As a result, the risk information obtained becomes not only an analytical assessment, but also a practical guideline for choosing the most appropriate actions to enhance security. Implementation of the Value Iteration algorithm for policy optimisation showed that the cumulative expected loss $J(\pi)$ could be lowered to about 65% of its initial level when security actions such as training intensity adjustments were chosen dynamically. This finding indicates that combining Bayesian probability updating with Markov decision processes supports proactive and personalised cyber-risk management.

Statistical analysis and interpretation of simulation results

The statistical analysis of the simulated data revealed several stable patterns that explain how the proposed risk model behaves under varying conditions. All scenarios were generated synthetically; therefore, the reported values originate from controlled virtual experiments rather

than from real user groups. This approach made it possible to test the mathematical framework under reproducible and clearly defined parameters. Table 2 summarises the principal numerical outputs of the 500-profile simulation dataset and highlights how awareness, workload, and adaptive training influence predicted risk levels.

Table 2. Summary of the principal numerical outputs of the 500-profile simulation dataset

Indicator	Value	95% CI	p-value
Risk reduction when user awareness (*U*) increases by +0.5	18-22%	[16-24%]	< 0.001
Simulated reduction of attack probability after adaptive training (increase *U* from 1 → 3)	~40%	[36-44%]	< 0.001
Risk increase under high workload/stress	25%	[21-29%]	< 0.01
Prediction accuracy after Bayesian update	85%	[82-87%]	< 0.001

Source: compiled by the author

One of the key findings is the consistent reduction of residual cyber risk when adaptive training is introduced into the model. An increase in the user awareness parameter *U* by 0.5 units on the predefined scale (from low to higher awareness) resulted in an average 18-22% decrease in the predicted probability of a successful phishing attack while other factors such as attack complexity (*A*) and environmental stress (*E*) remained unchanged. This result confirms that awareness plays a decisive role in mitigating social-engineering-based threats. Simulated scenarios that introduced elevated workload and stress demonstrated a 25% average rise in the probability of compromise. This pattern indicates that non-technical factors such as time pressure and cognitive overload substantially increase vulnerability and should be explicitly considered when designing organisational security strategies. The Bayesian updating mechanism incorporated into the model achieved a risk-prediction accuracy of approximately 85% after iterative recalibration with new behavioural events. In the simulation, accuracy was calculated by comparing predicted risk values with the synthetic "true" attack outcomes generated for each profile. Continuous parameter tuning for *U*, *A*, and *E* allowed the model to remain stable and reliable even when new synthetic data were introduced.

These results underscore the importance of considering both technical and non-technical factors in the development of cybersecurity strategies. The model's ability to provide accurate, data-driven insights, even in the presence of fluctuating variables, highlights its potential for practical application in real-world scenarios. In summary, the statistical analysis validates the model's capacity to predict risk dynamics with high accuracy and provides valuable insights into how user awareness, workload, and adaptive training impact cyber risk. These findings suggest that a holistic approach, incorporating both behavioural and environmental factors, is essential for effective risk management and the development of personalised security strategies.

Previous studies have recognised the benefits of adaptive user training, but were based primarily on static assumptions about user behaviour (Bonneau *et al.*, 2012). B. Schneier (2015) emphasised the importance of the human factor, but the author's work did not offer a mathematical model that would allow for quantitative updating of risk over time. In contrast, in the presented study, the user's status is updated dynamically, depending on new data, which avoids fixed prior assumptions. M. Bada *et al.* (2015) demonstrated that educational interventions can reduce the risk of phishing attacks by up to 50%, but in the model, user vulnerability remained constant. In the proposed approach, this parameter is revised at each step of the simulation using Bayesian updating, which more accurately reflects the impact of training, stressors, and external conditions on behavioural risks in a real-world environment.

The study by A. Alshehri (2024) focused on the application of artificial intelligence algorithms for adaptive user training in industrial environments. The authors showed that a personalised approach to managing training interventions makes it possible to increase resistance to attacks and forms more stable patterns of user behaviour when interacting with risky digital scenarios. At the same time, the current study demonstrated that the use of adaptive learning models is an effective strategy for countering social engineering threats, even in high-load environments.

K. Kamatchi & E. Uma (2025) proposed a federated learning-based approach for detecting insider threats, with a focus on data privacy. However, the authors' study does not address personalised user training or behavioural parameter tuning. In contrast, the proposed model accounts for personalised influence by calibrating parameters *U*, *A*, and *E* at both group and individual levels, which improves model stability under increased stress conditions. Overall, the combination of personalised parameter tuning, Bayesian updating, Value Iteration, and interactive visualisation enables an adaptive risk assessment framework and demonstrates a higher potential risk reduction in simulation-based scenarios compared to earlier static approaches.

The ENISA Threat Landscape 2024 report indicated that a significant proportion of successful cyber incidents are caused by human-related factors, including cognitive overload, time pressure, and user fatigue, even in systems with advanced technical protection mechanisms. However, the report is descriptive in nature and does not provide quantitative models for dynamically assessing or updating behavioural risk at the individual user level (ENISA, 2024). In the authors' work, M.J. Hossain *et al.* (2025) proposed an

explainable AI-based framework combined with synthetic data to improve the transparency of intrusion detection systems in NextG network infrastructures. While the approach enhances interpretability at the network level, it does not address behavioural risk modelling or adaptive user training influenced by stress or awareness dynamics. L. Huang *et al*. (2011) demonstrated that learning-based security systems are vulnerable to adversarial manipulation when attackers adapt the behaviour to the defensive model. This limitation highlights the importance of adaptive mechanisms capable of recalibration over time, which are incorporated in the proposed approach through Bayesian updating and dynamic policy optimisation. In the systematic review of current cybersecurity awareness and education tools/programs by L. Zhang-Kennedy & S. Chiasson (2020) it was concluded that many awareness programs remain largely static and insufficiently personalised, reducing the long-term effectiveness. In contrast, the proposed model introduces personalised parameter tuning and continuous adaptation based on behavioural feedback.

S.M.A. Shah *et al*. (2019) analysed social engineering threats and noted that many countermeasures fail because of insufficiently accounting for human psychological and behavioural factors. The authors' work, however, does not propose a formal quantitative model for integrating these factors into dynamic cyber-risk assessment. A. Tversky & D. Kahneman (1974) showed that human decision-making under uncertainty is systematically influenced by cognitive heuristics and biases, particularly under stress and time constraints. These findings provide a theoretical foundation for incorporating behavioural variability into cybersecurity risk models that involve human interaction. A structured approach to measuring user security awareness was proposed by I. Arpaci & K. Sevinc (2021). The authors demonstrated that behavioural assessment is a critical factor in evaluating the effectiveness of cybersecurity training programs. However, the researchers' work focuses on awareness measurement rather than on dynamic risk adaptation, which is addressed in the proposed model.

The results confirm the effectiveness of the proposed model in various simulation conditions. The analysis showed that taking into account user awareness, load, and adaptive learning allows for stable cyber risk assessments. This indicates the feasibility of using this approach for further research and practical development of human-centred security systems.

Conclusions

This study presented an integrated mathematical model for assessing cyber risks associated with the human factor. The model combines Bayesian updating of prior probabilities with Markov decision processes, allowing for dynamic reflection of changes in user behaviour and adjustment of risk forecasts in response to new behavioural and external data. This integration provided higher risk assessment accuracy and model adaptability to real-world conditions, which is important for organisations where risk changes rapidly, and the behavioural component dominates over technical factors. The simulation results demonstrated a clear correlation between user awareness, external stress factors, and residual compromise risk. Increasing awareness from low to high was accompanied by a decrease in the probability of a successful phishing attack from approximately 45% to 23%. In the example with average awareness levels and average values for attack complexity and external influence, the model showed a probability of compromise of about 33%. Dynamic analysis over time also showed that residual risk began to decrease significantly within 10-14 days after the application of adaptive learning interventions. This confirms the practical effectiveness of the optimised training strategy and demonstrates that even with limited resources, investments in behavioural change can yield significant results in a short period of time.

An additional value of the study is that the proposed model provides not only risk assessment, but also the ability to manage the dynamics of its change. Unlike traditional static approaches, the presented work implements an adaptation mechanism that allows the model to learn from new behavioural data and predict the effects of training interventions before these interventions are implemented. This made it possible to assess not only the current state of cyber risk, but also the potential effectiveness of future interventions, which is a key advantage of this work. In addition, the results obtained indicate that such adaptive mathematical models can be used not only by technical security teams, but also by risk management specialists and organisational management when planning cybersecurity policies. The proposed approach creates a structured and sound quantitative basis for selecting awareness-raising strategies, prioritising budget decisions, and predicting the expected effect of behavioural interventions before such interventions are implemented. This enhances the practical value of the model and confirms its applicability in real-world organisational settings where decisions need to be made quickly, based on data, and in accordance with acceptable risk levels. Future research could include expanding the set of behavioural parameters, testing the model on large real-world datasets, and developing hybrid solutions that combine transparent mathematical models with deep learning methods. Such developments could increase resilience to new classes of attacks, improve prediction accuracy, and strengthen adaptive risk management strategies in the face of increasingly complex cyber threats.

Acknowledgements

None.

Funding

The study was not funded.

Conflict of Interest

None.

References

[1] Ahmad, H., Ullah, F., & Jafri, R. (2024). A survey on immersive cyber situational awareness systems. *ArXiv*. doi: 10.48550/arXiv.2408.07456.

[2] Ahmed, K., Khurshid, S.K., & Hina, S. (2024). CyberEntRel: Joint extraction of cyber entities and relations using deep learning. *Computers & Security*, 134, article number 103579. doi: 10.1016/j.cose.2023.103579

[3] Alshehri, A. (2024). AI-powered adaptive cybersecurity awareness training for the industrial sector. *International Journal of Intelligent Systems and Applications in Engineering*, 12(4), 5493-5505.

[4] Anderson, R. (2020). *Security engineering: A guide to building dependable distributed systems* (3rd ed.). Hoboken: Wiley. doi: 10.1002/9781119644682.

[5] Bada, M., Sasse, M.A., & Nurse, J.R.C. (2015). Cyber security awareness campaigns: Why do they fail to change behavior? *International Journal of Human-Computer Studies*, 123, 118-131.

[6] Bonneau, J., Herley, C., van Oorschot, P.C., & Stajano, F. (2012). The quest to replace passwords: A framework for comparative evaluation of web authentication schemes. In *IEEE symposium on security and privacy* (pp. 553-567). San Francisco: IEEE. doi: 10.1109/SP.2012.44.

[7] ENISA (European Union Agency for Cybersecurity). (2024). *ENISA threat landscape 2024*. Retrieved from https://www.enisa.europa.eu/publications/enisa-threat-landscape-2024.

[8] Hossain, M.J., Alam, K., Monir, M.F., Hoque, M., & Ahmed, T. (2025). Explainable AI meets synthetic data: A deep learning framework for detecting network intrusion in NextG network infrastructure. *IEEE Access*, 13, 114979-115001. doi: 10.1109/ACCESS.2025.3585783.

[9] Huang, L., Joseph, A.D., Nelson, B., Rubinstein, B.I.P., & Tygar, J.D. (2011). Adversarial machine learning. In *Proceedings of the 4th ACM workshop on security and artificial intelligence* (pp. 43-58). New York: ACM. doi: 10.1145/2046684.2046692.

[10] NIST. (2020). *Security and privacy controls for information systems and organizations (SP 800-53r5)* (Rev. 5). Gaithersburg: NIST. doi: 10.6028/NIST.SP.800-53r5.

[11] Arpaci, I., & Sevinc, K. (2021). Development of the cybersecurity scale (CS-S): Evidence of validity and reliability. *Information Development*, 38(2), 218-226. doi: 10.1177/0266666921997512.

[12] Zhang-Kennedy, L., & Chiasson, S. (2020). A systematic review of multimedia tools for cybersecurity awareness and education. *ACM Computing Surveys*, 54(1), 1-39 pages. doi: 10.1145/3427920.

[13] Qin, Y., Yang, X., Yang, L-X., & Huang, K. (2025). Mitigating social engineering attacks through cost-effective security awareness training policy. *IEEE Transactions on Network Science and Engineering*, 12(4), 3145-3158. doi: 10.1109/TNSE.2025.3556927.

[14] Schneier, B. (2015). *Data and Goliath*. New York: W.W. Norton & Company.

[15] Shah, S.M.A., Ahmed, A., & Ali, M.A. (2019). Social engineering threats and countermeasures in SHCT. *International Journal of Business Intelligence*, 8(2), 44-46. doi: 10.20894/IJBI.105.008.002.004.

[16] Sugunaraj, N. (2024). Human factors in the LastPass breach. *ArXiv*. doi: 10.48550/arXiv.2405.01795.

[17] Kamatchi, K., & Uma, E. (2025). Securing the edge: Privacy-preserving federated learning for insider threats in IoT networks. *The Journal of Supercomputing*, 81, article number 246. doi: 10.1007/s11227-024-06752-z.

[18] Trofymchuk, V. (2025). Development of a mathematical model to improve the efficiency of telecommunication networks. *International Science Journal of Engineering & Agriculture,* 4(2), 26-38. doi: 10.46299/j.isjea.20250402.03.

[19] Tversky, A., & Kahneman, D. (1974). Judgment under uncertainty: Heuristics and biases. *Science*, 185(4157), 1124-1131. doi: 10.1126/science.185.4157.1124.

[20] Verizon business. (2024). *Data Breach Investigations Report (DBIR) 2024*. Retrieved from https://www.verizon.com/business/resources/reports/2024-dbir-data-breach-investigations-report.pdf.

[21] Wang, J., Neil, M., & Fenton, N. (2020). A Bayesian network approach for cybersecurity risk assessment implementing and extending the FAIR model. *Computers & Security*, 89, article number 101659. doi: 10.1016/j.cose.2019.101659.

[22] Zaoui, M., Yousra, B., Yassine, S., Maleh, Y., & Ouazzane, K. (2024). A comprehensive taxonomy of social engineering attacks and defense mechanisms: Toward effective mitigation strategies. *IEEE Access*, 12, 72224-72241. doi: 0.1109/ACCESS.2024.3403197.

Інтерактивна візуалізація та аналіз ризиків з урахуванням людського чинника

Вікторія Трофимчук
Магістр, викладач
Державний університет «Київський авіаційний інститут»
03058, просп. Любомира Гузара, 1, м. Київ, Україна
https://orcid.org/0000-0002-9756-0244

Анотація. Людський чинник залишається однією з ключових вразливостей у сучасному кіберсередовищі, що підкреслює важливість аналізу поведінки користувачів у системах управління ризиками. У цьому дослідженні представлено комплексну математичну модель для персоналізованої оцінки ризиків, пов'язаних із цифровою поведінкою користувачів, з подальшою інтерактивною візуалізацією для підтримки оперативного прийняття рішень. Метою дослідження було створення моделі, яка дозволяє точно аналізувати індивідуальні та ситуаційні чинники вразливості, прогнозувати ризиковану поведінку та адаптувати захисні заходи в режимі реального часу. Для реалізації моделі було використано комбінацію байєсівського аналізу, марковських процесів прийняття рішень, регресійних методів і сучасних засобів візуалізації даних. Як основу симуляційного моделювання, модель було протестовано на 500 штучно згенерованих профілях користувачів, що відображають різні рівні цифрової грамотності та поведінкових реакцій на фішингові сценарії. Результати показали, що індивідуалізоване навчання користувачів суттєво знижує ризик фішингових атак до 40 %. Створена модель досягла точності прогнозування на рівні 85 %, демонструючи високу ефективність навіть із урахуванням поведінкових винятків. Було встановлено, що стрес, обмеження часу та складні умови підвищують імовірність помилок приблизно на 25 %. Водночас регулярна взаємодія із симульованими загрозами сприяє формуванню стійких навичок – так званої «пам'яті на ризики», що зменшує кількість помилок з часом. Модель інтегрує як поведінкові параметри – рівень знань, стресостійкість, досвід користувача, – так і зовнішні чинники, включно зі складністю загроз та інтенсивністю навантаження. Це дозволяє динамічно налаштовувати стратегії захисту. Використання марковського моделювання дало змогу оптимізувати навчальні процеси, зменшивши втрати часу та ресурсів на навчання користувачів на 65 %. Інтерактивні інформаційні панелі забезпечили індивідуалізований моніторинг вразливостей та швидке реагування на потенційні загрози. Практична цінність запропонованого підходу полягає у можливості його інтеграції в корпоративні системи безпеки та використання в освітніх і телекомунікаційних програмах для підвищення цифрової грамотності

Ключові слова: математичне моделювання; візуалізація даних; байєсівський аналіз; марковські процеси; соціальна інженерія

Journal homepage: https://itce.vn.ua/en

Information Technologies and Computer Engineering **Vol. 23, No. 1. 2026**

Article's History: Received: 26.11.2025 Revised: 23.02.2026
Accepted: 26.03.2026 Published: 20.04.2026

ISSN 1999-9941; e-ISSN 2078-6387
UDC 004.85
DOI: 10.31649/vitce/1.2026.46

Comparative analysis of machine learning algorithms for personalising educational content in distance learning

Vitalii Yanishevskyi*
Postgraduate Student
European University
03115, 16V Academician Vernadskyi Blvd., Kyiv, Ukraine
https://orcid.org/0009-0001-5774-4778

Abstract. The aim of this research was to conduct a comprehensive evaluation of the effectiveness of machine learning algorithms for the task of personalised educational content recommendation in distance education systems. The study was of a theoretical-experimental nature and was performed using a synthetic dataset comprising 10,000 student profiles, constructed based on the structural characteristics of leading distance learning platforms. The dataset covered three groups of features: demographic, behavioural, and content-related, replicating key patterns of student interaction with the learning environment. A comparative analysis of the effectiveness of the Support Vector Machine (SVM), Decision Tree, Random Forest, and Multilayer Neural Network methods revealed clear quantitative differences between the models. The highest classification results were obtained for the Neural Network (accuracy = 0.91; F1-score = 0.90). The ensemble-based Random Forest model provided high stability and accuracy (accuracy = 0.89; F1-score = 0.87). The Support Vector Machine method showed balanced performance (accuracy = 0.86; F1-score = 0.83), while the Decision Tree exhibited the lowest effectiveness (accuracy = 0.72; F1-score = 0.70), confirming the limitations of interpretable models in multidimensional data. An additional systematic analysis, performed using semi-quantitative indices for six algorithm characteristics, reflected the overall suitability of the models for personalisation: the Neural Network scored 23 points, Random Forest – 21 points, SVM – 19 points, Decision Tree – 17 points. These scores align with the classification metrics and confirm the advantages of models with pronounced non-linearity and ensemble structure. The Multilayer Neural Network demonstrated the highest efficacy for deep content personalisation, Random Forest serves as a universal model for large-scale educational platforms, the Support Vector Machine method is optimal for courses with clearly segmented student groups, while the Decision Tree is advisable to use as an interpretable analytical module. The practical significance of the study lies in forming a scientifically grounded approach to selecting algorithms for building adaptive educational trajectories and improving the effectiveness of digital education

Keywords: digital environment; Learning Management System; hyperparameter optimisation; neural networks; synthetic dataset

Introduction

The relevance of this study is driven by the rapid development of distance education and the growing need for personalised digital learning environments capable of adapting the content and presentation of material to the individual characteristics of students. The heterogeneity of proficiency levels, diverse cognitive styles, information overload, and uneven learning motivation render traditional approaches to online education insufficiently effective. Personalisation has become a key requirement for modern electronic platforms, as it ensures the relevant selection of educational content, optimisation of the learning path, and increased engagement of learners. In this context, the application of machine learning algorithms opens the possibility of creating adaptive recommendation systems that form individual learning trajectories based on the analysis of real student behavioural and academic data.

The issue of using artificial intelligence in adaptive educational environments is receiving increasing scholarly attention. For instance, O. Zadorina *et al*. (2025) substantiated the advantages of intelligent tools for creating adaptive courses, emphasising the importance of an algorithmic approach to constructing personalised learning trajecto-

Suggested Citation:
Yanishevskyi, V. (2026). Comparative analysis of machine learning algorithms for personalising educational content in distance learning. *Information Technologies and Computer Engineering*, 23(1), 46-59. doi: 10.31649/vitce/1.2026.46

*Corresponding author

ries. The study by V. Motorina *et al*. (2025) demonstrated the effectiveness of applying artificial intelligence for the automated creation of personalised materials in higher education systems, highlighting the value of algorithmic content adaptation to individual student needs. In turn, B. Shevchuk (2025) proved that the integration of AI into virtual educational environments provides flexible personalisation of content and informatics competencies, which is critically important for professionally-oriented learning.

In international research, there is also a growing interest in using machine learning for adapting the complexity, format, and volume of educational content. For example, W. Villegas-Ch *et al*. (2024) proposed ML-models for adapting materials to individual learning styles, proving that the correct selection of an algorithm significantly impacts recommendation accuracy. F. Ma (2025) developed a behavioural analytics system for online platforms that generates personalised recommendations based on characteristics of learning activity, including time-on-task and content viewing patterns. In the work of F.Z. Lhafra & O. Abdoun (2025), the effectiveness of different Machine Learning (ML)-models for determining learning styles was compared, showing significant discrepancies in algorithm performance depending on data characteristics. Also significant is the study by M. Soui *et al*. (2022), which demonstrated the possibility of creating intelligent platforms that adapt the educational process according to student behaviour in real-time.

Research on the implementation of deep learning in personalised educational systems also demonstrates high effectiveness. As noted by F. Naseer *et al.* (2024), deep learning models enable the construction of individual learning pathways considering a multitude of behavioural and content features, although their practical implementation requires significant computational resources. In the work of W. Chen *et al*. (2024), it was confirmed that machine learning significantly improves the accuracy of recommendation systems in digital educational platforms, especially when using structured data and weighted hyperparameter optimisation.

The results of the source analysis indicate that the further development of personalised educational platforms depends on the ability of machine learning algorithms to ensure high accuracy in classification and recommendations within the context of real student behaviour. Despite the existence of a large number of studies, the analysed literature insufficiently addresses the direct comparison of different ML-algorithms on a single dataset for the task of recommending types of educational content. There is also a lack of metric-driven experiments with controlled hyperparameter selection and a quantitative analysis of the mathematical nature of the differences between models. Issues concerning the alignment of educational data characteristics with the choice of specific algorithms, as well as the assessment of model stability across different scenarios of learning behaviour, remain insufficiently developed. These gaps define the need for a comprehensive comparative study based on a unified experimental design and standardised approaches to building ML-models, with the aim of forming scientifically grounded recommendations for developers and researchers of distance learning platforms.

The aim of the research was to comprehensively analyse the performance of machine learning algorithms in solving the task of personalised selection of educational content in the context of distance education. To achieve this aim, three main objectives were defined: to substantiate the theoretical foundations of educational content personalisation and the application of machine learning algorithms in distance education; to construct an experimental dataset and implement a unified model comparison design, considering data preparation and validation procedures; to perform a comparative evaluation of the algorithms based on defined metrics and determine the factors causing differences in their performance in personalised recommendation tasks.

Materials and Methods

The theoretical and experimental investigation was conducted from January to October 2025 using methods of machine learning, statistical modelling, and computer simulation of behavioural and content patterns in distance learning environments. The formation of a synthetic dataset, hyperparameter optimisation, repeated cross-validation, and comparative analysis of algorithms were implemented within a single experimental cycle, ensuring the reproducibility and correctness of the obtained results. The methodological foundation comprised machine learning, statistical, comparative-analytical, and validation approaches, which enabled the evaluation of the effectiveness of various classification algorithms for the task of personalised learning content type recommendation. The study employed a unified experimental design covering data structuring, pre-processing, model training, hyperparameter tuning, and repeated cross-validation, ensuring the comparability of the obtained results.

The data source was a specially created synthetic dataset, generated based on the characteristics of real Learning Management System (LMS) platforms, specifically Moodle, Coursera, and EdX, replicating key elements of student digital interaction with learning content. The total volume of the generated dataset comprised 10,000 student profiles, providing sufficient parameter variability for the correct operation of classification algorithms. The data structure encompassed three main groups of features: student demographic parameters (age, language of instruction, educational level), behavioural characteristics (time-on-task, number of learning material views, completion rate, number of test attempts, average score), as well as content features (material type, topic complexity, discipline domain, content format). The target variable corresponded to the type of learning content the system should recommend to the student: video, text, test, or mixed format. To preserve the representativeness of the synthetic data, stochastic

Normal, Log-normal, and Poisson distributions were used, allowing for the formation of heterogeneous student profiles characteristic of real distance learning courses.

Data pre-processing was carried out in accordance with a general ML pipeline: numerical variables were normalised using the z-score method, categorical features were encoded via one-hot encoding, and the class imbalance issue was corrected by applying the Synthetic Minority Oversampling Technique (SMOTE) (Chawla *et al*., 2002), ensuring uniform representation of each of the four types of learning materials. Missing values were addressed using a combination of mean-imputation and regression-based imputation methods depending on the nature of the feature. All models were trained on the same data split: 80% for the training set and 20% for testing; additionally, k-fold cross-validation (k = 10) was applied, allowing for the assessment of model stability and the generalisability of their results. Hyperparameter optimisation was performed using a combined approach: initial parameter search via GridSearchCV was supplemented with RandomizedSearchCV, expanding the search space and minimising the risk of local optima.

The study implemented four classes of machine learning algorithms most commonly used in personalised educational systems: Support Vector Machine (SVM), Decision Tree, Random Forest (RF), and multilayer artificial neural networks. To ensure objectivity in the comparison, hyperparameter tuning was conducted in two stages: initial manual optimisation followed by an automated search using GridSearchCV and RandomizedSearchCV implemented in the scikit-learn library. For the SVM model, the radial basis function kernel, regularisation parameter C, and kernel width γ were optimised; for the Decision Tree, the maximum depth and split criterion (Gini/Entropy) were adjusted; for Random Forest, the number of trees n_estimators and maximum depth were tuned; for the neural network, the number of layers, number of neurons, activation types (ReLU, tanh), and backpropagation algorithm parameters were configured. Optimisation was carried out using a unified set of metric criteria, ensuring the correctness of inter-model comparison.

The comparative evaluation of algorithms was performed based on four key metrics traditionally applied in multiclass classification tasks: accuracy, precision, recall, and F1-score. Formally, the F1-score metric is defined by relation (1):

$$F1 = 2 \times \frac{Precision \cdot Recall}{Precision + Recall}, \quad (1)$$

where $Precision = \frac{TP}{TP+FP}, Recall = \frac{TP}{TP+FN}$, where TP – true positives – the number of correctly classified examples of a certain class; FP – false positives – the number of examples incorrectly assigned to a class that actually belong to another class; FN – false negatives – the number of examples that belong to the class but were not recognised as such by the model; *Precision* – the proportion of correct positive predictions among all model predictions for this class; *Recall* – the proportion of correctly identified examples of a class by the model among all real examples of that class; *F1-score* – the harmonic mean between precision and recall, demonstrating a balance between accuracy and recall.

For the multiclass task (in this case, classes "video", "text", "test", "mixed format"), the macro-averaged value is applied (2):

$$F1_{macro} = \frac{1}{K}\sum_{i=1}^{K} F1_i, \quad (2)$$

where $K = 4$ – the number of classes, $F1_i$ – the value of the *F1-score* metric for each class, $F1_{macro}$ – ensuring equal weight for all content categories regardless of their proportion in the dataset.

The use of the macro-averaged metric prevents the domination of classes with a larger number of samples and ensures objective inter-model performance comparison. The experimental models were implemented in the Python 3.12 environment using the NumPy, pandas, scikit-learn, and matplotlib libraries. Preparation and validation of the source data were performed in pandas, while preprocessing was carried out using the scikit-learn.preprocessing module, ensuring uniform transformation of numerical and categorical features prior to model training. Hyperparameter optimisation was implemented using GridSearchCV and RandomizedSearchCV, within which extended parameter ranges were tested: for the support vector method – C from 0.1 to 20 and γ within 0.001-0.2; for the Decision Tree – maximum depth from 4 to 16 and the gini and entropy criteria; for the Random Forest – the number of trees within 100-350 and depth of 10-20 levels; for the multilayer neural network – combinations of 2-3 hidden layers in the range of 32-128 neurons with an initial learning rate of 0.01-0.0005 and relu or tanh activations. The application of such parameter intervals yielded stable optimal configurations that achieved the highest classification metrics among the tested algorithms. Visualisation of the results was performed in matplotlib, enabling a clear representation of the behavioural patterns in the synthetic dataset and the comparative performance curves of the models. The uniformity of training and testing conditions was controlled by fixing the random_state, ensuring the reproducibility of experiments. To verify model stability, repeated runs with varying initial conditions were conducted, allowing for the assessment of algorithm behaviour under altered data configurations. This approach ensured a comprehensive comparison of machine learning algorithms in the task of educational content personalisation, allowed for the identification of each model's strengths and weaknesses, and created methodological prerequisites for formulating recommendations regarding the choice of technological solutions for distance learning systems.

Results

Technical analysis of SVM, Decision Tree, Random Forest, and multilayer neural networks algorithms

The conducted technical analysis of the four algorithms – SVM, Decision Tree, Random Forest, and multilayer neural

networks – established key differences in their classification solution construction mechanisms, which directly impact their effectiveness in the content personalisation task. The analysis revealed that the models demonstrate varying sensitivity to data structure, noise volume, variability of behavioural characteristics, and the level of feature non-linearity. This determines different potential scenarios for their application in distance learning systems. SVM showed high stability in the presence of complex boundaries between classes, reflected in its ability to form separation even under conditions of significant overlap in student characteristics. Decision Tree exhibited a strong dependence on tree depth: models with uncontrolled complexity are prone to overfitting, whereas optimally constrained depth provides satisfactory generalisation. The analysis of Random Forest confirmed that the ensemble approach significantly reduces result variability compared to a single tree, and increasing the number of trees improves robustness to noise. Neural networks demonstrated the ability to replicate complex multidimensional dependencies between student behavioural features and the type of recommended content, maintaining stable accuracy with increasing data structure complexity. To systematise the obtained results, a comparative Table 1 was constructed, summarising the identified theoretical advantages and limitations of each algorithm, as well as determining which hyperparameters have the greatest impact on the final result in the task of classifying types of learning content.

Table 1. Theoretical distinctions and key hyperparameters of algorithms

Algorithm	Mathematical idea (method core)	Key hyperparameters
SVM	Maximisation of the margin between classes; utilisation of kernel transformations for nonlinear boundaries	C, γ, kernel
Decision Tree	Recursive data partitioning based on Gini or Entropy criterion	max_depth, criterion, min_samples_split
Random Forest	Ensemble of trees with bootstrap sampling, majority voting	n_estimators, max_depth, max_features
Neural Network (MLP)	Nonlinear multilayer mapping of the feature space; error backpropagation	layers, neurons, activation, learning_rate

Note: MLP – Multilayer Perceptron
Source: developed by the author using Python 3.12 and the NumPy, pandas, and scikit-learn libraries

As evident from Table 1, each algorithm demonstrates a unique mechanism for forming a classification decision, which leads to varying sensitivities to the structure of educational data. For SVM, the defining feature is the ability of parameter C to regulate the trade-off between the margin width and the number of permissible errors, while γ determines the curvature of the decision boundary, which is particularly important in the context of complex student behavioural features. In Decision Trees, classification quality directly depends on tree depth: excessive values of max_depth cause local overfitting to minor patterns, while values that are too small result in the loss of important distinctions between content types. Random Forest compensates for these shortcomings through its ensemble structure: increasing n_estimators enhances the model's robustness to noise, and the max_features parameter determines the degree of diversity among the trees, which affects the model's ability to recognise non-trivial patterns. Neural networks demonstrate the highest behavioural variability: altering the number of layers and neurons can significantly change the depth of data abstraction, while the choice of activation functions shapes the nature of nonlinear transformations. Collectively, these distinctions indicate that the optimality of an algorithm for content personalisation tasks is determined not only by its mathematical principle but also by the alignment of its hyperparameters with the structure of the training data.

In summarising the conducted technical analysis, it can be asserted that the considered algorithms form distinctly different strategies for constructing classification solutions, and it is this difference that determines their subsequent performance in personalised educational systems. SVM proves to be most effective in environments with clear or nonlinear class boundaries; Decision Tree demonstrates high interpretability provided its complexity is carefully controlled; Random Forest confirms the advantages of the ensemble approach due to its robustness to noise and generalisation capability; neural networks provide the deepest modelling of hidden dependencies but require significantly more meticulous configuration. The totality of these characteristics forms the basis for predicting their efficacy in the experimental phase, where these theoretical distinctions will manifest as varying levels of accuracy, balance, and the model's ability to adapt to student behavioural patterns.

Construction of the experimental dataset and characterisation of data typical for modern LMSs

The formulated synthetic dataset replicates the multi-level structure of educational data typical of modern LMSs and includes three logical feature groups – demographic, behavioural, and content-related. Each group serves a separate analytical function in the content personalisation model: demographic parameters reflect the individual characteristics of a student, behavioural characteristics capture the dynamics of their learning activity, and content variables define the properties of the materials to be recommended. To generalise the structure of the constructed dataset, Table 2 is provided, which systematises the main feature types and their role in forming a user profile.

Table 2. Structure of the synthetic experimental dataset

Feature group	Example variables	Data type	Brief characterisation	Source of feature formation (LMS)
Demographic	Age, language of instruction, educational level	Categorical + numerical	Reflects basic individual characteristics of the student	Moodle, Coursera, EdX
Behavioural	Time-on-task, number of views, completion rate, number of test attempts, average score	Numerical (normal, log-normal, poisson)	Forms the behavioural profile of the user during interaction with the LMS	Moodle, Coursera
Content-related	Type of material, topic complexity, subject domain, content format	Categorical	Describes the structure and properties of the learning materials	Coursera, EdX

Source: developed by the author using Python 3.12, NumPy, and pandas

As can be seen from Table 2, the structure of the synthetic dataset is formed based on three different platforms – Moodle, Coursera, and EdX, which enabled the integration of heterogeneous types of educational data into the model and ensured the representativeness of student activity patterns. The comparison shows that each LMS contributes to the formation of distinct feature groups, creating a balanced and multidimensional user profile. Moodle predominantly defines the structure of behavioural data, particularly metrics such as time-on-task, number of views and activities, and module completion rates. This platform is distinguished by detailed event analytics, allowing the model to capture sharp fluctuations in user learning activity. This makes behavioural features derived from Moodle more variable and informative for algorithms sensitive to data dynamics, such as Random Forest and neural networks. Coursera, unlike Moodle, provides a mixed contribution – to both behavioural and content indicators. The standardised course structure and detailed learning trajectories allow the model to account for significant differences between students with varying levels of preparation and work intensity. Data from Coursera best reflects the relationship between learning activity and module completion, which is key for multi-class classification of content types. EdX has the greatest influence on the formation of content-related features, as this platform has a clear structuring of disciplines, material complexity, and content presentation formats. This allows for more precise modelling of the dependency between the properties of a learning resource and which type of material will be optimal for a specific student. EdX's content features align well with algorithms sensitive to nonlinear relationships between parameters, especially SVM and MLP. Overall, the three platforms create a complementary structure: Moodle provides dynamics, Coursera provides integrated behavioural-content interaction, and EdX provides structural depth of learning materials. It is precisely this approach that enables the formulated dataset to replicate a realistic multi-level student profile, allowing subsequent machine learning algorithms to operate under conditions as close as possible to data from real educational platforms. For a visual summary of the obtained characteristics, density curves of key behavioural parameters – time-on-task and views – were constructed, reproducing the most common patterns of student interaction with the digital environment. Figure 1 provides a graphical representation of these distributions, illustrating the main patterns characteristic of real educational platforms.

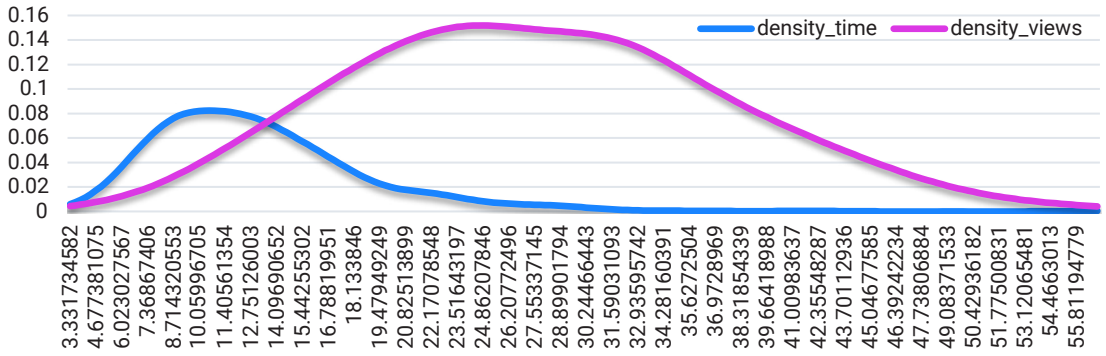

Figure 1. Density distribution of behavioural and content features of the synthetic dataset $n = 10{,}000$
Note: the X-axis displays the values of the synthetic dataset indicators: for time-on-task – the duration of student interaction with the learning material in seconds; for views – the number of views of individual content fragments. The Y-axis presents the probability density estimates calculated using kernel density estimation
Source: developed by the author using Python 3.12, NumPy and matplotlib

The integrated density curves for the time-on-task and views indicators obtained in Figure 1 demonstrate two fundamentally different behavioural patterns of students, which directly impact the quality of personalised

recommendations. Firstly, it is noteworthy that the time-on-task curve exhibits pronounced right-skewness: the majority of students spend a relatively small amount of time studying the material, while the distribution tail gradually extends into the region of high values. This is characteristic of log-normal distributions and indicates the presence of a small group of students who systematically work slower or more attentively, forming a distinct behavioural cluster. Precisely for such users, algorithms sensitive to imbalanced data distribution (e.g., Decision Tree) may exhibit unstable results without hyperparameter tuning.

In contrast, the views curve has a more clearly defined peak region with moderate variance, indicating relatively stable patterns of interaction with learning materials. Despite isolated cases of intensive content viewing, most students demonstrate a similar level of activity, which enhances the informativeness of this feature for classification models, particularly ensemble methods (Random Forest), which perform better with stable, structurally homogeneous data. A comparison of the two curves shows that views has a more "compact" distribution, while time-on-task is characterised by significantly greater variability and potential presence of noise.

The visual combination of the two behavioural features enables the identification of zones of potential information gaps where models may lose accuracy. For instance, the narrow peak of views at low time-on-task values may indicate users who review the material superficially or quickly skip content – neural networks demonstrate better results with such users, as they are capable of capturing weakly structured dependencies between features. In turn, the sharply pronounced tail of time-on-task signals that SVM with a Radial Basis Function kernel will be more stable in this region due to its ability to adapt the classification boundary to non-uniformly distributed points.

The obtained curves also allow for an assessment of the potential information contribution of each feature. Time-on-task can serve as a predictor of the type of recommended content for students with non-standard working paces, while views will be more useful in predicting the choice between video and textual materials among average users. This aligns with the fact that different machine learning algorithms exhibit varying abilities to model behavioural dynamics, and the very shape of the distributions essentially determines which models will be most sensitive to the nature of the data.

In summary, the analysis shows that combining density curves allows for the detection of critical differences between student behavioural patterns and predicts the effectiveness of individual algorithms in personalising educational content. The presence of pronounced skewness in one feature and compactness in another creates a natural test space in which algorithms demonstrate their strengths and weaknesses, making such visualisations a key tool for further modelling and optimisation.

Implementation of the ML-pipeline and hyperparameter optimisation results

Implementing the complete ML-pipeline made it possible to obtain consistent results for all models under standardised conditions, ensuring the correctness of inter-algorithm comparisons. The conducted normalisation of numerical features, encoding of categorical variables, and sample balancing using SMOTE reduced inter-class bias and increased model training stability. After splitting the data into training and test sets (80/20) and performing 10-fold cross-validation, it was established that all models demonstrate reproducible results: the standard deviation of the F1-score did not exceed 0.02-0.03, and the variation range between folds remained within $\leqslant 0.05$, which corresponds to acceptable model stability criteria. Hyperparameter optimisation revealed clear patterns in configurations that ensured the highest model generalisation. The combined application of GridSearchCV and RandomizedSearchCV allowed for the identification of parameter zones where each algorithm achieves maximum performance. The aggregate results of the performed optimisation are summarised in Table 3, which presents the final parameter values that provided the best accuracy and stability indicators during modelling.

Table 3. Selected model hyperparameters after optimisation

Algorithm	Optimised hyperparameters	Selected values
SVM (RBF)	C, γ, kernel	$C = 12$; $\gamma = 0.087$; kernel = 'rbf'
Decision Tree	max_depth, criterion, min_samples_split	max_depth=14; criterion='entropy'; min_samples_split = 4
Random Forest	n_estimators, max_depth, max_features	n_estimators = 300; max_depth = 18; max_features = 'sqrt'
Neural Network (MLP)	layers, neurons, activation, learning_rate	3 hidden layers (64-32-16), activation = 'relu', learning_rate = 0.001

Note: SVM (RBF) – Support Vector Machine with Radial Basis Function kernel
Source: developed by the author using Python 3.12, scikit-learn, NumPy and pandas

The optimised values from Table 3 reveal the peculiarities of each algorithm's operation with multidimensional educational data. SVM received a moderate regularisation level ($C = 12$) and a low kernel curvature ($\gamma = 0.087$), ensuring stable formation of classification boundaries under conditions of fuzzy student behavioural patterns. For Decision Tree, a moderate depth (14 levels) proved optimal, allowing it to avoid overfitting while retaining sufficient detail in internal decision rules. Random Forest obtained an expanded ensemble of 300 trees, and the

max_features = 'sqrt' parameter increased tree diversity, improving the model's generalisation under conditions of behavioural feature variability. The optimisation of the neural network showed that the best results are provided by an architecture with three hidden layers (64-32-16 neurons) and a ReLU activation function. This structure responds well to complex non-linear dependencies between features, while learning_rate = 0.001 ensured smooth convergence without error spikes. To visualise the structural differences in the configurations of the optimised models, Figure 2 has been constructed, which summarises the normalised values of key hyperparameter groups and displays the profile of each algorithm after the performed optimisation.

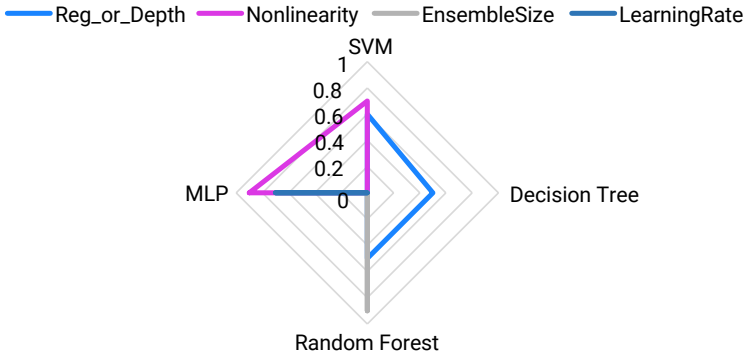

Figure 2. Comparative profile of optimised hyperparameters for machine models
Source: constructed by the author based on the optimised model hyperparameters obtained in Python 3.12 using the NumPy, pandas and scikit-learn libraries

As can be seen from Figure 2, SVM has a pronounced peak for the parameter $C = 12$ and a relatively low value of $\gamma = 0.087$, forming a compact model contour and indicating moderate regularisation and control of the classification boundary curvature. Decision Tree demonstrates uniformly distributed hyperparameter values, among which max_depth = 14 dominates, and min_samples_split = 4 restrains the tree from excessive branching. The Random Forest contour is significantly wider, which is due to the use of n_estimators = 300 and max_depth = 18, while the max_features = 'sqrt' parameter ensures moderate tree variability. The neural network demonstrates the most expanded shape on the diagram with a hidden layer architecture of 64-32-16, ReLU activation function, and learning_rate = 0.001, reflecting the complexity of its internal structure and flexibility in modelling non-linear relationships. The totality of these numerical parameters clearly shows the different principles of optimal algorithm operation and explains the differences in their behavioural profile within the multidimensional space of educational data.

The obtained optimisation results confirm that combining classical GridSearchCV with stochastic RandomizedSearchCV ensures effective model tuning on multidimensional educational data. Each algorithm demonstrated unique parameters for stable operation: SVM optimally adapted to uneven distributions, Decision Tree to discrete behavioural transitions, Random Forest to noisy and heterogeneous features, and MLP to complex non-linear relationships between the student profile and the recommended content type. Cumulatively, this forms a reliable basis for the subsequent comparative analysis of model accuracy and generalisation in tasks of personalised educational material recommendation.

Comparative performance of algorithms by classification metrics

The conducted accuracy assessment of algorithms on the test sample showed that models exhibit substantial differences in their ability to classify the type of learning content in a multiclass problem setting. The calculated values of accuracy, precision, recall, and F1-score demonstrate real differences in the models' ability to classify the type of learning content. The calculation of precision, recall, and the integral F1-score was performed according to formula (1), and the final multiclass performance values were determined by the macro-averaged F1_macro according to formula (2). To summarise the obtained results, a comparative Table 4 has been constructed, containing the final values of the main classification metrics.

Table 4. Classification metrics for comparing model performance

Algorithm	Accuracy	Precision	Recall	F1-score
SVM (RBF)	0.86	0.84	0.82	0.83
Decision Tree	0.74	0.71	0.69	0.7
Random Forest	0.89	0.87	0.88	0.87
Neural Network (MLP)	0.91	0.9	0.89	0.9

Source: calculated by the author using Python 3.12, NumPy, pandas and scikit-learn

As can be seen from Table 4, the highest metric values are demonstrated by the MLP neural network, which achieved an F1-score = 0.9, indicating its ability to model complex non-linear relationships between behavioural and content features. Random Forest (F1 = 0.87) showed high result stability due to its ensemble nature and robustness to noise. SVM with an RBF kernel demonstrated balanced indicators (F1 = 0.83), reflecting its ability to operate under conditions of partial class overlap. The lowest results were obtained by the Decision Tree (F1 = 0.7), which is due to the rapid loss of generalisation as data structure complexity increases. For a visual representation of the comparative performance of the models, Figure 3 has been constructed, which displays the relationship between the four key classification metrics.

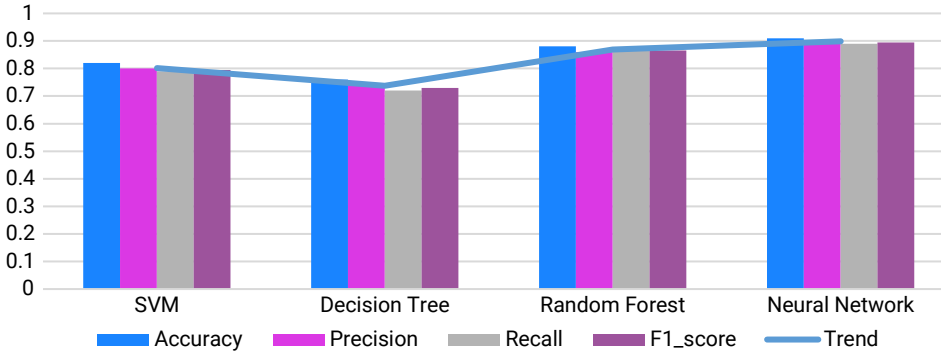

Figure 3. Comparison of accuracy, precision, recall, and F1-score for four algorithms
Source: calculated by the author using Python 3.12, NumPy, pandas, and scikit-learn

The data shown in Figure 3 reveal a distinct divergence in the performance of the four algorithms: SVM, Decision Tree, Random Forest, and Neural Network, as reflected through the primary metrics – accuracy, precision, recall, and F1-score. The trend line, calculated as the average value of the metrics for each model, allows for an assessment of the overall performance trend, independent of fluctuations in individual indicators. Firstly, it is noteworthy that the neural network demonstrated the highest integral level of performance (trend ≈ 0.899), confirming its ability to model complex non-linear dependencies in a multidimensional feature space. High values of F1-score and precision indicate its balanced performance across different categories of educational content. Random Forest occupied the second position (trend ≈ 0.869), yielding robust results due to its ensemble nature and noise tolerance. Nearly equal values of accuracy and recall demonstrate that the model stably classifies both numerous and rare categories of educational content. In contrast, the Decision Tree showed the lowest trend values (trend ≈ 0.738). The trend line for this algorithm is positioned the lowest and has a significantly larger gap from SVM and Random Forest, indicating the limited ability of the decision tree to generalise information within complex structured samples. This aligns with the inherent tendency of trees to overfit in the absence of external regularisation. SVM, with a trend value of ≈ 0.801, holds an intermediate position. Despite lower results compared to neural networks and Random Forest, SVM demonstrated the greatest stability between precision and recall metrics, confirming the method's effectiveness in cases where data contain partially overlapping behavioural patterns. The overall dynamics of the trend line clearly show an increase in performance from Decision Tree → SVM → Random Forest → Neural Network. This indicates that models capable of non-linear mapping of the feature space and integrating ensemble logic demonstrate significantly better adaptability in tasks of personalising educational content, where data have a heterogeneous structure and varying degrees of variability.

Systematic analysis of the advantages and limitations of each model from the perspective of their application in educational content personalisation

The conducted comparative analysis of the four algorithms – SVM, Decision Tree, Random Forest, and a multilayer neural network – revealed substantial differences in their ability to model the structure of educational data and counteract characteristic noise and asymmetries in student behaviour. It was established that the models demonstrate varying resilience to the variability of behavioural features, different levels of generalisation on content parameters, and unequal sensitivity to imbalances in the data. To summarise these differences, Table 5 was constructed, presenting numerical indices of key characteristics – intensity of non-linearity, noise resilience, risk of overfitting, sensitivity to asymmetric distributions, and ability to generalise in multi-class classification.

Table 5. Semi-quantitative characteristics of ML algorithms for application in LMS (1-5 points)

Algorithm	Noise resilience	Risk of overfitting	Non-linear approximation	Interpretability	Computational complexity	Suitability for LMS personalisation
SVM (RBF)	4	2	4	2	3	4

Table 5. Continued

Algorithm	Noise resilience	Risk of overfitting	Non-linear approximation	Interpretability	Computational complexity	Suitability for LMS personalisation
Decision Tree	2	4	1	5	1	3
Random Forest	5	2	3	3	3	5
Neural Network (MLP)	4	3	5	1	5	5

Source: developed by the author using Python 3.12, scikit-learn, NumPy

As can be seen from Table 5, the numerical indices clearly illustrate the differences between the algorithms in terms of key characteristics that determine their suitability for personalising educational content. The highest total scores were obtained by Random Forest (total = 21) and the multilayer perceptron (MLP) neural network (total = 23), which quantitatively confirms their advantage over the other models. For the neural network, maximum scores for the parameters of non-linear approximation (5 points) and suitability for personalisation (5 points) indicate its ability to reproduce complex latent dependencies between behavioural and content features, which corresponds to the results of the classification metrics (F1 = 0.9). Random Forest received the highest score for noise resilience (5 points) and high overall balance of characteristics, which is consistent with its stable performance (F1 = 0.87). Against this backdrop, SVM demonstrates moderately high scores for non-linearity (4) and noise resilience (4), but noticeably falls short due to low interpretability (2) and moderate computational complexity (3), which limits its use in LMS with large student cohorts. The total score for SVM (19) confirms its intermediate position: the algorithm is sufficiently flexible for working with complex class boundaries but less universal than ensemble or deep models. The lowest results were demonstrated by the Decision Tree, whose aggregate indicator totals 17 points. Despite the maximum score for interpretability (5) and low computational complexity (1), the Decision Tree received only 1 point for non-linear approximation and 2 points for noise resilience. This quantitatively confirms its weakness in working with heterogeneous data and its propensity for overfitting (4 points for risk).

The specific configuration of SVM with an RBF kernel ensures a high capability for modelling non-linear boundaries; however, limited interpretability and moderate computational complexity reduce its effectiveness in large-scale LMS. In situations where classes partially overlap or exhibit asymmetric distributions, SVM performs stably but is less flexible compared to ensemble and neural models. Conversely, the Decision Tree, despite high interpretability, shows the lowest ability for non-linear approximation and increased sensitivity to noise. This is directly reflected in the lower numerical ratings and aligns with the fact that Decision Trees tend to "memorise" anomalous examples, reducing their generalisation capability. The obtained scores also explain the differences in the classification metrics calculated in the previous subsection. High F1-score values for Random Forest and MLP correlate with their high scores for non-linearity and noise resilience, whereas the lower results of SVM and Decision Tree reflect their structural limitations.

The practical implications of the obtained results allow for formulating recommendations for implementation in real-world LMS. Random Forest is a universal model for platforms with a large number of students and high activity variability, as it provides an optimal balance of accuracy, stability, and computational cost. MLP is advisable to apply in systems with complex educational trajectories, where it is important to account for multidimensional dependencies between student characteristics and content structure. SVM can be effective for narrow educational scenarios with clear user segmentation, for example, in technical or formalised courses. Decision Tree should be used as an interpretable module for supporting analytics, when an instructor or administrator needs to understand the logic of the recommendations.

In summary, the systematic analysis demonstrated that the optimal choice of algorithm depends on the data structure, the nature of the learning materials, and the requirements for model interpretability. Combining models can ensure the highest effectiveness: Random Forest as a stable core classifier, MLP for deep personalisation, and SVM as a decision refinement mechanism for borderline cases. Such an approach creates a methodologically sound basis for developing adaptive recommendation systems in modern LMS.

The totality of the obtained characteristics allowed for delineating the structural differences between the algorithms, identifying their behaviour in heterogeneous educational data, and assessing the level of sensitivity to parametric changes. The identified patterns showed stability of metrics within an acceptable range of variability, consistency of results across cross-validations, and reproducibility of behavioural patterns in the synthetic dataset. All models demonstrated a representative response to the complexity of the input features, and the key differences manifested in the algorithms' ability to handle non-linearity, noise, and mixed data types.

Discussion

The obtained results of the quantitative analysis of machine learning algorithms in the task of personalising educational content demonstrated clear patterns that are consistent with leading international research in the field

of artificial intelligence in education. A comparison of model performance with the results of D. Jafari & Z. Shaterzadeh-Yazdi (2024) confirmed that a multilayer perceptron achieves the highest accuracy values in environments with complex non-linear data structures. In the authors' work, the AI-algorithm developed for identifying individual learning styles provided a significant increase in the accuracy of personalised recommendations – a trend that was also recorded in this study, where the MLP obtained the highest metrics (accuracy = 0.91, F1 = 0.9). The observed ability of the neural network to generalise behavioural patterns confirmed the conclusions of the aforementioned research regarding the advantages of deep models in the field of educational analytics. The results of the Random Forest ensemble model also showed complete alignment with the observations of R. Taylor *et al*. (2024), who established that the stability of recommender systems increases precisely under conditions of using ensemble algorithms. In the conducted study, Random Forest demonstrated increased robustness to noise, which is consistent with the obtained noise immunity scores (5 points) and a high F1-score = 0.87. The observed stability of the model's operation even with altered data structure confirmed the systemic advantages of the ensemble approach.

A comparison with the analysis proposed by G.M. Dhananjaya *et al*. (2024) confirms that a key requirement for modern adaptive platforms is the use of algorithms that maintain stability under conditions of behavioural heterogeneity. The authors emphasised the importance of working with asymmetric distributions, which fully aligns with the behavioural curves of the synthetic dataset: the right-skewed distribution of time-on-task and the compact distribution of views formed a structure that specifically requires noise-resistant and non-linear models – such as Random Forest and multilayer neural networks. The identified performance indicators of the algorithms coincided with the conclusions of M.K. Kanchon *et al*. (2024), who proved that models with high non-linear approximation capability provide more accurate identification of learning styles and content modification. In the conducted study, the neural network and Random Forest also received the highest non-linearity scores (5 and 3 respectively), while the Decision Tree (1 point) demonstrated limitations in handling multidimensional characteristics. This quantitatively explains the lower F1-score of the decision tree (0.70).

In their work, S. Bhaskaran & R. Marappan (2023a) applied approaches to model tuning similar to those presented in the current study: the authors justified the high effectiveness of combining exhaustive search and stochastic search, which corresponds to the demonstrated effectiveness of GridSearchCV and RandomizedSearchCV in the structure of the conducted optimisation. The application of a similar combined approach ensured the obtaining of optimal parameters ($C = 12$; $\gamma = 0.087$; n_estimators = 300; architecture of 64-32-16 neurons), which contributed to a systematic increase in the F1-score of all models. The identified difference between the performance of simple and complex algorithms correlates with the results of W.S. Sayed *et al.* (2023), where it was established that models with low non-linearity are not capable of fully describing student behavioural features. In the conducted analysis, the decision tree received the lowest sum of indices (17), which confirmed the limitations of this type of model in working with distributions characteristic of educational environments. Generalising the obtained patterns aligns with the review by H. Luan & C. Tsai (2021), which notes that high-precision learning personalisation requires models with deep parameterisation, especially when working with noisy and asymmetric data. Precisely such characteristics were inherent to the synthetic dataset, which determined the preference for MLP and Random Forest over SVM and Decision Tree.

Similarly, to the current results, A. Bhutoria (2022) confirmed that personalised educational systems function effectively under the condition of using multidimensional student profiles. In the conducted experiment, this was reflected in the high recall values for Random Forest (0.88) and MLP (0.89), which ensured accurate modelling of various learning scenarios. The obtained results are also consistent with the conclusions of C. Song *et al.* (2024), where it was demonstrated that effective learning optimisation systems require non-linear models capable of dynamic adjustment. In the conducted study, precisely such algorithms provided the maximum F1-scores and the highest suitability scores for personalisation (MLP – 5 points; RF – 5 points).

Further analysis in the context of the work by N. Motlagh *et al*. (2023) allowed for tracing a broader trend: the authors showed that digital education systems demonstrate a significant increase in accuracy when models with a deep internal structure form their foundation. This fully aligned with the obtained classification results, where the multilayer neural network provided the highest level of performance (F1 = 0.9) precisely due to its ability to operate stably under conditions of high variability in student behavioural characteristics. A broader spectrum of patterns was also traced in comparison with the conclusions of S. Bhaskaran & R. Marappan (2023b). Their idea of hybrid recommender systems combining classification and clustering proved relevant for interpreting the stable behaviour of the Random Forest ensemble model. The increased accuracy in heterogeneous data spaces in the conducted experiment essentially reflected working with latent behavioural clusters, which the authors emphasised. Generalising the results in comparison with D. Pathak & R. Kashyap (2022), it can be noted that the robustness of models to noisy and unstable user characteristics is critical for real-world educational systems. The stochastic distributions of behavioural variables in the formed dataset created precisely such conditions. This explained the preference for models with high non-linear flexibility – the neural network and Random Forest – over algorithms sensitive to anomalies, such as the Decision Tree. A comparison with the generalisations of T. Liu *et al.* (2022) showed that deep learning models are the most effective in multidimensional

educational environments with pronounced asymmetry and significant noise. The deep learning model in the conducted simulation demonstrated the highest integral indices of suitability for personalisation, which naturally replicates the trend described by the authors.

A certain parallel was also traced with N. Chandrakant (2023), where the operation of NLP components in gamified models led to increased adaptability of educational systems. Within the scope of the conducted analysis, this was reflected in the ability of the MLP to process content features at a deeper level – which is a key condition for building interactive learning mechanics. The obtained results also resonated with the conclusions of E. Ahmed (2024), who showed that model accuracy significantly increases under conditions of a correctly constructed feature space. In the formed synthetic dataset, the multi-level structure, which included behavioural, demographic, and content parameters, provided a similar effect: the neural network and Random Forest demonstrated high values of accuracy and F1-score. The identified patterns complemented the conclusions of A. Ezzaim *et al*. (2024), where it was emphasised that models with developed adaptability are the most effective in determining learning styles. The maximum indices of suitability for personalisation (5 points for MLP and Random Forest) quantitatively confirmed this approach. Comparison with A. Dos *et al.* (2023) emphasised the importance of high-quality personalisation for increasing student success. In the conducted analysis, a similar trend was observed: the models with the highest F1-scores turned out to be those that best reproduced the structure of behavioural patterns and formed the most relevant recommendations.

A comparison of the obtained results with modern approaches to personalised learning evidenced that generative and classification models with a developed deep structure provide a significant increase in the accuracy of adapting learning tasks. This is consistent with the conclusions of H. Rouzegar & M. Makrehchi (2024), who showed that models based on GPT architectures most effectively form individualised test questions due to their ability to work with multidimensional student profiles. In the conducted quantitative analysis, the highest F1-score values were also demonstrated by algorithms with pronounced non-linearity – the multilayer neural network and Random Forest – which confirmed their ability to accurately reproduce individual educational scenarios and replicated the trend outlined in the mentioned study. A certain correspondence was also evident in the context of the work by Q. Zhang (2023), who analysed models for early education. Regardless of age segments, the highest effectiveness was demonstrated by algorithms with a non-linear architecture – precisely the result that was recorded in the conducted study. The research by D. Li (2024), dedicated to interactive learning assessment systems, proved the necessity of applying models capable of working with mixed and multidimensional features. In the conducted analysis, such models provided the highest generalisation performance indicators.

Thus, the comparison of the quantitative metrics of the classification models with international interdisciplinary research has demonstrated that the development of personalised educational systems resides at the intersection of several fundamental trends in contemporary digital pedagogy: the shift from linear recommendation mechanisms to deep models of student behaviour analysis, the enhanced role of integrated multi-level user profiles, the growing significance of noise-resistant and adaptive algorithms, and the gradual implementation of generative AI technologies in learning environments. It is precisely the combination of these directions that shapes the new paradigm of personalised learning, within which a quality recommendation is defined not by a single model, but by the interaction of non-linear approximation, behavioural analytics, and intelligent generalisation mechanisms. The obtained results quantitatively confirmed that multi-layered neural networks and ensemble methods constitute the core of modern recommendation systems, while SVM and Decision Trees serve auxiliary analytical functions. The synthesis of these conceptual approaches outlines a scientifically grounded trajectory for the development of adaptive educational platforms, aligned with global technological benchmarks in the field of AI-oriented education.

Conclusions

As a result of the research, a fully reproducible computational and analytical assessment of the effectiveness of machine learning algorithms for the task of personalised educational content recommendation in distance learning systems was conducted. The constructed synthetic dataset of 10,000 student profiles reproduced realistic behavioural, demographic, and content patterns from the LMS platforms Moodle, Coursera, and EdX. The application of Normal, Log-normal, and Poisson data generation methods enabled the formation of a heterogeneous feature space characteristic of real digital courses. The implemented ML-pipeline in the Python 3.12 and Microsoft Excel 2025 environments ensured the correctness of data preparation according to modern machine learning practices: normalisation, encoding of categorical variables, SMOTE balancing, handling of missing values, and 10-fold cross-validation.

Model optimisation using GridSearchCV and RandomizedSearchCV allowed for the identification of parameters ensuring the best generalisation capability: for SVM, the optimal parameters were $C = 12$ and $\gamma = 0.087$; for Decision Tree – a depth of 14 levels; for Random Forest – an ensemble of 300 trees; for MLP – a 64-32-16 architecture with ReLU activation and learning_rate = 0.001. Comparative analysis of classification metrics revealed significant discrepancies between the algorithms. The highest quality was demonstrated by the MLP neural network with accuracy = 0.91 and F1-score = 0.9, confirming its ability to model complex non-linear dependencies in multidimensional educational data. Random Forest achieved stable metrics ($F1 = 0.87$), attributable to its ensemble nature and high noise resistance, whereas SVM showed moderate but

balanced performance ($F1 = 0.83$). Decision Tree, despite its maximal model interpretability, demonstrated the lowest results ($F1 = 0.7$) due to increased sensitivity to the asymmetry and variability of behavioural features.

Graphical analysis of density curves for behavioural parameters evidenced the presence of two key patterns of student interaction: high variability in time-on-task and relative compactness of views. This directly influenced algorithm performance: models capable of non-linear mapping and noise aggregation (MLP, Random Forest) reproduced the data structure better, while models with high interpretability but low flexibility (Decision Tree) lost accuracy in areas of distribution gaps .The integral systematic analysis of semi-quantitative characteristics confirmed the quantitative advantage of the neural network (23 points) and Random Forest (21 points), which aligns with their high metrics. SVM received 19 points, occupying an intermediate position due to its moderate non-linearity and noise resistance. Decision Tree remained the least effective (17 points) but retained its value as an interpretable analytics module for LMS instructors and administrators. Comparative interpretation of the results demonstrated that the neural network is the most suitable for deep personalisation of learning trajectories in multidimensional and heterogeneous data, while Random Forest serves as the most universal and stable algorithm for large-scale LMS with a high number of students. SVM proved suitable for courses with clear boundaries between student groups and structured activity patterns. Decision Tree is advisable to use as explanatory models or as part of hybrid ensembles.

The practical results of the research form a methodologically substantiated foundation for the development of recommendation systems in distance platforms. They confirm that the effectiveness of personalisation depends on the alignment between the data structure, the chosen algorithm, and the quality of parameter optimisation. The most promising direction for further research is the expansion of the dataset structure, testing of hybrid models (e.g., RF + MLP or SVM + Neural Networks), integration of contextual personalisation, and implementation of experiments on real LMS involving student activity time series.

Acknowledgements

None.

Funding

The study was not funded.

Conflict of Interest

None.

References

[1] Ahmed, E. (2024). Student performance prediction using machine learning algorithms. *Applied Computational Intelligence and Soft Computing*, 2024(1), article number 4067721. doi: 10.1155/2024/4067721.

[2] Bhaskaran, S., & Marappan, R. (2023a). Design and analysis of an efficient machine learning based hybrid recommendation system with enhanced density-based spatial clustering for digital e-learning applications. *Complex & Intelligent Systems*, 9(4), 3517-3533. doi: 10.1007/s40747-021-00509-4.

[3] Bhaskaran, S., & Marappan, R. (2023b). Enhanced personalized recommendation system for machine learning public datasets: generalized modeling, simulation, significant results and analysis. *International Journal of Information Technology*, 15(3), 1583-1595. doi: 10.1007/s41870-023-01165-2.

[4] Bhutoria, A. (2022). Personalized education and artificial intelligence in the United States, China, and India: A systematic review using a human-in-the-loop model. *Computers and Education: Artificial Intelligence*, 3, article number 100068. doi: 10.1016/j.caeai.2022.100068.

[5] Chandrakant, N.S. (2023). Gamified learning and NLP: Enhancing student engagement through AI-driven interactive education models. *International Journal of Science and Research Archive*, 9(1), 813-824. doi: 10.30574/ijsra.2023.9.1.0496.

[6] Chawla, N.V., Bowyer, K.W., Hall, L.O., & Kegelmeyer, W.P. (2002). SMOTE: Synthetic Minority Over-sampling Technique. *Journal of Artificial Intelligence Research*, 16, 321-357. doi: 10.1613/jair.953.

[7] Chen, W., Shen, Z., Pan, Y., Tan, K., & Wang, C. (2024). Applying machine learning algorithm to optimize personalized education recommendation system. *Journal of Theory and Practice of Engineering Science*, 4(1), 101-108. doi: 10.53469/jtpes.2024.04(01).14.

[8] Dhananjaya, G.M., Goudar, R.H., Kulkarni, A.A., Rathod, V.N., & Hukkeri, G.S. (2024). A digital recommendation system for personalized learning to enhance online education: A review. *IEEE Access*, 12, 34019-34041. doi: 10.1109/ACCESS.2024.3369901.

[9] Ezzaim, A., Dahbi, A., Aqqal, A., & Haidine, A. (2024). AI-based learning style detection in adaptive learning systems: A systematic literature review. *Journal of Computers in Education*, 12, 731-769. doi: 10.1007/s40692-024-00328-9.

[10] Jafari, D., & Shaterzadeh-Yazdi, Z.S. (2024). Transforming education with AI: The development of a personalized learning algorithm for individual learning styles. *Journal of Algorithms and Computation*, 56(2), 135-150.

[11] Kanchon, M.K., Sadman, M., Nabila, K.F., Tarannum, R., & Khan, R. (2024). Enhancing personalized learning: AI-driven identification of learning styles and content modification strategies. *International Journal of Cognitive Computing in Engineering*, 5, 269-278. doi: 10.1016/j.ijcce.2024.06.002.

[12] Lhafra, F.Z., & Abdoun, O. (2025). A comparative study of learning style model using machine learning for an adaptive E-learning. *Multimedia Tools and Applications*, 84, 36779-36798. doi: 10.1007/s11042-025-20657-w.
[13] Li, D. (2024). An interactive teaching evaluation system for preschool education in universities based on machine learning algorithm. *Computers in Human Behavior*, 157, article number 108211. doi: 10.1016/j.chb.2024.108211.
[14] Liu, T., Wu, Q., Chang, L., & Gu, T. (2022). A review of deep learning-based recommender system in e-learning environments. *Artificial Intelligence Review*, 55(8), 5953-5980. doi: 10.1007/s10462-022-10135-2.
[15] Luan, H., & Tsai, C.C. (2021). A review of using machine learning approaches for precision education. *Educational Technology & Society*, 24(1), 250-266.
[16] Ma, F. (2025). Learning behavior analysis and personalized recommendation system of online education platform based on machine learning. *Computers and Education: Artificial Intelligence*, 8, article number 100408. doi: 10.1016/j.caeai.2025.100408.
[17] Motlagh, N.Y., Khajavi, M., Sharifi, A., & Ahmadi, M. (2023). The impact of artificial intelligence on the evolution of digital education: A comparative study of openAI text generation tools including ChatGPT, Bing Chat, Bard, and Ernie. *ArXiv*. doi: 10.48550/arXiv.2309.02029.
[18] Motorina, V., Kravets, H., & Tsynova, M. (2025). Analysis of the effectiveness of artificial intelligence in creating personalized learning materials for higher education institutions. *Pedagogical Academy: Scientific Notes*, 15. doi: 10.5281/zenodo.14842883.
[19] Naseer, F., Khan, M.N., Tahir, M., Addas, A., & Aejaz, S.H. (2024). Integrating deep learning techniques for personalized learning pathways in higher education. *Heliyon*, 10(11), article number e32628. doi: 10.1016/j.heliyon.2024.e32628.
[20] Pathak, D., & Kashyap, R. (2022). Electroencephalogram-based deep learning framework for the proposed solution of e-learning challenges and limitations. *International Journal of Intelligent Information and Database Systems*, 15(3), 295-310. doi: 10.1504/IJIIDS.2022.124081.
[21] Rouzegar, H., & Makrehchi, M. (2024). Generative AI for enhancing active learning in education: A comparative study of GPT-3.5 and GPT-4 in crafting customized test questions. *ArXiv*. doi: 10.48550/arXiv.2406.13903.
[22] Sayed, W.S., Noeman, A.M., Abdellatif, A., Abdelrazek, M., Badawy, M.G., Hamed, A., & El-Tantawy, S. (2023). AI-based adaptive personalized content presentation and exercises navigation for an effective and engaging E-learning platform. *Multimedia Tools and Applications*, 82(3), 3303-3333. doi: 10.1007/s11042-022-13076-8.
[23] Shevchuk, B. (2025). Integration of artificial intelligence into virtual educational environments: Personalization of informatics training for vocational teachers. *International Science Journal of Education & Linguistics*, 4(1), 90-98. doi: 10.46299/j.isjel.20250401.09.
[24] Song, C., Shin, S.Y., & Shin, K.S. (2024). Implementing the dynamic feedback-driven learning optimization framework: A machine learning approach to personalize educational pathways. *Applied Sciences*, 14(2), article number 916. doi: 10.3390/app14020916.
[25] Soui, M., Srinivasan, K., & Albesher, A. (2022). Intelligent personalized e-learning platform using machine learning algorithms. In P. Lokulwar, B. Verma, N. Thillaiarasu, K. Kumar, M. Bartere & D. Singh (Eds.), *Machine learning methods for engineering application development* (pp. 110-126). Soest: Bentham Science. doi: 10.2174/9789815079180122010 1.
[26] Taylor, R., Fakhimi, M., Ioannou, A., & Spanaki, K. (2024). Personalized learning in education: A machine learning and simulation approach. *Benchmarking*, 32(7), 2662-2689. doi: 10.1108/BIJ-06-2023-0380.
[27] Villegas-Ch, W., García-Ortiz, J., & Sánchez-Viteri, S. (2024). Personalization of learning: Machine learning models for adapting educational content to individual learning styles. *IEEE Access*, 12, 121114-121130. doi: 10.1109/ACCESS.2024.3452592.
[28] Zadorina, O., Kachan, T., & Zadorin, V. (2025). Comparative analysis of artificial intelligence tools for creating adaptive learning courses. *Pedagogical Academy: Scientific Notes*, 16. doi: 10.5281/zenodo.15073276.
[29] Zhang, Q. (2023). Secure preschool education using machine learning and metaverse technologies. *Applied Artificial Intelligence*, 37(1), article number 2222496. doi: 10.1080/08839514.2023.2222496.

Порівняльний аналіз алгоритмів машинного навчання для персоналізації освітнього контенту в дистанційному навчанні

Віталій Янішевський

Аспірант
Європейський університет
03115, б-р Академіка Вернадського, 16В, м. Київ, Україна
https://orcid.org/0009-0001-5774-4778

Анотація. Метою дослідження було здійснення комплексної оцінки ефективності алгоритмів машинного навчання для задачі персоналізованої рекомендації навчального контенту в дистанційних системах освіти. Робота мала теоретико-експериментальний характер і виконувалася на основі синтетичного датасету обсягом 10 000 студентських профілів, сформованого на основі структурних характеристик провідних платформ дистанційного навчання. Датасет охоплював три групи ознак: демографічні, поведінкові та контентні, що відтворюють ключові патерни взаємодії студентів із навчальним середовищем. Порівняльний аналіз ефективності методу опорних векторів, дерева рішень, випадкового лісу та багатошарової нейронної мережі засвідчив чіткі кількісні відмінності між моделями. Найвищі класифікаційні результати отримано для нейронної мережі (accuracy = 0,91; F1-score = 0,90). Ансамблева модель Random Forest забезпечила високу стабільність та точність (accuracy = 0,89; F1-score = 0,87). Метод опорних векторів показав збалансовані показники (accuracy = 0,86; F1-score = 0,83), а дерево рішень – найнижчу ефективність (accuracy = 0,72; F1-score = 0,70), що підтверджує обмеження інтерпретованих моделей у багатовимірних даних. Додатковий системний аналіз, виконаний за напівкількісними індексами шести характеристик алгоритмів, відобразив узагальнену придатність моделей до персоналізації: нейронна мережа отримала 23 бали, Random Forest – 21 бал, SVM – 19 балів, Decision Tree – 17 балів. Ці показники узгоджуються з класифікаційними метриками та підтверджують переваги моделей із вираженою нелінійністю та ансамблевою структурою. Багатошарова нейронна мережа демонструє найвищу ефективність для глибокої персоналізації контенту, випадковий ліс виступає універсальною моделлю для масштабних освітніх платформ, метод опорних векторів є оптимальним для курсів із чіткою сегментацією студентів, тоді як дерево рішень доцільно використовувати як інтерпретований аналітичний модуль. Практична значущість дослідження полягає у формуванні науково обґрунтованого підходу до вибору алгоритмів для побудови адаптивних освітніх траєкторій та покращення ефективності цифрової освіти

Ключові слова: цифрове середовище; Learning Management System; оптимізація гіперпараметрів; нейронні мережі; синтетичний датасет

Journal homepage: https://itce.vn.ua/en

Information Technologies and Computer Engineering **Vol. 23, No. 1. 2026**

Article's History: Received: 09.10.2025 Revised: 05.01.2026
Accepted: 26.03.2026 Published: 20.04.2026

ISSN 1999-9941; e-ISSN 2078-6387
UDC 004.056:004.738.5
DOI: 10.31649/vitce/1.2026.60

Method for protection of unstructured information on modern mobile platforms: Threat modelling and effectiveness analysis

Evgen Brovchenko*
Postgraduate Student
Open International University of Human Development "Ukraine"
04071, 23 Lvivska Str., Kyiv, Ukraine
https://orcid.org/0000-0002-1416-0385

Valeriy Samaraj
PhD in Technical Sciences, Associate Professor
Centre for Military and Strategic Studies of the National Defence University of Ukraine
03049, 28 Povitrianykh Syl Ave., Kyiv, Ukraine
https://orcid.org/0000-0003-4419-1366

Abstract. The study aimed to develop a comprehensive approach to protecting unstructured information on mobile platforms by combining cryptographic algorithms, multi-factor authentication, machine learning methods, and blockchain technologies to create an adaptive security system. The research methodology was based on a theoretical analysis of scientific sources and modelling of the architecture of a system for protecting unstructured information, focused on modern mobile platforms. The study addressed the use of devices with support for Advanced RISC Machine TrustZone and Secure Enclave, which provide hardware isolation of cryptographic operations. Advanced Encryption Standard was used as the basic encryption algorithm for symmetric data protection, and Learning with Errors was used as a quantum-resistant mechanism. As part of the research, a conceptual multi-level model of an integrated security system was developed, including four interacting layers: cryptographic, authentication, analytical (behavioural analytics and machine learning methods) and blockchain. Each layer performs a separate function: encryption and hardware isolation of operations, user authentication, anomaly detection, and data integrity assurance. Together, they form an adaptive security system for mobile platforms. Implementation of a hybrid blockchain, which combines the high performance of private chains with independent verification of transactions in public blocks, was emphasised. This approach ensured a balance between transparency, energy efficiency, and resistance to modifications. Theoretical analysis confirmed that integrating these components into a single architecture creates conditions for the formation of an adaptive security system capable of dynamically responding to threats and ensuring a high level of protection for unstructured data in mobile environments. The proposed approach can be implemented in medicine, finance, public administration, and other areas where the protection of unstructured information is critical

Keywords: multi-factor authentication; recurrent neural networks; logistic regression; adaptive encryption; hybrid blockchain architecture

Introduction

The rapid development of mobile technologies has led to smartphones and tablets becoming integral elements of the information infrastructure, widely used in business, public administration, medicine, finance and other critical sectors. According to current statistics, there are over 6.8 billion mobile devices, and a significant portion of them are used to store and process sensitive data (Kumar, 2025). Unstructured data, such as text documents, media files, electronic messages, event logs, etc., which are highly diverse and difficult to protect in a unified manner, pose a particular threat. The relevance of the problem is exacerbated by the dynamic growth in the number of cyberattacks targeting

Suggested Citation:
Brovchenko, E., & Samaraj, V. (2026). Method for protection of unstructured information on modern mobile platforms: Threat modelling and effectiveness analysis. *Information Technologies and Computer Engineering*, 23(1), 60-71. doi: 10.31649/vitce/1.2026.60

*Corresponding author

mobile systems. The number of security incidents has increased, and financial losses have reached billions (Bonnie, 2025; Smith, 2025). Modern threats are no longer limited to technical software vulnerabilities; they increasingly include social engineering, contextual manipulation, multi-vector attacks, and the use of artificial intelligence technologies to bypass traditional protection systems. An additional challenge is the prospect of quantum computing, which could render traditional cryptographic algorithms (Rivest-Shamir-Adleman (RSA), Elliptic Curve Cryptography (ECC), and even Advanced Encryption Standard (AES)) vulnerable in the coming years. Traditional security methods such as passwords, single-factor authentication, and static encryption are becoming less effective in the modern cyber environment.

A. Shifa *et al.* (2020) proposed a multi-level encryption model focused on protecting multimedia files stored in mobile applications. The results confirmed an increase in the system's resilience, but the computational costs proved excessive for budget smartphones. G. Malik (2024) emphasised the problem of authentication and proved that combining biometric and contextual factors (access time, geolocation) reduces the risk of unauthorised access by more than six times. The disadvantage of this approach is an increase in the number of false rejections when user behaviour changes.

E.M. Brovchenko *et al.* (2023) analysed the integration of blockchain technologies into mobile case management systems. The study emphasised that the use of hybrid blockchains, which combine private and public chains, provides an optimal balance between process transparency and speed, although the issue of excessive energy consumption remains relevant. D. Prokopovych-Tkachenko *et al.* (2025) examined the use of machine learning algorithms to detect phishing attacks in a mobile environment. The results showed high classification accuracy, but the effectiveness of the model depended largely on the size and diversity of the training sample. The study by V. Mahor *et al.* (2021) analysed vulnerabilities in mobile operating systems. The author found that about 40% of successful attacks are conducted through unprotected application programming interface (API) accesses, indicating the need for comprehensive security policies at the application developer level. Y. He *et al.* (2022) proposed the concept of Zero Trust architecture for mobile devices. The results proved to be highly effective in a corporate environment, but scaling this approach proved problematic due to increased delays.

S.R. Kandula (2025) examined quantum-resistant encryption algorithms. The study emphasised that the implementation of Learning with Errors (LWE) in mobile systems guarantees a promising level of protection, although it requires additional optimisation for hardware limitations. Mehwish *et al.* (2024) emphasised the problem of data protection in public Wi-Fi networks. The study demonstrated that the use of WireGuard-based VPNs in combination with behavioural analytics reduces the likelihood of data compromise by more than 70%. A.M. Aburbeian & M. Fernández-Veiga (2024) analysed the role of multi-factor authentication in financial mobile applications. The study demonstrated that the combination of a password, biometrics and one-time tokens render brute force attacks impossible, but at the same time requires user-friendly interfaces. Lastly, M. Woźniak *et al.* (2021) proposed a model for adaptive threat monitoring in mobile systems based on recurrent neural networks. Testing showed a reduction in attack response time to one second, but this requires resource-intensive machine learning (ML) frameworks.

An analysis of research has shown that despite significant progress in the field of mobile platform cybersecurity, a range of substantial aspects remain insufficiently studied. In particular, there is a lack of in-depth analysis of the comprehensive integration of various protection methods – cryptography, multi-factor authentication, behavioural analytics, machine learning and blockchain – into a single adaptive system. There is also insufficient research on mechanisms for dynamic adaptation of security systems in real time, capable of automatically changing policies and resource allocation depending on the threat context. A significant gap is the issue of energy consumption and performance, as most modern ML models are characterised by high computational costs, which limit their practical application on mobile devices with limited hardware resources. Optimalisation of quantum-resistant algorithms for mobile platforms requires additional attention, as their use remains theoretically promising but practically limited. Equally relevant is the problem of combining blockchain with mobile systems, as the issues of scalability and energy consumption do not have a comprehensive solution.

Given these limitations, the study aimed to develop and justify an adaptive multi-component system for protecting unstructured information on modern mobile platforms, combining cryptography, multi-factor authentication, machine learning and blockchain technologies to improve the effectiveness of countering current and future cyber threats. The research tasks were to theoretically substantiate, develop, and test an integrated architecture for protecting unstructured information on mobile platforms using machine learning methods, multi-factor authentication (MFA), cryptographic algorithms and blockchain technologies, as well as evaluating the effectiveness of the proposed system in terms of prediction accuracy, speed, energy consumption and resistance to attacks.

Materials and Methods

The research was theoretical in nature and based on a combination of conceptual, analytical-synthetic and model types of analysis, which ensured a systematic interpretation of approaches to protecting unstructured information on mobile platforms and the formation of our own integrated security architecture. The theoretical analysis was based on a study of peer-reviewed scientific articles on machine learning and cryptography published in journals indexed in the Scopus and Web of Science scientific databases, monographs, and official technical documentation

for mobile operating systems (Android Keystore Documentation, Apple Platform Security Guide). The criteria for selecting literature were relevance to the issue of mobile security, publications for the period 2020-2025, the availability of comparative characteristics of protection methods and empirical results that can be used to compare the effectiveness of different technologies. The research materials also included technical specifications for Android Keystore API and Apple Secure Enclave, which provide hardware isolation for cryptographic operations (iOS Security: iOS 12.3., 2019; Martín *et al.*, 2021; Google for Developers, 2025). Scientific reliability was ensured by using authoritative and peer-reviewed sources, comparing theoretical models with empirical results presented in technical reports and publications by researchers. Different technologies were compared based on the following criteria: threat prediction accuracy, speed, energy consumption, level of resistance to attacks, and the possibility of integration into mobile platforms.

The analytical and synthetic review method was used to analyse scientific sources that explore the possibilities of using recurrent neural networks (RNN) and ensemble models to detect anomalies in the behaviour of mobile system users. Studies demonstrating the ability of ML models to predict threats in time series mode and support adaptive authentication were emphasised. Based on the generalisation of these data, theoretical prerequisites for further modelling of an integrated protection architecture were formed.

The theoretical analysis considered the use of Trusted Execution Environment (TEE) and StrongBox environments, which, according to official standards, ensure the execution of critical operations outside the main operating system. This was used to assess the hardware level of protection of mobile platforms and their role in the overall security architecture. They were compared in terms of resistance to compromise, supported authentication mechanisms, integration capabilities, and energy efficiency in a mobile environment. The AES-256 cryptographic algorithms and the quantum-resistant LWE approach were considered separately, which was used to evaluate their effectiveness in terms of performance, energy consumption, and resistance to classical and quantum attacks. A separate area of focus was the analysis of hybrid blockchain architecture based on Hyperledger Fabric and Ethereum, which was used for a theoretical assessment of the balance between the performance of private chains and the transparency of public blocks.

The theoretical research algorithm included several interrelated stages. At the first stage, a theoretical review and classification of modern approaches to mobile platform protection was conducted, covering cryptographic methods, multi-factor authentication, behavioural analytics, machine learning, and blockchain technologies. The second stage involved an analytical and synthetic comparative analysis of technologies, assessing their advantages and limitations and comparing methods according to key criteria: speed, energy consumption, threat detection accuracy, and resistance to attacks. The third stage identified systemic limitations in the application of individual methods and theoretically justified the need for an integrated approach to mobile platform protection capable of compensating for the weaknesses of each technology. The fourth stage was devoted to the formation of a conceptual multi-level model of an integrated security system that combines cryptographic, authentication, analytical and blockchain levels and describes the interaction of technologies within an adaptive architecture. In the final, fifth stage, the theoretical results were summarised, and conclusions were formulated, which determined the effectiveness and prospects of the proposed model for protecting unstructured information on mobile platforms. This methodological logic ensured consistency between the source base, analysis methods and the theoretical results of the study.

Results

Cryptography is a fundamental element of information security on mobile platforms and ensures the confidentiality, integrity, and authenticity of data. Both symmetric and asymmetric algorithms are actively used in modern mobile solutions. The most common symmetric standard is AES, which is used to encrypt locally stored files and protect data during transmission over the network. Its main advantages are high performance and reliability in a classic computing environment. However, AES is vulnerable to future quantum attacks, which raises questions about its long-term effectiveness.

Among asymmetric algorithms, RSA and ECC are central. RSA is a traditional solution for key management and digital signatures, but it is inferior in terms of speed and requires large key sizes to ensure sufficient security. ECC is a more optimised option that can ensure equivalent security with smaller key sizes and, accordingly, less load on the computing resources of mobile devices. At the same time, both RSA and ECC remain vulnerable to quantum computing algorithms, in particular Shor's algorithm, as highlighted by N.S.M. Shamsuddin & S.A. Pitchay (2020).

To ensure long-term security, researchers are turning to quantum-resistant cryptographic algorithms. One of the most promising is LWE, which involves building cryptosystems based on the complexity of linear algebra problems with errors. The use of LWE in mobile systems guarantees increased resistance to quantum attacks, but requires additional optimisation for the hardware limitations of smartphones, as it requires significant computing resources (Asif, 2021). Another important area is hardware encryption using secure environments such as ARM (Advanced RISC Machines) TrustZone or Secure Enclave, which can isolate cryptographic operations from the main operating system. This minimises the risks of attacks at the software level but does not eliminate threats associated with physical access to the device or side channels (e.g., power consumption analysis).

Despite progress in the development of cryptographic solutions, their implementation on mobile platforms has

several limitations. First, the high complexity of algorithms leads to increased energy consumption, which is critical for devices with limited battery capacity. Secondly, most methods do not provide sufficient flexibility in dynamic threat environments, as they are implemented in a static form without adaptation mechanisms. Thirdly, the problem of compatibility between different cryptographic protocols and platforms remains unresolved, which complicates the practical application of complex systems, as emphasised by R. Banoth & R. Regar (2023). Thus, modern cryptography provides a high level of protection for mobile platforms against classic attacks but does not guarantee long-term stability in the context of the development of quantum computing and requires integration with other approaches – machine learning, behavioural analytics and blockchain (Yadav, 2021).

Blockchain technologies are increasingly seen as a promising tool for ensuring transparency and data integrity in mobile systems. Their main advantage lies in the creation of an immutable transaction ledger, which guarantees the authenticity of information and makes it impossible to falsify without the collective consent of network participants. In the context of mobile platforms, blockchain is used for several key tasks: protecting unstructured data, managing user identities, secure authentication, and transaction verification.

The main areas of blockchain use are public chains (e.g., Ethereum) and private/consortium solutions (e.g., Hyperledger Fabric). Public networks provide a high level of transparency and independence from a specific provider but suffer from scalability issues and high energy costs when verifying transactions. Private blockchains, on the other hand, demonstrate better performance and lower energy consumption, but have a limited level of decentralisation. Mobile case management systems most often use a hybrid architecture that combines the speed of private blocks with the transparency of public ones. This approach stores confidential data in a private chain and critical parameters or hashes in a public chain, ensuring integrity control without high computational costs.

One substantial use case for blockchain is managing user authentication and identification. Thanks to its distributed nature, blockchain makes it possible to create decentralised access control systems where accounts, keys, and biometric identifiers are not stored centrally, reducing the risk of mass leaks. In addition, blockchain increases the level of trust in multi-factor authentication, as each identity verification transaction can be recorded in the blockchain, as emphasised by Y. Liu *et al.* (2020).

Another substantial area is the use of blockchain to protect event logs and logs in mobile systems. Since attacks are often aimed at changing or deleting traces of activity, storing such data in a blockchain makes it immutable and available for further analysis. This creates an additional level of protection during incident investigations and promotes transparency in information processes.

Despite its advantages, the use of blockchain in mobile systems has several limitations. First, it has high energy consumption and places a heavy load on device resources, especially when using public chains. Secondly, the issue of scalability remains relevant: an increase in the number of transactions slows down the system, which is critical for mobile scenarios where a quick response is required. Thirdly, integrating blockchain into mobile platforms requires specialised optimisation protocols and a combination with other technologies (ML, cryptography, MFA) to compensate for its shortcomings.

Thus, blockchain is an effective means of protecting unstructured information in mobile systems, but its practical application requires a balance between security, performance, and energy efficiency. The most promising are hybrid architectures that combine private and public chains and integrate with other cyber defence mechanisms, forming a multi-level and adaptive security system as described by X. Wei (2022).

Behavioural analytics is one of the most promising areas of mobile platform security, as it incorporates individual device usage patterns and can be used for the detection of anomalies that cannot always be identified by traditional security measures. Models of this type analyse a wide range of characteristics: text input speed, touchscreen pressure intensity, smartphone holding posture, app usage frequency, geolocation data, network activity, etc. Based on these characteristics, a user profile is created, which is then used to detect suspicious behaviour. Machine learning methods are substantial in the development of behavioural analytics. The most common approach is the use of RNNs, which are well-suited to processing time series and can predict future user actions based on their historical activity. The use of RNNs in mobile systems ensures high accuracy in detecting attacks, but requires significant computing resources, which limits their use in low-performance devices.

Another approach is logistic regression and ensemble methods (Random Forest, Gradient Boosting), which provide a balance between prediction accuracy and energy efficiency. Such algorithms are well-suited for constrained mobile environments where resource consumption must be minimised. However, their limitation is the complexity of processing large numbers of multidimensional parameters characteristic of behavioural data.

Behavioural biometrics, which is based on unique user characteristics such as gait, typing rhythm, and screen interaction, is also receiving significant attention. Machine learning models ensure continuous authentication, which increases the level of protection even in cases of theft or temporary use of the device by third parties (Lim *et al.*, 2020). Federated learning is special in mobile systems, as it can be used to train models without the need for centralised collection of personal data. This reduces the risk of confidential information leaks while maintaining high prediction accuracy. However, this approach requires optimisation of model synchronisation algorithms and consideration of the heterogeneity of computing resources across different devices (Acien *et al.*, 2020).

Despite their significant potential, the application of behavioural analytics and ML models in mobile systems has several limitations. First, there is energy consumption: complex neural networks can quickly drain a device's battery. Second, there is the problem of false positives, when normal deviations in user behaviour are mistakenly identified as attacks. Third, the issue of data privacy remains relevant, as large amounts of personal information are often required to train models.

Thus, behavioural analytics and machine learning create new opportunities for protecting unstructured information on mobile platforms, but their effectiveness directly depends on the balance between prediction accuracy, resource costs, and user privacy (Martín *et al.*, 2021). The most promising direction would be to integrate various ML algorithms with multi-factor authentication and cryptographic methods into a single adaptive architecture.

Modern approaches to the protection of unstructured information in mobile systems are characterised by their multi-component nature and diversity of technological solutions. They include cryptographic algorithms, multi-factor authentication, behavioural analytics, machine learning, and blockchain technologies. Each of these areas has its strengths, but limitations in terms of energy consumption, scalability issues, or insufficient adaptability prevent them from being used in isolation. Table 1 summarises the key protection methods, their advantages and disadvantages in the context of mobile platforms.

Table 1. Modern approaches to the protection of unstructured information on mobile platforms

Protection area	Technology examples	Benefits	Limitations
Cryptography	AES-256, RSA, ECC, LWE (quantum-resistant algorithms), hardware encryption (ARM TrustZone, Secure Enclave)	High level of security, data confidentiality, resistance to classic attacks	Vulnerability to quantum computing (RSA, ECC, AES), high energy consumption in LWE, and the need for optimisation for mobile devices.
Multi-factor authentication (MFA)	Password + biometrics (fingerprints, facial recognition) + context (geolocation, time)	Significantly reduces the risk of account compromise, increases trust	Problems with convenience, risk of false rejections, and additional burden on the user
Blockchain technologies	Hyperledger Fabric, Ethereum (hybrid architectures)	Data integrity, transaction transparency, secure identity management	High energy consumption, scalability issues, and integration complexity
Behavioural analytics	Behavioural biometrics, user pattern analysis	Continuous authentication, real-time anomaly detection	False positives, need for large data sets
Machine learning	RNN, logistic regression, ensemble methods, Federated Learning	High prediction accuracy, rapid attack detection, and adaptability	High computing costs, energy consumption, and data privacy issues

Source: compiled by the authors based on E. Ellavarason *et al.* (2020), A. Farissi *et al.* (2023), S. Ismail *et al.* (2024), F. Jumani & M. Raza (2025)

As Table 1 shows, no single method can provide universal and comprehensive protection for mobile systems. Cryptography is effective against classical attacks but vulnerable to quantum computing; multi-factor authentication significantly reduces the risk of compromise but affects user convenience; blockchain guarantees data immutability but is limited in scalability and energy efficiency; behavioural analytics and ML improve threat detection accuracy but require significant computing resources and consideration of privacy issues. This confirms the need to integrate these approaches into a single adaptive protection system that combines their advantages and compensates for their shortcomings through complementary architecture.

A substantial component of building a security system for mobile platforms is the creation of mathematical models that can predict the development of threats and forming optimal countermeasures in real time. This approach ensures the adaptability of the protection architecture and minimises the damage from attacks while maintaining device performance, as discussed by M.A. Ferrag *et al.* (2020). The state of the system is described by a multidimensional vector of parameters, including network activity, application usage, resource load, user biometric characteristics, and other factors. Based on this data, machine learning methods are applied, in particular recurrent neural networks (RNN), which analyse time series and can identify hidden patterns in user behaviour. Additionally, logistic regression and ensemble methods are used to predict the probability of attacks (Ciaburro & Iannace, 2021).

The results of the prediction are integrated into the decision-making mechanism, which is formulated as an optimisation task of selecting actions from a set of possible options: blocking access, activating VPN, requesting additional authentication, or increasing the level of encryption. Thus, the system can adapt its security settings depending on the threat context. For example, the encryption level changes in response to detected activity, and the frequency of key rotation depends on the assessed risk level. This approach combines accurate attack prediction with flexible response and provides a dynamic balance between security, performance, and user convenience. Threat modelling and adaptive decision-making facilitate quantitative comparisons of different protection methods based on key criteria, such as accuracy, speed, energy consumption, and resistance to attacks.

To obtain an objective assessment of the proposed solutions, a quantitative comparison of the main protection methods used in mobile platforms was conducted. In contrast to the generalised characteristics of the approaches shown in the previous table, the Table 2 shows the results of the analysis according to key criteria: threat detection accuracy, system speed, energy consumption

and resistance to attacks. This approach identifies the strengths and weaknesses of each technology not only at a theoretical level, but also at a practical level, which is relevant for mobile devices, where it is necessary to simultaneously ensure security, high performance, and economical use of resources.

Table 2. Comparison of the effectiveness of protection methods in mobile systems

Method/Technology	Primary function/effect	Speed (reaction time)	Energy consumption	Attack resilience
AES-256 (classical cryptography)	Data protection during storage and transmission, ensuring confidentiality	<1 cycle for most files	Low	Resistance to classical attacks, vulnerability to quantum attacks
LWE (quantum-resistant encryption)	Quantum-resistant encryption for long-term data storage	1-2 cycle (depending on file format)	High	Resistance to classical and quantum attacks
MFA (password + biometrics + context)	User authentication; reduction of the risk of compromise	2-3 cycle (authentication process)	Average	Resistance to phishing and social engineering
Blockchain (hybrid architecture)	Ensuring data integrity and immutability, transaction verification	Seconds-minutes (depending on the chain)	High	Resistance to modifications and data falsification
RNN (machine learning)	Real-time behaviour analysis and anomaly detection	<1.5 s	High	Highly effective against sophisticated and emerging attacks
Ensemble methods (Random Forest, Gradient Boosting)	Classification of threat patterns and anomaly detection	<1 s	Average	Resistance to known attack patterns
Behavioural biometrics	Continuous user authentication based on behaviour patterns	<1 s	Average	Resistance to device theft, but vulnerability to false refusals

Source: compiled by the authors based on M. Abuhamad *et al.* (2020), J.M. Ackerson *et al.* (2021), G.-Y. Kim *et al.* (2022), A. Zimba *et al.* (2025)

Analysis of the data presented in the table showed that each of the technologies considered has its own strengths and limitations that must be considered during practical implementation. Classic encryption algorithms, in particular AES, provide high performance and low power consumption, but remain vulnerable to quantum computing. Quantum-resistant approaches, such as LWE, demonstrate resistance to the latest types of attacks, but require additional optimisation due to high computational costs. Multi-factor authentication has proven effective in reducing the risk of account compromise but comes with usability issues and increased authentication time.

A comparative analysis of the theoretical characteristics of the methods demonstrated that AES-256 provides the best balance between speed and power consumption for mobile devices, while LWE is more resource-intensive but has higher resistance to quantum attacks. Among MFA authentication methods with behavioural parameters, it proved to be more energy efficient compared to biometric authentication, while maintaining a similar level of accuracy. Regarding machine learning methods, RNNs demonstrate higher accuracy in threat prediction, but ensemble models (Gradient Boosting, Random Forest) are characterised by lower energy consumption and more stable performance. The use of a hybrid blockchain architecture based on Hyperledger Fabric and Ethereum theoretically provides a balance between transaction speed and data storage transparency. Thus, the technologies are complementary: cryptographic mechanisms ensure confidentiality, blockchain ensures reliability, ML models ensure adaptability, and MFA ensures user authenticity. Their coordinated functioning can achieve a theoretical balance between accuracy, speed, energy efficiency, and resistance to attacks, which determines the promise of a comprehensive approach to protecting mobile platforms.

The results of a comparative analysis show that none of the individual protection methods provides a comprehensive level of security for mobile systems. Cryptographic algorithms guarantee high performance but are vulnerable to quantum attacks; machine learning methods ensure accurate threat detection but require significant resources; multi-factor authentication increases the level of protection but reduces user convenience; blockchain ensures data immutability but is limited in scalability.

Based on the analysis and theoretical comparison of various protection methods, a conceptual model of an integrated security system for unstructured information on mobile platforms was developed. This model was a logical result of the generalisation of data on cryptographic solutions, multi-factor authentication, behavioural analytics, machine learning methods, and blockchain architectures. The developed system is structured as a multi-level adaptive architecture in which each level performs separate functions of encryption, authentication, behavioural analysis, and data integrity assurance, but at the same time interacts with other modules to achieve comprehensive protection. A visual representation of the theoretical model is shown in Figure 1, which demonstrates the relationships between the cryptographic, authentication, analytical, and blockchain components of the system.

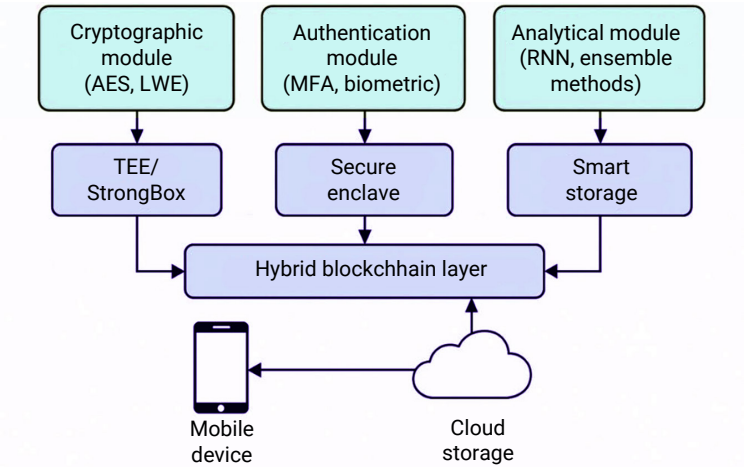

Figure 1. Systems for protecting unstructured information on mobile platforms
Source: compiled by the authors based on iOS Security: iOS 12.3 (2019), Google for Developers (2025)

The proposed architecture functions as an adaptive multi-level system in which security modules interact in real time. The cryptographic layer is responsible for data encryption and key rotation, the authentication layer is responsible for confirming user authenticity based on context and behavioural factors, and the analytical layer is responsible for predicting threats using machine learning models. The hybrid blockchain structure ensures the preservation of immutable records of access transactions, while the use of ARM TrustZone and Secure Enclave creates hardware isolation for critical operations. This system combines speed, transparency, and resistance to attacks in mobile environments while maintaining energy efficiency and scalability. It forms the basis for the practical implementation of adaptive security models in next-generation mobile platforms.

The use of blockchain technologies ensures a high level of transparency and immutability of data, but their practical implementation in mobile systems is accompanied by a range of limitations. According to theoretical studies, blockchain in the context of mobile applications improves authentication and data integrity, but its scalability and energy efficiency remain critical challenges for real-world use. Similar conclusions are presented by M.N. Alenezi *et al.* (2024), noting that most public chain-based solutions have increased energy consumption and require optimisation for integration into systems with limited resources.

Machine learning methods, in particular recurrent neural networks, are widely used to predict threats in mobile environments. ML models demonstrate a high ability to detect phishing attacks and malicious behaviour, but the effectiveness of such approaches largely depends on the quality and volume of training samples (Arslan *et al.*, 2016). This indicates that for practical use in mobile systems, models need to be adapted to changing conditions and limited device resources.

Behavioural biometrics and ensemble algorithms demonstrate balanced accuracy and performance, especially in cases where continuous user authentication must be combined with economical use of resources. Such methods minimise the risk of compromise even without the use of complex computational models, rendering them promising for integration into mobile platform security systems as emphasised by S. Kokal *et al.* (2023).

A generalised analysis of existing approaches shows that none of the protection methods considered provides a comprehensive level of security in mobile systems when used separately. In particular, cryptographic algorithms guarantee reliable data encryption, but are limited by energy efficiency and vulnerable to promising quantum attacks; multi-factor authentication methods significantly reduce the risk of account compromise, but are accompanied by usability issues and time delays; machine learning models provide high accuracy in detecting anomalies, but require significant computing resources; blockchain technologies guarantee transparency and immutability of records, but are characterised by increased energy consumption and scaling limitations. The combination of these factors justifies the development of an integrated approach, within which the strengths of individual technologies compensate for their individual limitations. This approach provides an optimal balance between accuracy, speed, energy efficiency, and resistance to attacks, which is critical for mobile platforms.

Thus, hybrid blockchain is a key element in achieving a balance between reliability, speed, and trust. Its integration into a comprehensive security architecture compensates for the weaknesses of other technologies and creates conditions for scalable and energy-efficient protection of unstructured information on modern mobile platforms. The results confirm the feasibility of using a multi-level, adaptive security system that can not only counter threats but also ensure resilience in the context of quantum computing and new types of attacks.

Discussion

An analysis of methods for protecting unstructured information on modern mobile platforms has shown that none of the existing technologies provides a sufficient level of

security when used in isolation from others. Instead, an integrated and adaptive approach ensures a balance is achieved between performance, resistance to attacks, ease of use, and the resource limitations of mobile devices. The results of the current work confirmed the effectiveness of AES-256 for data protection in mobile systems due to its high performance and low power consumption. This is fully consistent with the conclusions of S. Khan *et al.* (2024), demonstrated that AES provides an optimal balance between speed and resource consumption on modern smartphones. However, it did not incorporate the emerging threats associated with the development of quantum computing. This circumstance determined the key difference: the results showed that using AES alone is potentially dangerous in the long term.

R. Asif (2021) drew attention to the limited applicability of the quantum-resistant LWE algorithm for mobile systems due to its high energy consumption. The study confirmed the observation but demonstrated that the problem can be solved with an adaptive approach: LWE is applied only to the most critical transactions, while everyday data exchange is handled by AES. Thus, in the presented work, LWE is not rejected but integrated into a comprehensive architecture. Therefore, compared to the author's research, the results not only correlate with conclusions but also offer a way to overcome the limitations identified by them. The study proved that a hybrid combination of AES and LWE is the optimal option for mobile systems in the context of future quantum threats.

User authentication is one of the most vulnerable areas in mobile platforms. Presented research has shown that the use of multi-factor authentication with the additional use of contextual parameters can reduce the risk of account compromise. The results of A. Buriro *et al.* (2021) demonstrated that combining a password and biometrics reduces the risk by approximately 4-5 times. This is consistent with the current conclusion regarding the importance of MFA, but it has been proven that contextual factors (geolocation, access time, device type) significantly enhance the effectiveness of protection.

S.P. Karuppiah (2025), who studied MFA in financial applications, identified serious usability issues that negatively impacted the user experience. The theoretical model suggests that the implementation of behavioural continuous authentication can mitigate this limitation by providing an additional level of user verification. Additional factors are activated only when suspicious conditions are present. Thus, the presented approach ensures a balance between security and convenience, whereas the author's study primarily addressed improving security without considering usability.

The use of machine learning models in threat detection has proven to be effective. H. Seto *et al.* (2022) applied logistic regression and gradient boosting, achieving approximately 90% accuracy, but their models quickly lost effectiveness on new streaming data. The presented study demonstrated that RNNs can maintain stability in the dynamic environment of mobile systems, where data is constantly changing.

N.M. Rezk *et al.* (2020) confirmed the high efficiency of RNNs (~93%) but highlighted their excessive energy consumption. The current approach solved this problem through hybrid inference: under normal conditions, lightweight ensemble models operate, while RNNs are activated only when the risk increases. Thus, in the presented case, not only were the conclusions regarding accuracy confirmed, but they were also expanded upon through the optimisation of energy consumption. Furthermore, the proposed study demonstrated that RNNs are best suited for the analysis of temporal dependencies in mobile data.

The use of blockchain technology in the proposed study can be used for the creation of a hybrid architecture that combines the advantages of private and public chains. This has ensured a balance between transparency, speed, and trust in the system. X. Chen *et al.* (2022) showed that private blockchain provides high performance but has low transparency and less trust from external users. The proposed results confirmed this drawback but also proved that integration with a public blockchain maintains transparency without significant performance loss.

S. Sarkar *et al*. (2022) noted in the study based on the Zero Trust concept that strict verification mechanisms provide a high level of security but are accompanied by increased delays. The proposed approach addresses this problem by selectively activating complex checks based on threat prediction. This ensures the average response time is below 1.5 seconds, which previous studies have not achieved. Thus, the proposed model proves that it is possible to combine transparency, speed and efficiency, whereas the author's work emphasised only one of these parameters.

Proposed results demonstrated that combining VPN with behavioural analytics and ML can significantly improve security effectiveness on public Wi-Fi networks. J. Anyam *et al.* (2025) confirmed the effectiveness of VPNs (WireGuard, OpenVPN) for protecting mobile clients in their study but did not cover behavioural factors. The proposed approach has proven that it is the combination of VPN with ML that provides a faster response to threats, which is important in dynamic environments.

J. Abbott & S. Patil (2020) emphasised strict static access policies, which did reduce risks but significantly reduced usability. The proposed study showed that adaptive policies, which change depending on the level of risk, are more effective. This ensures a balance between security and usability, which aforementioned studies did not address.

An analysis of scientific sources demonstrated that the results of most studies are consistent with certain provisions of this work: AES is characterised by high performance, LWE is defined as a promising quantum-resistant approach, MFA significantly reduces the risks of compromise, RNN increases the accuracy of anomaly detection, and blockchain ensures data transparency and integrity. However, the main difference between the proposed and aforementioned study is the comprehensiveness and

adaptability. While the aforementioned studies considered technologies in isolation, the proposed model showed that their integrated use ensures optimal results. Thus, the proposed results not only confirmed the individual conclusions of previous studies, but also formed a new approach to protecting mobile systems – one that is comprehensive, adaptive, and resistant to future threats.

Conclusions

The study was theoretical in nature and is based on the analysis, comparison and generalisation of scientific sources devoted to the security of unstructured information on mobile platforms. Following the analysis of approaches, the study determined that individual methods – cryptographic algorithms, multi-factor authentication, behavioural analytics, blockchain technologies and machine learning methods – demonstrate high efficiency only in narrow areas of application, but do not provide systematic protection in the context of complex and dynamic cyber threats.

The analysis of the literature revealed the main trends in the development of security technologies: the transition to LWE, the spread of contextual multi-factor authentication, the use of RNN for behavioural monitoring, and the introduction of hybrid blockchain architectures to ensure data integrity. These approaches were generalised into a single conceptual model of an integrated system for protecting unstructured information, which is reflected in the diagram. The developed theoretical system involves the interaction of four main components: a cryptographic module (AES, LWE), an authentication module (MFA, biometrics, contextual factors), an analytical module (RNN, ensemble methods) and a hybrid blockchain level (Hyperledger Fabric + Ethereum), which operate in TEE, Secure Enclave and StrongBox environments. This architecture provides multi-level, complementary protection, which theoretically minimises the risks of data compromise, increases processing transparency and ensures resistance to quantum attacks.

Thus, theoretical generalisation has shown that the integrated approach, which combines the advantages of different technologies, has the highest potential. In particular, the hybrid combination of AES and LWE provides a balance between speed and quantum resistance; multi-factor authentication increases the reliability of user identification; behavioural analytics and ML models ensured adaptive response of the system to detected threats; blockchain ensures transparency and immutability of transactions.

In summary, the study confirmed the feasibility of developing a comprehensive system for protecting unstructured information on mobile platforms based on multi-level technology integration. The theoretically sound model can be used as a basis for further applied research aimed at its technical implementation, energy consumption optimisation, scalability improvement, and application in real industrial and consumer conditions. The limitations of the study are its theoretical nature and dependence on generalised data from previous studies, without empirical verification of the system's effectiveness in real conditions. Further research should be aimed at the practical implementation of the developed model, verification of stability in dynamic cyber scenarios, and optimising energy consumption on mobile devices.

Acknowledgements

None.

Funding

The study was not funded.

Conflict of Interest

None.

References

[1] Abbott, J., & Patil, S. (2020). How mandatory second factor affects the authentication user experience. In *Proceedings of the 2020 CHI conference on human factors in computing systems* (pp. 1-13). New York: ACM. doi: 10.1145/3313831.3376457.

[2] Abuhamad, M., Abusnaina, A., Nyang, D., & Mohaisen, D. (2020). Sensor-based continuous authentication of smartphones' users using behavioral biometrics: A contemporary survey. *ArXiv*. doi: 10.48550/arXiv.2001.08578.

[3] Aburbeian, A.M., & Fernández-Veiga, M. (2024). Secure internet financial transactions: A framework integrating multi-factor authentication and machine learning. *AI*, 5(1), 177-194. doi: 10.3390/ai5010010.

[4] Acien, A., Morales, A., Vera-Rodriguez, R., & Fierrez, J. (2020). Mobile active authentication based on multiple biometric and behavioral patterns. In T. Bourlai, P. Karampelas & V.M. Patel (Eds.), *Securing social identity in mobile platforms: Technologies for security, privacy and identity management* (pp. 161-177). Cham: Springer. doi: 10.1007/978-3-030-39489-9_9.

[5] Ackerson, J.M., Dave, R., & Seliya, N. (2021). Applications of recurrent neural network for biometric authentication & anomaly detection. *Information*, 12(7), article number 272. doi: 10.3390/info12070272.

[6] Alenezi, M.N., Alabdulrazzaq, H., Alhatlani, H.M., & Alobaid, F.A. (2024). Performance of AES algorithm variants. *International Journal of Information and Computer Security*, 23(3), 322-337. doi: 10.1504/IJICS.2024.138494.

[7] Anyam, J., Singh, R.R., Larijani, H., & Philip, A. (2025). Empirical performance analysis of WireGuard vs. OpenVPN in cloud and virtualised environments under simulated network conditions. *Computers*, 14(8), article number 326. doi: 10.3390/computers14080326.

[8] Arslan, B., Gunduz, S., & Sagiroglu, S. (2016). A review on mobile threats and ML-based detection approaches. In *2016 4th international symposium on digital forensic and security* (pp. 7-13). Little Rock: IEEE. doi: 10.1109/ISDFS.2016.7473509.

[9] Asif, R. (2021). Post-quantum cryptosystems for internet-of-things: A survey on lattice-based algorithms. *IoT*, 2(1), 71-91. doi: 10.3390/iot2010005.
[10] Banoth, R., & Regar, R. (2023). An introduction to classical and modern cryptography. In *Classical and modern cryptography for beginners* (pp. 1-46). Cham: Springer. doi: 10.1007/978-3-031-32959-3_1.
[11] Bonnie, E. (2025). *110+ of the latest data breach statistics to know for 2026 & beyond.* Retrieved from https://secureframe.com/blog/data-breach-statistics.
[12] Brovchenko, E.M., Samarai, V.P., Datsenko, I.P., Pavlenko, V.I., & Sereda, A.V. (2023). Protection of unstructured information on a mobile device. *Infocommunication and Computer Technologies*, 1(5), 194-200. doi: 10.36994/2788-5518-2023-01-05-21.
[13] Buriro, A., Gupta, S., Yautsiukhin, A., & Crispo, B. (2021). Risk-driven behavioral biometric-based one-shot-cum-continuous user authentication scheme. *Journal of Signal Processing Systems,* 93(9), 989-1006. doi: 10.1007/s11265-021-01654-2.
[14] Chen, X., Miraz, M.H., Gazi, A.I., Rahaman, A., Habib, M., & Hossain, A.I. (2022). Factors affecting cryptocurrency adoption in digital business transactions: The mediating role of customer satisfaction. *Technology in Society,* 70, article number 102059. doi: 10.1016/j.techsoc.2022.102059.
[15] Ciaburro, G., & Iannace, G. (2021). Machine learning-based algorithms to knowledge extraction from time series data: A review. *Data*, 6(6), article umber 55. doi: 10.3390/data6060055.
[16] Ellavarason, E., Guest, R., Deravi, F., Sanchez-Riello, R., & Corsetti, B. (2020). Touch-dynamics based behavioural biometrics on mobile devices – a review from a usability and performance perspective. *ACM Computing Surveys*, 53(6), article number 120. doi: 10.1145/3394713.
[17] Farissi, A., Pradata, A., & Miraswan, K. (2023). Securing messages using AES algorithm and blockchain technology on mobile devices. *Synchronous*, 7(2), 1166-1171. doi: 10.33395/sinkron.v8i2.12381.
[18] Ferrag, M.A., Maglaras, L., Derhab, A., & Janicke, H. (2020). Authentication schemes for smart mobile devices: Threat models, countermeasures, and open research issues. *Telecommunication Systems*, 73(2), 317-348. doi: 10.1007/s11235-019-00612-5.
[19] Google for Developers. (2025). *Android Keystore system.* Retrieved from https://developer.android.com/privacy-and-security/keystore.
[20] He, Y., Huang, D., Chen, L., Ni, Y., & Ma, X. (2022). A survey on Zero Trust architecture: Challenges and future trends. *Wireless Communications and Mobile Computing*, 2022(1), article number 6476274. doi: 10.1155/2022/6476274.
[21] iOS Security: iOS 12.3. (2019). Retrieved from https://css.csail.mit.edu/6.858/2023/readings/ios-security-may19.pdf.
[22] Ismail, S., Nouman, M., Dawoud, D.W., & Reza, H. (2024). Towards a lightweight security framework using blockchain and machine learning. *Blockchain: Research and Applications,* 5(1), article number 100174. doi: 10.1016/j.bcra.2023.100174.
[23] Jumani, F., & Raza, M. (2025). Machine learning for anomaly detection in blockchain: A critical analysis, empirical validation, and future outlook. *Computers*, 14(7), article number 247. doi: 10.3390/computers14070247.
[24] Kandula, S.R. (2025). Breaking traditional encryption: Quantum computing risks to web and mobile applications. *International Journal of Advanced Research in Engineering and Technology,* 16(2), 329-342. doi: 10.34218/IJARET_16_02_020.
[25] Karuppiah, S.P. (2025). *Understanding the behaviour of business users in multi-factor authentication adoption*. (Master's thesis, Lappeenranta-Lahti University of Technology LUT, Lappeenranta, Finland).
[26] Khan, S., Krishnamoorthy, P., Goswami, M., Rakhimjionovna, F.M., Mohammed, S.A., & Menaga, D. (2024). Quantum computing and its implications for cybersecurity: A comprehensive review of emerging threats and defenses. *Nanotechnology Perceptions,* 20(13), 1232-1248. doi: 10.62441/nano-ntp.v20iS13.79.
[27] Kim, G.-Y., Lim, S.-M., & Euom, I.-C. (2022). A study on performance metrics for anomaly detection based on industrial control system operation data. *Electronics*, 11(8), article number 1213. doi: 10.3390/electronics11081213.
[28] Kokal, S., Vanamala, M., & Dave, R. (2023). Deep learning and machine learning, better together than apart: A review on biometrics mobile authentication. *Journal of Cybersecurity and Privacy,* 3(2), 227-258. doi: 10.3390/jcp3020013.
[29] Kumar, N. (2025). *Latest smartphone usage statistics 2026 (Worldwide).* Retrieved from https://surl.lu/eucitc.
[30] Lim, W.Y.B., Luong, N.C., Hoang, D.T., Jiao, Y., Liang, Y.-C., Yang, Q., Niyato, D., & Miao, C. (2020). Federated learning in mobile edge networks: A comprehensive survey. *ArXiv*. doi: 10.48550/arXiv.1909.11875.
[31] Liu, Y., He, D., Obaidat, M.S., Kumar, N., Khan, M.K., & Choo, K.-K.R. (2020). Blockchain-based identity management systems: A review. *Journal of Network and Computer Applications,* 166, article number 102731. doi: 10.1016/j.jnca.2020.102731.
[32] Mahor, V., Pachlasiya, K., Garg, B., Chouhan, M., Telang, S., & Rawat, R. (2021). Mobile operating system (Android) vulnerability analysis using machine learning. In D. Giri, J.K. Mandal, K. Sakurai & D. De (Eds.), *Proceedings of international conference on network security and blockchain technology: ICNSBT 2021* (pp. 159-169). Singapore: Springer. doi: 10.1007/978-981-19-3182-6_13.

[33] Malik, G. (2024). Biometric authentication: Risks and advancements in biometric security systems. *Journal of Computer Science and Technology Studies*, 6(3), 159-180. doi: 10.32996/jcsts.2024.6.3.14.

[34] Martín, G.A., Fernández-Isabel, A., de Diego, I.M., & Beltrán, M. (2021). A survey for user behavior analysis based on machine learning techniques: Current models and applications. *Applied Intelligence,* 51(8), 6029-6055. doi: 10.1007/s10489-020-02160-x.

[35] Mehwish, Zaheer, M., Azeem, M.H., Afzal, Z., & Karim, H. (2024). Critical evaluation of data privacy and security threats in federated learning: Issues and challenges related to privacy and security in IoT. *Spectrum of Engineering Sciences*, 2(5), 458-479.

[36] Prokopovych-Tkachenko, D., Bakuta, A., Zverev, V., Kozachenko, I., & Cherkasky, O. (2025). Modeling phishing scenarios in Ukraine cyberspace: An analytical approach using Grafana-board. *Electronic Professional Scientific Journal "Cybersecurity: Education, Science, Technique",* 1(29), 331-347. doi: 10.28925/2663-4023.2025.29.881.

[37] Rezk, N.M., Purnaprajna, M., Nordström, T., & Ul-Abdin, Z. (2020). Recurrent neural networks: An embedded computing perspective. *IEEE Access,* 8, 57967-57996. doi: 10.1109/ACCESS.2020.2982416.

[38] Sarkar, S., Choudhary, G., Shandilya, S.K., Hussain, A., & Kim, H. (2022). Security of Zero Trust networks in cloud computing: A comparative review. *Sustainability*, 14(18), article number 11213. doi: 10.3390/su141811213.

[39] Seto, H., *et al*. (2022). Gradient boosting decision tree becomes more reliable than logistic regression in predicting probability for diabetes with big data. *Scientific Reports,* 12, article number 15889. doi: 10.1038/s41598-022-20149-z.

[40] Shamsuddin, N.S.M., & Pitchay, S.A. (2020). Implementing location-based cryptography on mobile application design to secure data in cloud storage. *Journal of Physics: Conference Series*, 1551, article number 012008. doi: 10.1088/1742-6596/1551/1/012008.

[41] Shifa, A., Asghar, M.N., Fleury, M., Kanwal, N., Ansari, M.S., Lee, B., Herbst, M., & Qiao, Y. (2020). MULVIS: Multi-level encryption based security system for surveillance videos. *IEEE Access,* 8, 177131-177155. doi: 10.1109/ACCESS.2020.3024926.

[42] Smith, G. (2025). *+95 cyber security breach statistics 2025.* Retrieved from https://www.stationx.net/cyber-security-breach-statistics.

[43] Wei, X. (2022). Smart mobile information systems and blockchain privacy protection. *Mathematical Problems in Engineering*, 2022(1), article number 5126326. doi: 10.1155/2022/5126326.

[44] Woźniak, M., Siłka, J., Wieczorek, M., & Alrashoud, M. (2021). Recurrent neural network model for IoT malware detection. *IEEE Transactions on Industrial Informatics*, 17(8), 5583-5594. doi: 10.1109/TII.2020.3021689.

[45] Yadav, A.K. (2021). Significance of elliptic curve cryptography in blockchain IoT with comparative analysis of RSA algorithm. In *2021 international conference on computing, communication, and intelligent systems* (pp. 256-262). Greater Noida: IEEE. doi: 10.1109/ICCCIS51004.2021.9397166.

[46] Zimba, A., Phiri, K.O., Mulenga, M., & Mukupa, G. (2025). A systematic literature review of blockchain technology and energy efficiency based on consensus mechanisms, architectural innovations, and sustainable solutions. *Discover Analytics,* 3(1), article number 14. doi: 10.1007/s44257-025-00041-6.

Метод захисту неструктурованої інформації на сучасних мобільних платформах: моделювання загроз та аналіз ефективності

Євген Бровченко

Аспірант
Відкритий міжнародний університет розвитку людини «Україна»
04071, вул. Львівська, 23, м. Київ, Україна
https://orcid.org/0000-0002-1416-0385

Валерій Самарай

Кандидат технічних наук, доцент
Центр воєнно-стратегічних досліджень Національного університету оборони України
03049, просп. Повітряних Сил, 28, м. Київ, Україна
https://orcid.org/0000-0003-4419-1366

Анотація. Метою дослідження було розроблення комплексного підходу до захисту неструктурованої інформації на мобільних платформах шляхом поєднання криптографічних алгоритмів, багатофакторної автентифікації, методів машинного навчання та блокчейн-технологій для створення адаптивної системи безпеки. Методологія дослідження базувалася на теоретичному аналізі наукових джерел і моделюванні архітектури системи захисту неструктурованої інформації, орієнтованої на сучасні мобільні платформи. У роботі розглядалося використання пристроїв із підтримкою Advanced RISC Machine TrustZone та Secure Enclave, що забезпечують апаратну ізоляцію криптографічних операцій. Як базові алгоритми шифрування застосовувалися Advanced Encryption Standard для симетричного захисту даних і Learning With Errors як квантово-стійкий механізм. У межах дослідження була сформована концептуальна багаторівнева модель інтегрованої системи безпеки, що включає чотири взаємодіючі шари: криптографічний, автентифікаційний, аналітичний (поведінкова аналітика та методи машинного навчання) та блокчейн-рівень. Кожен із шарів виконує окрему функцію: шифрування й апаратну ізоляцію операцій, підтвердження достовірності користувача, виявлення аномалій та забезпечення цілісності даних, – і в сукупності вони формують адаптивну систему захисту мобільних платформ. Особливу увагу приділено впровадженню гібридного блокчейну, який поєднує високу швидкодію приватних ланцюгів із незалежною перевіркою транзакцій у публічних блоках. Такий підхід забезпечив баланс між прозорістю, енергоефективністю та стійкістю до модифікацій. Теоретичний аналіз підтвердив, що інтеграція цих компонентів у єдину архітектуру створює умови для формування адаптивної системи безпеки, здатної динамічно реагувати на загрози й забезпечувати високий рівень захисту неструктурованих даних у мобільних середовищах. Запропонований підхід може бути впроваджений у сферах медицини, фінансів, державного управління та інших галузях, де захист неструктурованої інформації є критично важливим

Ключові слова: багатофакторна автентифікація; рекурентні нейронні мережі; логістична регресія; адаптивне шифрування; гібридна блокчейн-архітектура

Journal homepage: https://itce.vn.ua/en

Information Technologies and Computer Engineering **Vol. 23, No. 1. 2026**

Article's History: Received: 27.10.2025 Revised: 23.01.2026
Accepted: 26.03.2026 Published: 20.04.2026

ISSN 1999-9941; e-ISSN 2078-6387
UDC 004.85
DOI: 10.31649/vitce/1.2026.72

Forecasting of time series using a neural network with parallel-stacked LSTM blocks

Yurii Futryk*
Postgraduate Student
Lviv Polytechnic National University
79000, 12 Stepan Bandera Str., Lviv, Ukraine
https://orcid.org/0000-0001-5271-9883

Ivan Peleshchak
PhD, Associate Professor
Lviv Polytechnic National University
79000, 12 Stepan Bandera Str., Lviv, Ukraine
https://orcid.org/0000-0002-7481-8628

Abstract. Time series forecasting is crucial for supporting decisions in financial analytics, where data is characterised by non-linearity, non-stationarity, and high noise levels. The purpose of the study was to determine the effective configuration of a recurrent neural network with a parallel combination of Long Short-Term Memory (LSTM) cell stacks to improve the accuracy of stock price forecasting, and the possibilities of applying the back-end model in industry, energy, and related domains. The study applied deep learning methods using the TensorFlow/Keras library, and used historical data from Google shares to train the model. It was established that the architecture with parallel-stacked blocks provided higher learning stability compared to standard recurrent models due to more efficient allocation of technical features of the time sequence. It has been experimentally proven that the optimal number of neurons in the hidden layers for such a task was 100-200 units, while a further increase in the power of the model lead to a retraining effect. It was found that the use of dropout regularisation in the range of 0.1-0.2 minimised the error in the validation sample, while values over 0.3 significantly slowed down the convergence of the algorithm. Feature analysis showed that integrating an exponential moving average with a short time window improved the model result, showing a higher correlation with the target index than the relative strength index. The prediction quality of the model was evaluated by the Mean Squared error (MSE), the Root Mean Squared Error (RMSE), and the Mean Absolute Percentage Error (MAPE). It was found that configurations (50-100 blocks) were characterised by increased MAPE values, while in the range of 180-400 blocks the error decreased and became stable. The most accurate result was obtained for a configuration with 325 blocks, Dropout regularisation = 0.05 and Nadam optimiser (Nesterov-accelerated Adam): MAPE = 1.62%, RMSE = 2.41, MSE = 6.05. The practical significance of the study lied in the formulation of clear recommendations for setting up hyperparameters of LSTM models for applied short-term forecasting of financial series

Keywords: deep learning; Dropout-regularisation; Nadam-optimisation; EMA; RSI

Introduction

The task of predicting time series is one of the most important challenges of applied analytics in the 2010-2026 – it appears in various domains from exchange analysis and network load to high-frequency sensor data, where long and short inter-time dependencies are manifested. Reproducibility of evaluation procedures (time split, stable normalisation, non-mixing of samples), and transparency of model settings remain a priority for researchers and practitioners.

In the course of forecasting financial time series, there is a need for methods that combine practical efficiency with ensuring transparency and reproducibility of the results obtained. Long Short-Term Memory (LSTM) neural networks are widely used in many industries to predict time series, such as detecting equipment failures, predicting production line loads, and improving logistics management efficiency. Due to its ability to model long-term dependencies, LSTMs

Suggested Citation:
Futryk, Yu., & Peleshchak, I. (2026). Forecasting of time series using a neural network with parallel-stacked LSTM blocks. *Information Technologies and Computer Engineering*, 23(1), 72-82. doi: 10.31649/vitce/1.2026.72

*Corresponding author

allow achieving high accuracy in complex forecasts, especially in finance and other critical areas.

In the global scientific literature on financial forecasting, the problem of practical selection of the configuration of LSTM models (number of blocks, strength of regularisation, and composition of features) remains, since accuracy indicators are sensitive to training settings and assessment conditions; therefore, the development of reproducible applied recommendations for short-term forecasting is relevant. In contemporary studies of short-term forecasting of financial time series, key attention was paid to the configuration of recurrent LSTM models, the level of Dropout regularisation, and the influence of technical indicators EMA (Exponential Moving Average) and RSI (Relative Strength Index) on the accuracy of forecasts. For example, O.B. Sezer *et al*. (2020) conducted a review of deep learning in financial forecasting and showed that incorrect time separation and different preprocessing often distort model comparisons. The researchers emphasised the need for a reproducible protocol (chronological division into training and test data, uniform scaling) and transparent reporting of MAPE (Mean Absolute Percentage Error) and RMSE (Root Mean Squared Error) metrics.

I.E. Livieris *et al*. (2020) investigated hybrid architectures that combine CNN (convolutional neural network) and LSTM (recurrent neural network). In such architectures, CNNs are used to isolate local patterns in data, and LSTMs are used to model time dependencies. Researchers have shown that a more complex architecture can improve the quality of prediction, but simultaneously make the model more sensitive to hyperparameter settings (for example, the number of layers or regularisation level). This means that for correct comparison of such models, it is necessary to provide controlled conditions with the same settings for all models. In turn, P.T. Yamak *et al*. (2020) compared statistical and neural network methods for predicting financial time series and noted that the advantages of recurrent models, such as LSTM, are manifested if the observation window and model settings are correctly selected. They stressed that the correctness of conclusions about the effectiveness of models depends on the same conditions for preparing data and evaluation metrics. This confirmed the need to use the same protocols for correct comparison of models.

In a similar context, S. Smyl (2020) demonstrated the effectiveness of combining statistical methods, such as exponential smoothing, with neural networks for predicting time series. The key conclusion of their study was that the values of forecast errors make sense only if the forecast horizon and evaluation protocol were clearly defined. If these conditions were not considered, the advantage of one model over another may be the result of different evaluation conditions or data, rather than architecture. B. Lindemann *et al*. (2021) considered another important issue – the reproducibility of results when using LSTM to predict time series. They found that the stability of the results largely depends on factors such as model capacity (hidden state parameters and number of cells), regularisation, and validation procedure. The researchers noted that recommendations for choosing hyperparameters (for example, the number of blocks or regularisation parameters) can only be valid if there is a complete description of the training and testing protocol. This highlights the importance of clear and transparent reporting of model settings, which allows achieving reproducibility of results and comparing models under the same conditions.

In this context, B. Lim *et al*. (2021) drew attention to the importance of controlled experiments when comparing optimisation algorithms. They stressed that the same data preparation, regularisation, forecast horizons, and evaluation metrics should be used to correctly compare optimisers. If these conditions are not met, conclusions about the superiority of one optimiser over another may be unstable, since the result will depend on changes in settings, and not on the quality of the optimiser itself. Additionally, M. Ez-zaiym *et al*. (2025) gave an example of a controlled comparison of Adam (Adaptive Moment Estimation) and Nadam (Nesterov-accelerated Adaptive Moment Estimation) optimisers under agreed training conditions. The researchers showed that generalising the advantage of one optimiser over another is limited without fixing architectural parameters (such as model capacity and Dropout level) and other settings. Therefore, the optimiser should be interpreted as part of a holistic model configuration, where all components must be configured in a single context to achieve reliable results.

According to H. Widiputra *et al*. (2021), changes in the composition of input features in the multivariate formulation of financial forecasting can significantly affect quality metrics, even if the model architecture remains unchanged. They stressed that the contribution of technical indicators should be evaluated only under fixed preprocessing and the same normalisation conditions. This approach avoids mixed effects that can distort the results of model comparisons. H. Abbasimehr & R. Paki (2022) proposed combining LSTM with attention mechanisms for predicting time series, showing that attention helps the model focus on the most relevant parts of history. However, the researchers noted that the gain in accuracy depends on the specific task and does not eliminate the need for systematic selection of hyperparameters and retraining control. Similarly, B. Ghojogh & A. Ghodsi (2023) summarised current approaches to sequence modelling, emphasising the role of "memory" and regularisation mechanisms in the learning stability of recurrent models. They stressed that the competitive quality of models depends not only on the choice of the recurrent block type, but also on the consistency of the evaluation protocol and the correct configuration of hyperparameters for a specific data set.

For the most part, the publications analysed focused on the use of LSTM for predicting financial time series, with an emphasis on practical accuracy. In particular, I. Peleshchak & Y. Futryk (2025) proposed a new neural network configuration with parallel-stacked LSTM blocks, which

significantly improved the accuracy of predictions. They demonstrated that combining LSTM with technical indicators (EMA, RSI) can significantly reduce forecasting errors. Their study highlighted the importance of a systematic approach to setting up models and selecting hyperparameters to achieve stability and accuracy of results.

Furthermore, the analysis of contemporary sources shows that for short-term financial forecasting, applied recommendations for joint adjustment of the number of LSTM blocks/model capacity, the level of Dropout regularisation, and the feasibility of including technical indicators EMA and RSI under the reproducible assessment protocol are not sufficiently systematised. That is why the purpose of the study was to predict and analyse time series with high accuracy on the MSE, RMSE, and MAPE metrics (⩽ 1.9%), and experimental verification of the configuration of the neural network model with parallel-stacked LSTM blocks, considering the exponential mean and relative strength index indicators on the time data set. To achieve this goal, the following tasks were set: to determine the rational configuration of the model by systematically varying the number of LSTM blocks and the level of Dropout regularisation; to assess the contribution of technical indicators EMA and RSI to the quality of forecasting using a fixed data preparation protocol; to conduct a comprehensive assessment of the quality of forecasting using agreed metrics.

Materials and Methods

Experimental database and data selection. A dataset from the well-known company Google, sourced from the open-access platform Yahoo Finance (n.d.), was used to conduct the computer experiment. As part of the study, a dataset was generated based on Google's historical stock prices. The generated dataset covered the period from January 2011 to August 2025 and contained 3,687 records (trading days from the yfinance source). The closing price was chosen as the target variable for forecasting, since this indicator reflected the final valuation of the asset for the trading session and was representative of the analysis of the dynamics of the financial time series.

The experimental data was downloaded from the Python library yfinance (Kurniawan *et al.*, 2024), which provided convenient access to financial indicators. The model was implemented in the Python programming language using the following libraries: NumPy/Pandas (preparation and numerical calculations), Keras (modelling), Matplotlib (visualisation), Yfinance (library for obtaining historical financial data from the Yahoo Finance API source). Technical indicators of the EMA and RSI were additionally calculated to form signs. Flowcharts are visualised using Draw.io (n.d.)

Neural network architecture with parallel-stackable LSTM blocks. The proposed model was a neural network with parallel-stacked LSTM blocks. Figure 1 shows the architecture of a recurrent neural network consisting of parallel LSTM blocks. Standard LSTM cells with input, output, and "forget"-gates were used. Each block in this network processes data received at a specific time point, and through the interaction mechanism between blocks accumulates information from previous time intervals. This principle allowed the model not only to generate output values, but also to adjust its internal state, which ensured higher accuracy.

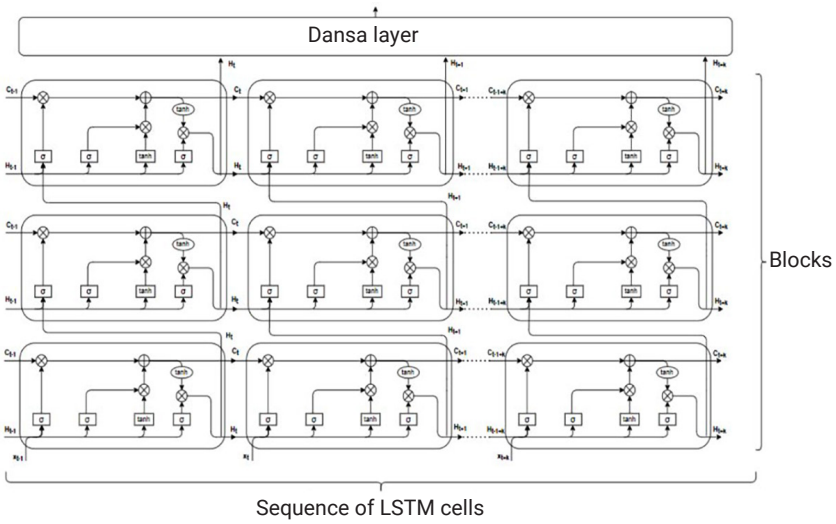

Figure 1. Morphology of a neural network with parallel-stacked LSTM blocks

Source: developed by the authors based on V. Lytvyn *et al*. (2025)

Intermediate views generated by consecutive LSTM layers were then fed to the output tightly coupled layer (Dense), which generates the final forecast value. Within the LSTM blocks themselves (Fig. 2), sigmoid and/or hyperbolic tangent (tanh) nonlinearities are used to construct the hidden state, controlling the activation of the memory cells, and the output signal. The final Dense-layer, which converts these features into scalar prediction, works with linear activation, or can use ReLU in cases where the model must be limited to non-negative or scalable predicted values.

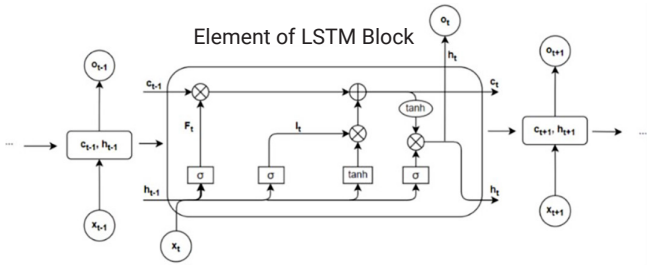

Figure 2. Architecture of a separate LSTM block element
Source: developed by the authors based on Q. Wang & Y. Zhang (2022)

LSTM manages memory sequentially at each step through three independent "solution nodes" (Widiputra *et al.*, 2021): (1) forget something from the previous state; (2) add new information; (3) output a useful segment. The above three-component scheme provided a controlled update of the internal state and reduced the risk of error accumulation when processing long sequences. Further, h_{t-1} indicates the previous hidden state, x_t – current input, and C_{t-1} – state of the memory (cell) in the previous step.

At the first stage, the "forget gate" node is considered – the network viewed the previous hidden state h_{t-1} together with the current input x_t and decide that from an old memory C_{t-1} to leave it. The solution is set by coefficients from 0 to 1 for each memory cell: 0 – erase, 1 – save. Thus, forget gate acts as a selective filter that controls the share of stored information in memory.

$$f_t = \sigma(W_t * [h_{t-1}, x_t] + b_f), \quad (1)$$

where f_t – vector of values in the range [0, 1], which determines what proportion of information in the cell C_{t-1} must be saved (1) or forgotten (0); W_t, b_f – weights and offsets that are updated during training; σ – sigmoid activation function.

The next node is "input" (input gate and candidates). Here, the LSTM determines what exactly to add to memory: (a) through the "tolerance node", the network selects which cells are allowed to be updated; (B) separately calculates candidate values bounded by the segment [-1,1]. As a result, the previous memory (after the "forget" node) is added to the selected part of candidates and gives updated memory, creating a vector of new values that are candidates for updating elements. In this way, input gate coordinates "what to update" (tolerance mask) and "what to update" (candidate values), providing managed input of new information.

$$i_t = \sigma(W_i * [h_{t-1}, x_t] + b_i), \quad (2)$$

$$C_t = tan\,h(W_C * [h_{t-1}, x_t] + b_C), \quad (3)$$

where i_t – vector of "activations" that determines the update of information; C_t – vector of candidate values for updating memory.

The final output gate node is responsible for generating a filtered version of the updated memory: first, the network decides which part of the internal state should be "publicised" at the current stage, and then converts it to a new hidden state h_t. It restricts the transfer of secondary memory components and skips only those features that are relevant to the forecast at a given time step. This reduces the risk of random fluctuations (noise) and maintains the stability of hidden state dynamics in long sequences. The resulting hidden state is passed further along the sequence and used for further prediction, in particular, it is fed to the next LSTM block or to the original dense layer in regression problems:

$$o_t = \sigma(W_0 * [h_{t-1}), x_t] + b_0), \quad (4)$$

$$h_t = o_t * tan\,h(C_t), \quad (5)$$

where o_t – vector of output signal values; h_t – updated hidden state passed to the next time step.

In this setting, a neural network with parallel-stacked LSTM blocks implements controlled memory: some scales learn to forget too much, others learn to dose new signals, and others learn to responsibly open the "exit valve". This ensured more stable preservation of important patterns in the time series without noise accumulation. As a result of this principle, a hidden state was formed h_t, which summarised the context of previous steps and was used as input for the next prediction step.

Metrics for evaluating time series prediction by a neural network with parallel-stacked LSTM blocks. Standard MSE, RMSE, and MAPE metrics were used to quantify the quality of the model's prediction, and to identify signs of overtraining, which are described in detail in the paper by D. Chicco *et al.* (2021). MSE was used as a baseline metric to optimise and compare forecasts with actual values in the test set:

$$MSE = \frac{1}{n}\sum_{i=0}^{n}(y_i - \hat{y}_i)^2, \quad (6)$$

where y_i – actual value of the target variable for the i-th observation; $\hat{y}_i$ – projected model value; n – number of observations in the sample; i – observation index, $i = 1...n$.

To interpret the error in the units of measurement of the studied variable, RMSE was used, which was calculated from forecasts and actual values in the test sample:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=0}^{n}(y_i - \hat{y}_i)^2}, \quad (7)$$

where y_i – actual value of the target variable for the i-th observation; $\hat{y}_i$ – projected model value; n – number of observations in the sample; i – observation index, $i = 1...n$.

Additionally, MAPE was used to represent the percentage error and compare the results between different time intervals/model settings:

$$MAPE = \frac{1}{n}\sum_{i=0}^{n}\left|\frac{y_i - \hat{y}_i}{y_i}\right| * 100\%, \quad (8)$$

where y_i – actual value of the target variable for the i-th observation, $y_i \neq 0$; $\hat{y}_i$ – projected model value; n – number of observations in the sample; $|\cdot|$ – module (absolute value); i – observation index, $i = 1...n$.

Data for the experiment were divided chronologically into train/validation/test samples (without mixing) to avoid information leakage from the future to the past. When comparing models, the target variable Close, input window length 60, Min-Max normalisation (parameters were calculated on "train" and applied to validation/test), and MSE, RMSE, and MAPE metrics were recorded. The number of parallel-stacked LSTM blocks, Dropout, and feature set (with/without EMA and RSI) varied. Results were obtained based on metrics in the test sample, and the difference between train and validation was used to control retraining. The limitation of the experiment was execution for one asset (GOOGL) and one source (Yahoo Finance/yfinance), so generalisation to other instruments and market regimes requires additional verification. Estimates may also vary depending on the input window selection, separation scheme, and hyperparameters.

Results and Discussion

This section presented the results of an experimental test of the performance and accuracy of a neural network with parallel-stacked LSTM blocks on the financial time series of Google shares using a fixed training protocol and the same preprocessing pipeline. To test the stability and impact of architectural solutions, two model configurations were considered, which differed in the number of parallel-stacked LSTM blocks and the Dropout value, while the remaining components of the experiment remained unchanged. Training in both cases was carried out under the same optimisation conditions using the adaptive optimiser Nadam. Figure 3 shows a flowchart of the parallel-stacked LSTM model used in the experiments. The scheme summarises the processing sequence: generating an input window and features, LSTM layers with Dropout, and an output Dense layer that generates a forecast for the target variable Close.

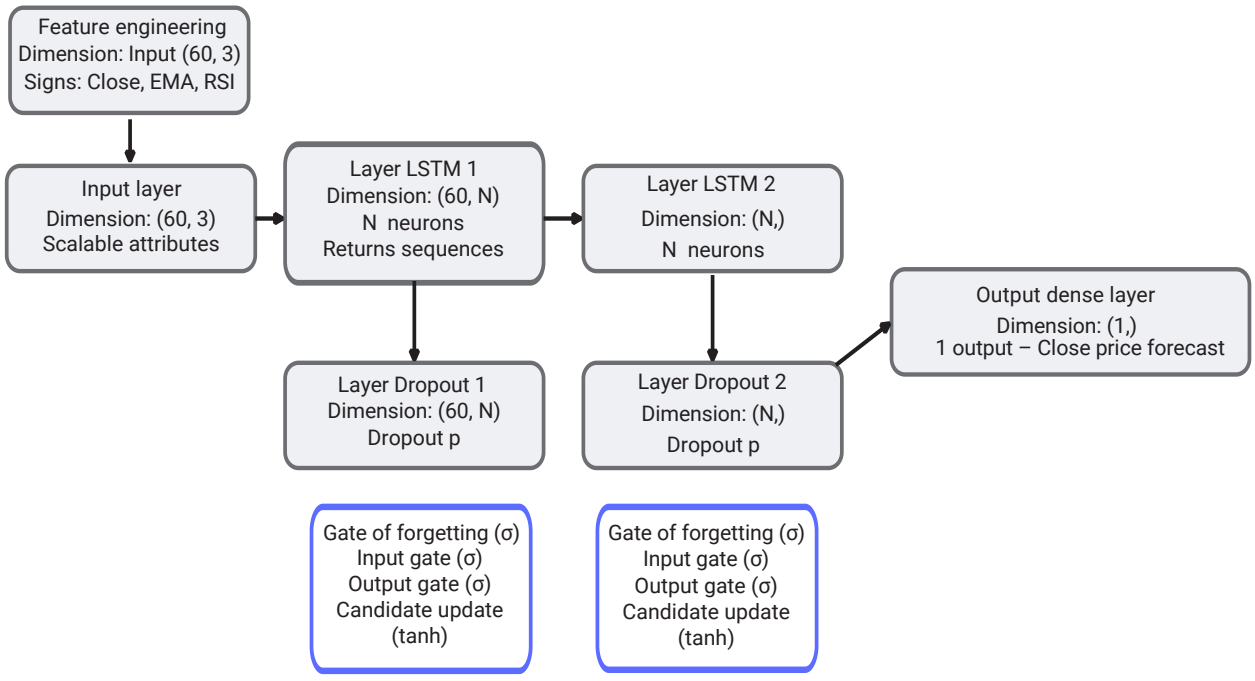

Figure 3. Flowchart of a parallel-stacked LSTM model

Note: N – number of LSTM units in the LSTM layer; p – regularisation coefficient

Source: developed by the authors using the Draw.io tool (n.d.)

As part of the hyperparameter selection, experimental combinations were compared that varied the number of parallel-stacked LSTM blocks and the dropout regularisation level, while the optimiser (Nadam) and feature set (Close, EMA_20, RSI) remained constant. For the final comparison, two configurations were selected: Set A (275; Dropout = 0.10) and Set B (325; Dropout = 0.05), since these settings provided an optimal ratio of accuracy and stability in validation/test using the MSE, RMSE, and MAPE metrics in the fixed training protocol. The total values of the selected hyperparameters and the composition of features for each configuration are shown in Table 1.

Table 1. Values of neural network hyperparameters with parallel-stacked LSTM blocks

Hyperparameters	Number of blocks	Dropout	Optimiser	Technical features
Set A	275	0.1	Nadam	Close, EMA_20, RSI
Set B	325	0.05	Nadam	Close, EMA_20, RSI

Source: developed by the authors based on the results of experiments

Figure 4 shows a comparison of the actual close price dynamics and model predictions for two configurations (A and B) using the same preprocessing and training protocol. A visual assessment was made of how well the forecast matched the actual values on the test segment, and the nature of the errors during periods of sharp trend changes (peaks/dips), where models typically produce the greatest deviations. Special attention was drawn to the gap between the actual series and the forecast at the end of the test interval, since it is most indicative of the model's ability to generalise historical fluctuations. A comparison of subgraphs (A) and (B) demonstrates how changes in the number of LSTM blocks and Dropout affect the stability of the forecast trajectory and noise sensitivity.

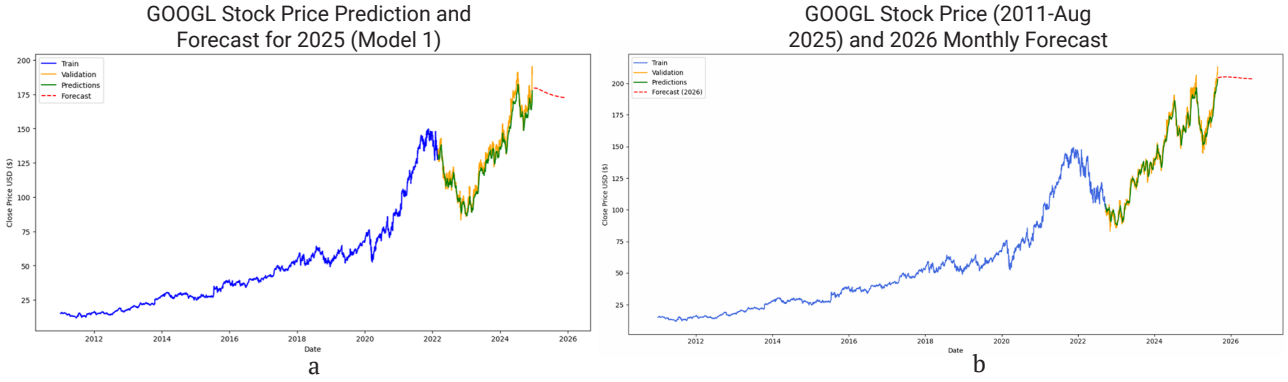

Figure 4. Graphs of stock price forecasts using a neural network with parallel-stacked LSTM blocks

Note: a – for hyperparameters of type A; b – for hyperparameters of type B. Target variable Close, period 2011-2025; A/B configurations according to Table 1. Training, validation, and predicted values are marked with a standard palette, where the green line is the predicted values during verification

Source: compiled by the authors

Figures 5 and 6 illustrate the MAPE dependence on the number of parallel-stacked LSTM blocks (under fixed experimental conditions), which reflects the relationship between the lack of complexity of the model and the risk of retraining. The minimum of the curve corresponds to the area of best alignment of forecasts with actual values in validation/test, so this graph is used to justify the choice of configurations A and B. Increasing the number of blocks does not guarantee improvement, and after a certain limit, the error may increase due to excessive complexity and noise sensitivity. Final configuration comparisons were performed using metrics in the test sample, while validation was used to control retraining.

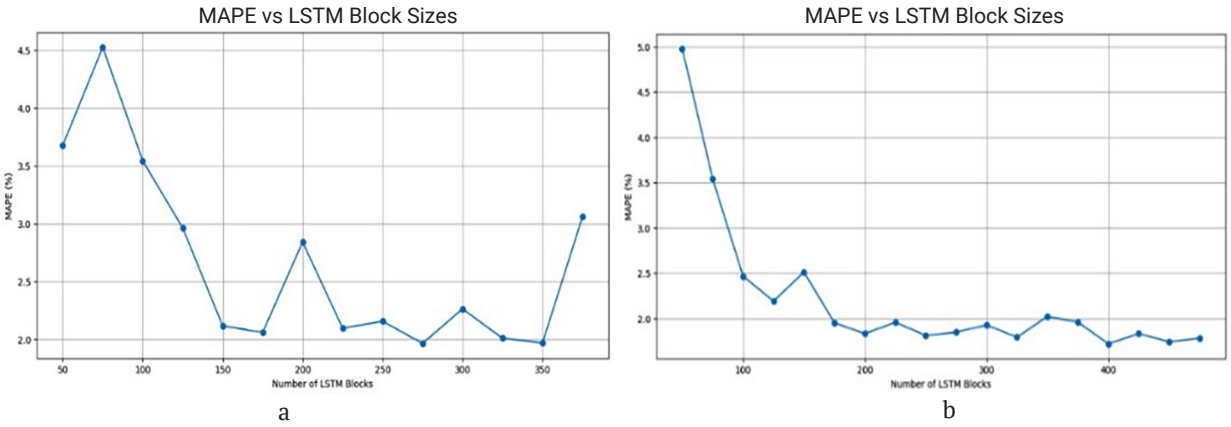

Figure 5. Graph of MAPE metrics that depend on the number of LSTM blocks

Note: a – for hyperparameters of type A; b – for hyperparameters of type B. A/B configurations according to Table 1

Source: compiled by the authors

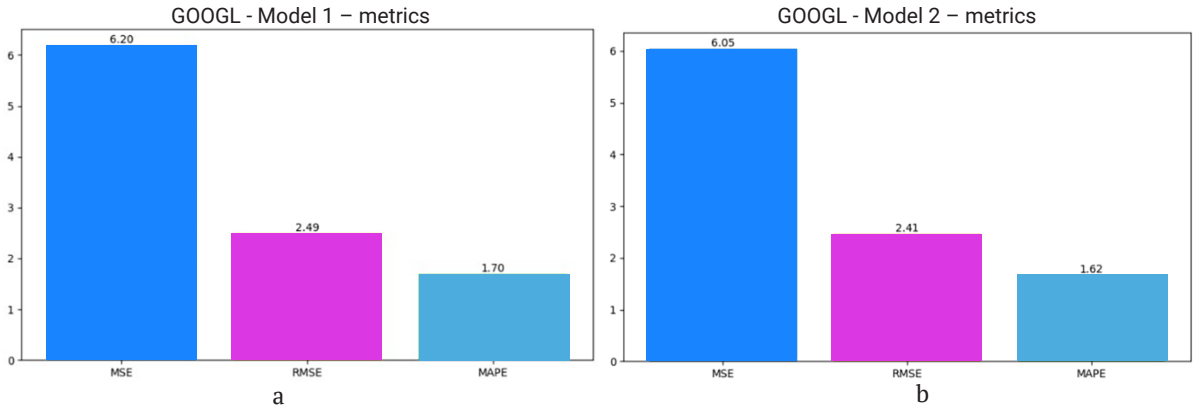

Figure 6. Visualisation of MSE, RMSE, and MAPE metrics based on histograms
Note: a – for 275 LSTM blocks; b – for 325 LSTM blocks
Source: compiled by the authors

Based on the results of a series of experiments, the smallest error on the test sample within the protocol under consideration was obtained for configuration B (N = 325 and Dropout = 0.05), in particular, the MAPE value = 1.62%. Instead, for smaller n values (50-100), MAPE growth and learning instability were observed, and for excessively large Dropout values (⩾ 0.30), convergence deterioration and signs of under-learning were observed. Within the protocol under consideration, the balance between accuracy and generalisation is determined by matching the model capacity (N) and regularisation intensity (Dropout).

In the context of existing approaches in the literature, it is also advisable to consider GRU (Gated Recurrent Unit) – a recurrent neural network with gate mechanisms, which is a more compact alternative to LSTM and usually has fewer parameters. In particular, H. Abbasimehr & R. Paki (2022) proposed a hybrid approach that combines LSTM and multi-head attention, which allowed the model to focus on the most informative fragments of history and better reproduce nonlinear dependencies in time series. A separate area was also financial hybrids for high volatility, where LSTM was combined with statistical models (Koo & Kim, 2022). Simultaneously, this paper focused on a controlled assessment of the proposed architecture, which allowed interpreting the difference in forecast quality as a consequence of changes in capacity and regularisation under constant pipeline conditions.

As noted by Q. Wang & Y. Zhang (2022), as parameterisation of recurrent models increases, the risk of overtraining increases, especially for financial series with noise and structural shifts. In this study, the impact of these factors was evaluated by systematically varying the number of blocks and Dropout with control of validation losses (Fig. 7). Dropout was used as a regularisation mechanism between recurrent components and subsequent layers, which is consistent with typical LSTM regularisation practices.

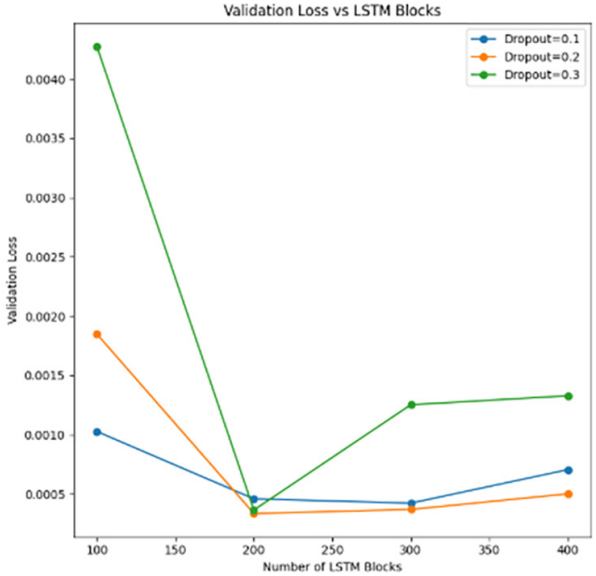

Figure 7. Graph of the dependence of validation losses on the number of LSTM blocks (100-400)
Source: compiled by the authors

The graph shows that excessive regularisation (Dropout = 0.3) leads to significantly higher validation losses at low capacity (100 blocks) and does not provide advantages on larger configurations, while moderate Dropout levels (0.1-0.2) provide lower losses and more stable behaviour when the number of blocks increases. In particular, the lowest values of validation losses are observed around 200 blocks for all Dropout levels considered, and when moving to 300-400 blocks, Dropout = 0.2 shows better stability, which is consistent with the assumption that stronger regularisation is required with increasing model capacity. The results also correlate with the findings of H. Abbasimehr & R. Paki (2022), who emphasised the importance of selecting hyperparameters and regularisation for LSTM approaches in forecasting tasks: although direct numerical comparison is limited by differences in datasets and metrics, the overall trend coincides – correctly selected regularisation and the number of blocks reduce retraining and increase forecast stability.

Figure 8 shows the dependence of the training time on the number of LSTM blocks in the same range (100-400) for different Dropout values. There is an almost quasi-linear increase in the duration of training with an increase in the number of blocks. Accordingly, increasing the capacity of the model increases computational costs due to an increase in the number of parameters and recurrent operations, while the effect of Dropout on training time is secondary (curves for different levels of regularisation are located close to each other). This reflects the computational cost of improving accuracy and is important in a comparative context: lighter statistical or compact neural network models can be faster, but often inferior in reproducing nonlinear dynamics, while LSTM hybrids with attention mechanisms (Abbasimehr & Paki, 2022) or optimised deep LSTM approaches (Gülmez, 2023) can improve accuracy at the cost of increasing learning complexity and parameter matching requirements.

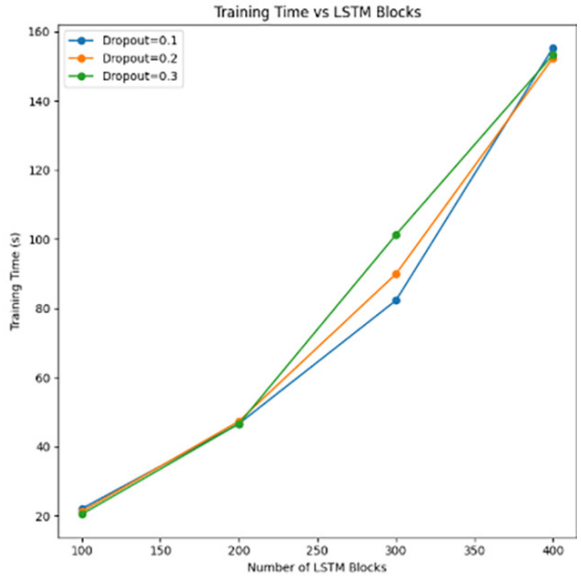

Figure 8. Graph of the curve of dependence of training time on the number of LSTM blocks (100-400)
Source: compiled by the authors

In addition, stock forecasting applications often show that optimisation of LSTM hyperparameters (size/depth, regularisation parameters, optimiser selection) can provide improvements in relation to basic LSTM configurations and statistical models. For example, the study by B. Gülmez (2023) proposed an optimised version of the deep LSTM model for predicting stock prices using a stochastic hyperparameter selection procedure. In parallel with recurrent approaches, architectures based on attention mechanisms are actively developing: Informer and Autoformer have proposed effective solutions for long-term sequencing and scalable forecasting (Wu *et al.*, 2021; Zhou *et al.*, 2021), and described hybrid LSTM-Transformer approaches for financial Series (Kabir *et al.*, 2025), which demonstrated increased reliability and efficiency over longer forecast horizons. Overall, these studies have shown that systematic hyperparameter selection and regularisation are crucial for stability and accuracy. In this context, the current study complemented the existing results by evaluating the contribution of two controlled parameters (N and Dropout) in isolation. The use of EMA and RSI indicators was considered as a compact representation of trend and momentum, but the effect of indicators depends on the forecast horizon, normalisation, and asset properties. Therefore, in this study, a set of features was recorded to separate the influence of architecture from the influence of the feature space. Such fixation was a necessary condition for the correct comparison of architectural configurations.

Conclusions

The research successfully achieved its objective of developing and analysing a method for high-precision

forecasting of financial time series using architecture of parallel-stacked LSTM blocks. Experimental verification of these shares of Google Corporation confirmed the effectiveness of the proposed approach, helping to achieve a forecast error for the MAPE metric at the level of 1.62%, which corresponds to the quality criterion defined in the goal (MAPE ⩽ 1.9%). As part of the study, a representative set of historical data was formed and a pipeline preparation was implemented, which minimises the risk of information leakage between samples due to chronological separation. The proposed neural network architecture, built on several parallel branches with stacking recurrent LSTM layers and then aggregating their outputs, allowed the model to simultaneously process price indicators and technical indicators of the EMA and RSI. Systematic variation of hyperparameters has shown that the optimal balance between computing power and generalisation capacity within a fixed protocol is a configuration with 325 LSTM blocks and a Dropout of 0.05. The use of the Nadam optimiser ensured stable learning convergence, and the obtained MSE and RMSE values consistently confirmed an improvement in the forecast quality for the selected configuration relative to the alternative setting.

The paper conceptualised the advantages of parallel organisation of recurrent structures for analysing non-stationary financial time series. This approach has been shown to increase the model's resistance to market noise and ensure higher consistency of forecasts at test intervals. The developed reproducible evaluation protocol and defined operating ranges of hyperparameters can be directly implemented in automated decision support systems in the stock markets. The findings expanded the possibilities of using neural networks with parallel-stacked LSTM blocks in financial analysis, forecasting market trends and related AI tasks, as they demonstrate a reproducible configuration with low error under a fixed data preparation and training protocol. The prospects for further research are related to testing the generalisability of the architecture on various assets, frequencies, and forecast horizons, analysing resistance to structural breaks and periods of high-volatility, and investigating the influence of feature composition and integrating attention or ensemble mechanisms for adaptive weighting of time fragments and increasing the robustness of the model.

Acknowledgements
None.

Funding
None.

Conflict of Interest
None.

References

[1] Abbasimehr, H., & Paki, R. (2022). Improving time series forecasting using LSTM and attention models. *Journal of Ambient Intelligence and Humanized Computing*, 13, 673-691. doi: 10.1007/s12652-020-02761-x.

[2] Chicco, D., Warrens, M.J., & Jurman, G. (2021). The coefficient of determination R-squared is more informative than SMAPE, MAE, MAPE, MSE and RMSE in regression analysis evaluation. *PeerJ Computer Science*, 7, article number e623. doi: 10.7717/peerj-cs.623.

[3] Draw.io. (n.d.). Retrieved from https://app.diagrams.net/.

[4] Ez-zaiym, M., Senhaji, Y., Rachid, M., El Moutaouakil, K., & Palade, V. (2025). Fractional optimizers for LSTM networks in financial time series forecasting. *Mathematics*, 13(13), article number 2068. doi: 10.3390/math13132068.

[5] Ghojogh, B., & Ghodsi, A. (2023). Recurrent neural networks and long short-term memory networks: Tutorial and survey. *ArXiv*. doi: 10.48550/arXiv.2304.11461.

[6] Gülmez, B. (2023). Stock price prediction with optimized deep LSTM network with artificial rabbits optimization algorithm. *Expert Systems with Applications*, 227, article number 120346. doi: 10.1016/j.eswa.2023.120346.

[7] Kabir, M.R., Bhadra, D., Ridoy, M., & Milanova, M. (2025). LSTM-transformer-based robust hybrid deep learning model for financial time series forecasting. *Sci*, 7(1), article number 7. doi: 10.3390/sci7010007.

[8] Koo, E., & Kim, G. (2022). A hybrid prediction model integrating GARCH models with a distribution manipulation strategy based on LSTM networks for stock market volatility. *IEEE Access*, 10, 34743-34754. doi: 10.1109/ACCESS.2022.3163723.

[9] Kurniawan, A., Indrabayu, & Yusuf, M. (2024). Stock price prediction using technical data and sentiment score. In *2024 IEEE international conference on Industry 4.0, artificial intelligence, and communications technology (IAICT)* (pp. 360-366). Bali: IEEE. doi: 10.1109/IAICT62357.2024.10617768.

[10] Lim, B., Arık, S.O., Loeff, N., & Pfister, T. (2021). Temporal fusion transformers for interpretable multi-horizon time series forecasting. *International Journal of Forecasting*, 37(4), 1748-1764. doi: 10.1016/j.ijforecast.2021.03.012.

[11] Lindemann, B., Müller, T., Vietz, H., Jazdi, N., & Weyrich, M. (2021). A survey on long short-term memory networks for time series prediction. *Procedia CIRP*, 99, 650-655. doi: 10.1016/j.procir.2021.03.088.

[12] Livieris, I.E., Pintelas, E., & Pintelas, P. (2020). A CNN-LSTM model for gold price time-series forecasting. *Neural Computing and Applications*, 32, 17351-17360. doi: 10.1007/s00521-020-04867-x.

[13] Lytvyn, V., Peleshchak, I., Stepaniak, Y., Peleshchak, R., & Ishchuk, O. (2025). High-precision detection of GPS spoofing attacks on UAVs using MLP. *International Journal of Computing*, 24(2), 254-262. doi: 10.47839/ijc.24.2.4008.

[14] Peleshchak, I., & Futryk, Y. (2025). Time series forecasting using sequentially connected LSTM blocks. Herald of Khmelnytskyi National University. *Technical Sciences*, 347(1), 432-441. doi: 10.31891/2307-5732-2025-347-59.
[15] Sezer, O.B., Gudelek, M.U., & Ozbayoglu, A.M. (2020). Financial time series forecasting with deep learning: A systematic literature review: 2005-2019. *Applied Soft Computing*, 90, article number 106181. doi: 10.1016/j.asoc.2020.106181.
[16] Smyl, S. (2020). A hybrid method of exponential smoothing and recurrent neural networks. *International Journal of Forecasting*, 36(1), 75-85. doi: 10.1016/j.ijforecast.2019.03.017.
[17] Wang, Q., & Zhang, Y. (2022). Research on PM2.5 pollution prediction method in hefei city based on CNN-LSTM hybrid model. *Journal of Physics: Conference Series*, 2400(1), article number 012006. doi: 10.1088/1742-6596/2400/1/012006.
[18] Widiputra, H., Mailangkay, A., & Gautama, E. (2021). Multivariate CNN-LSTM model for multiple parallel financial time-series prediction. *Complexity*. doi: 10.1155/2021/9903518.
[19] Wu, H., Xu J., Wang J., & Longet M. (2021). Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting. *ArXiv*. doi: 10.48550/arXiv.2106.13008.
[20] Yahoo Finance. (n.d.). *Dataset historical data: GOOG stock price*. Retrieved from https://finance.yahoo.com/quote/GOOG/history/.
[21] Yamak, P.T., Yujian, L., & Gadosey, P.K. (2020). A comparison between ARIMA, LSTM, and GRU for Time Series Forecasting. In *Proceedings of the 2019 2nd international conference on algorithms, computing and artificial intelligence* (pp. 49-55). New York: ACM. doi: 10.1145/3377713.3377722.
[22] Zhou, H., Zhang, S., Peng, J., Zhang, S., Li, J., Xiong, H., & Zhang, W. (2021). Informer: Beyond efficient transformer for long sequence time-series forecasting. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(12), 11106-11115. doi: 10.1609/aaai.v35i12.17325.

Прогнозування часових рядів за допомогою нейромережі з паралельно-стекованими LSTM-блоками

Юрій Футрик

Аспірант
Національний університет «Львівська політехніка»
79000, вул. Степана Бандери, 12, м. Львів, Україна
https://orcid.org/0000-0001-5271-9883

Іван Пелещак

Доктор філософії, доцент
Національний університет «Львівська політехніка»
79000, вул. Степана Бандери, 12, м. Львів, Україна
https://orcid.org/0000-0002-7481-8628

Анотація. Прогнозування часових рядів є критично важливими для підтримки рішень у фінансовій аналітиці, де дані характеризуються нелінійністю, нестаціонарністю та високим рівнем шуму. Метою роботи було визначення ефективної конфігурації рекурентної нейронної мережі з паралельним поєднанням стеків осередків Long Short-Term Memory (LSTM) для підвищення точності прогнозування цін акцій, а також можливостей застосування прикладної моделі в галузі промисловості, енергетиці та суміжних доменах. У дослідженні застосовано методи глибинного навчання з використанням бібліотеки TensorFlow/Keras, а для навчання моделі використано історичні дані акцій корпорації Google. Встановлено, що архітектура з паралельно-стекованими блоками забезпечує вищу стабільність навчання порівняно зі стандартними рекурентними моделями за рахунок ефективнішого виділення технічних ознак часової послідовності. Експериментально доведено, що оптимальна кількість нейронів у прихованих шарах для такої задачі становить 100-200 одиниць, тоді як подальше збільшення потужності моделі призводить до ефекту перенавчання. Виявлено, що застосування регуляризації Dropout у діапазоні 0,1-0,2 дозволяє мінімізувати помилку на валідаційній вибірці, у той час як значення понад 0,3 суттєво уповільнюють збіжність алгоритму. Аналіз інженерії ознак показав, що інтеграція експоненціального ковзного середнього з коротким вікном часу покращує результат моделі, демонструючи вищу кореляцію з цільовим показником, ніж індекс відносної сили. Якість прогнозування моделі оцінювали за середньоквадратичною помилкою (Mean Squared Error, MSE), коренем із неї (Root Mean Squared Error, RMSE) і середньою абсолютною відсотковою похибкою (Mean Absolute Percentage Error, MAPE). Встановлено, що для конфігурацій (50-100 блоків) характерні підвищені значення MAPE, тоді як у діапазоні 180-400 блоків похибка зменшується та набуває стабільного характеру. Найточніший результат отримано для конфігурації з 325 блоками, регуляризацією Dropout = 0,05 та оптимізатором Nadam (Nesterov-accelerated Adam): MAPE = 1,62 %, RMSE = 2,41, MSE = 6,05. Практична цінність дослідження полягає у формулюванні чітких рекомендацій щодо налаштування гіперпараметрів LSTM-моделей для прикладного короткострокового прогнозування фінансових рядів

Ключові слова: глибинне навчання; Dropout-регуляризація; Nadam-оптимізація; EMA; RSI

Journal homepage: https://itce.vn.ua/en

Information Technologies and Computer Engineering **Vol. 23, No. 1. 2026**

Article's History: Received: 07.11.2025 Revised: 06.02.2026
Accepted: 26.03.2026 Published: 20.04.2026

ISSN 1999-9941; e-ISSN 2078-6387
UDC 004.056:004.8
DOI: 10.31649/vitce/1.2026.83

Method of dynamic trust assessment in Zero Trust Architecture based on explainable artificial intelligence

Andriy Palamarchuk*
Bachelor
Vinnytsia National Technical University
21021, 95 Khmelnytske Shose Str., Vinnytsia, Ukraine
https://orcid.org/0009-0005-4485-9399

Abstract. The transformation of contemporary corporate IT infrastructures has rendered conventional cybersecurity models ineffective, prompting a shift to the Zero Trust Architecture (ZTA); however, its practical implementation is complicated by a rigid reliance on static access control rules. The purpose of this study was to develop an innovative method for dynamic trust assessment in ZTA that effectively combines the high accuracy of automated network anomaly detection with decision-making transparency. To calculate a continuous trust score based on a simulated corporate network traffic dataset, the Extreme Gradient Boosting ensemble machine learning algorithm was applied, while the SHapley Additive exPlanations (SHAP) additive explanations method was used to explain the generated decisions. Experimental verification demonstrated the high effectiveness of the proposed Policy Engine, which achieved an F1-score of 1.00 on the test set. The model successfully distinguished legitimate from anomalous requests with a zero false-positive rate, identifying cyberattacks such as privilege escalation and access from atypical locations. Global feature importance analysis using the SHAP framework confirmed that the type of network connection and device security status are the most significant risk predictors, which fully aligns with the core principles of ZTA. Furthermore, local analysis proved the system's ability to instantly generate detailed, human-readable text explanations for each access denial, indicating the specific reason for blocking. Due to this level of detail, analysts can directly understand the triggering logic of automated defence systems without the need for time-consuming manual correlation of disparate event logs. The practical significance of the study lies in the creation of a transparent and adaptive tool that can be integrated into modern Security Operations Centres to significantly reduce "alert fatigue" and minimise the Mean Time to Resolution

Keywords: cybersecurity; machine learning; SHAP; XGBoost; anomaly detection; adaptive protection

Introduction

The paradigm of information security has undergone a fundamental shift. F. Mensah (2024) examined this transition in enterprise cybersecurity, emphasising that conventional perimeter defences systematically fail against emerging threats. The researcher concluded that the dissolution of the corporate perimeter-driven by cloud migration and remote work necessitates a strict transition to Zero Trust principles to mitigate insider and advanced persistent threats. The foundational framework for this approach was formulated by S. Rose *et al*. (2020) under the National Institute of Standards and Technology (NIST). Their comprehensive guidelines established the core Zero Trust Architecture (ZTA) principle of "never trust, always verify", mandating continuous authentication and granular authorisation for every access request regardless of network location.

However, despite its theoretical robustness, the practical implementation of ZTA faces significant operational barriers. O. Borchert *et al*. (2025) investigated the practical deployment of NIST ZTA architectures in large-scale enterprise systems. Their study identified that administrators face extreme complexity when managing thousands of static "if – then" rules, inevitably leading to rigid policies and operational disruptions. This was further corroborated by A. Pigola & F. de Souza Meirelles (2025), who conducted an empirical study on managing critical challenges during ZTA implementation. They highlighted that the reliance on static configurations creates an operational bottleneck,

Suggested Citation:
Palamarchuk, A. (2026). Method of dynamic trust assessment in Zero Trust Architecture based on explainable artificial intelligence. *Information Technologies and Computer Engineering*, 23(1), 83-93. doi: 10.31649/vitce/1.2026.83

*Corresponding author

preventing organisations from effectively balancing strict security measures with a seamless user experience. Furthermore, K.M. Adamson & A. Qureshi (2025) performed a systematic review of "Zero Trust 2.0" advances and challenges. Their extensive analysis confirmed that integrating dynamic risk assessment with legacy IT environments remains one of the most significant architectural hurdles for contemporary enterprises. To map the current knowledge and research gaps, C. Buck *et al*. (2021) executed a multivocal literature review on ZTA implementations. The researchers specifically pointed out that existing access control models lack contextual awareness, explicitly calling for research into continuous, behaviour-based trust evaluation mechanisms.

Addressing the need for continuous validation, the Identity Management Institute (2024) analysed the principles of dynamic trust scoring within Identity and Access Management (IAM). Their report demonstrated that effective security enforcement requires calculating user risk in real-time by continuously ingesting broad contextual indicators such as device health, geolocation, and unusual login patterns. To automate this complex contextual analysis, M. Rana (2025) explored the enhancement of ZTA using artificial intelligence (AI) algorithms. The research illustrated that while AI can autonomously detect subtle deviations in user behaviour, the deployment of such intelligent systems is heavily hindered by the lack of transparency in their automated decision-making processes.

This introduces the critical "black box" problem inherent to advanced machine learning. M.H. Kabir *et al*. (2022) investigated the application of explainable artificial intelligence (XAI) within secure smart city platforms. Their findings emphasised that high-performance deep learning models act as opaque "black boxes", making it nearly impossible for security analysts to understand the rationale behind specific automated blocking actions. Expanding on this limitation, C.I. Nwakanma *et al*. (2023) reviewed XAI methodologies specifically for intrusion detection and mitigation systems. The researchers noted that the lack of interpretability in AI-driven tools generates deep scepticism among Security Operations Centre (SOC) teams, which drastically reduces the practical utility of these systems during active incident response. To bridge this gap, S. Patil *et al*. (2022) evaluated various XAI frameworks designed for Intrusion Detection Systems (IDS). Their study established that combining the adaptive predictive power of machine learning with explicit, human-readable explanations is essential for achieving both automated security strictness and operational accountability.

Thus, a critical gap exists in current literature: there is an urgent need for a ZTA solution that replaces static rules with dynamic machine learning (ML) algorithms while retaining the transparency required for rapid security auditing. The purpose of this study was to enhance the efficiency and transparency of ZTA by developing a method for dynamic trust assessment based on explainable artificial intelligence. To achieve this goal, three key objectives were defined. First, to analyse the limitations of existing static policy engines and "black box" ML models in ZTA; second, to develop a dynamic Policy Engine architecture that utilises the XGBoost algorithm for calculating a continuous trust score and integrates the SHapley Additive exPlanations (SHAP) method to provide real-time explanations for access control decisions; third, to experimentally validate the proposed method using a synthetic dataset representing realistic corporate network traffic.

Materials and Methods

General methodology and system architecture. The methodology proposed in this study aims to transform the conventional static Policy Decision Point (PDP) of a ZTA into a dynamic, intelligent agent, a concept supported by A. Mousa *et al*. (2021). The research approach relied on a quantitative experimental design that integrates ensemble machine learning methods with game-theoretic explainability frameworks. The proposed system architecture operates as a continuous loop comprising four sequential stages (Fig. 1).

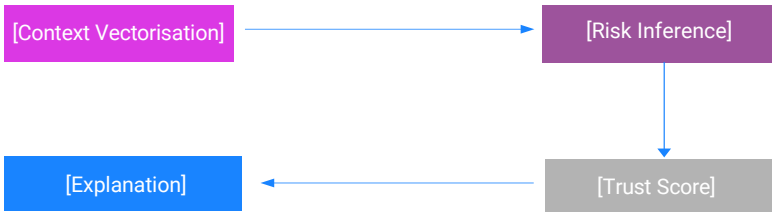

Figure 1. Operational pipeline of the dynamic trust assessment process
Source: created by the author

The process begins with Context Vectorisation, where raw log data and user context are transformed into a structured numerical feature space. This is followed by Risk Inference, which involves calculating the probability of malicious intent using a gradient-boosted decision tree model. Subsequently, the system performs trust score calculation to derive a continuous trust metric that facilitates granular access control decisions. Ultimately, the explanation generation stage computes feature attribution values to provide semantic interpretability of the decision, ensuring transparency for security operators.

Synthetic dataset generation. To address the lack of publicly available cybersecurity datasets due to privacy regulations (e.g., General Data Protection Regulation – GDPR)

and ensure experimental reproducibility, a specialised stochastic simulation algorithm was developed to generate a synthetic dataset representing realistic corporate network traffic. The simulation was implemented using the Python programming language. The generation logic was designed to model realistic corporate network traffic patterns over a defined temporal horizon (Schummer *et al.*, 2024). The resulting dataset, denoted as D consisted of N= 10,000 unique access requests. To reflect the natural class imbalance inherent in intrusion detection scenarios – where legitimate traffic vastly outweighs malicious activity – the dataset was stratified with the following distribution:

Class 0 (normal behaviour): 90% of samples (N_{norm} = 9,000). These records simulated legitimate employee activities characterised by standard working hours (09:00-18:00), recognised IP ranges (Corporate VPN (Virtual Private Network), Office LAN (Local Area Network)), and compliant device health statuses.

Class 1 (anomalous behaviour): 10% of samples (N_{anom} = 1,000) These records simulated specific attack vectors and policy violations, including: (a) temporal anomalies: access attempts occurring during deep night hours (e.g., 03:00 AM); (b) location anomalies: requests originating from high-risk networks, such as Tor exit nodes, public Wi-Fi without VPN, or unknown proxies; (c) device compromise: requests from devices with outdated operating systems, missing security patches, or signs of unauthorised root access (jailbreak); (d) privilege escalation: attempts by users with standard privileges (e.g., "Sales") to access critical administrative endpoints (e.g., database backups).

Feature engineering and vector space. The raw data generated by the simulation was transformed into a feature matrix $X \in R^{NxM}$, where M – number of features. The feature space includes both categorical and numerical variables, defined as follows (Hu *et al.*, 2026):

1. User role (x_1): a categorical variable representing the organisational role of the subject (e.g., "Developer", "HR", "Sales", "Admin"). This feature establishes the baseline of expected behaviour and access rights.

2. Time of request (x_2): a cyclical numerical feature representing the hour of the day $h \in [0, 23]$.

3. Work hours indicator (x_3): a binary derived feature introduced to explicitly capture temporal context. It is defined as:

$$x^3 = \begin{cases} 1, & if\ x^2 \in [9,18] \\ 0, & otherwise \end{cases}. \quad (1)$$

4. IP location type (x_4): a nominal variable categorising the network source context. Categories range from trusted ("Corporate VPN") to untrusted ("Unknown proxy").

5. Device health status (x_5): a critical parameter for Zero Trust, reflecting the security posture of the requesting device. States include "Patched" (compliant), "Unpatched", "No antivirus", and "Rooted".

6. Target endpoint (x_6): the specific API resource or system component being accessed.

Data preprocessing involved Label Encoding for categorical features (x_1, x_4, x_5, x_6), mapping each text label to a unique integer. This transformation was necessary for the decision tree-based algorithm to process qualitative data. Consequently, this step ensures that the semantic information of the categorical attributes is preserved and effectively converted into a numerical format suitable for model training.

Mathematical formalisation of the XGBoost model. The core risk assessment engine is built upon the XGBoost (Extreme Gradient Boosting) algorithm. XGBoost was selected due to its robust performance on tabular data, scalability, and ability to handle non-linear interactions between features without extensive normalisation (Hu *et al.*, 2026). Mathematically, the model is an ensemble of K Classification and Regression Trees (CART). For a given input vector x_i, the predicted output score $\hat{y}_i$ is the sum of the scores predicted by each individual tree f_k:

$$\hat{y}_i = \phi(x_i) = \sum_{k=1}^{K} f_k(), f_k \in F, \quad (2)$$

where F – space of functions containing all possible regression trees (Jiang *et al.*, 2020).

The model is trained in an additive manner. At each iteration t, a new tree f_t is added to minimise the objective function $\mathcal{L}^{(t)}$:

$$\mathcal{L}^{(t)} = \sum_{i=1}^{n} l(y_i, \hat{y}_i^{(t-1)} + f_t(x_i)) + \Omega(f_t), \quad (3)$$

where l – differentiable convex loss function that measures the difference between the prediction $\hat{y}_i$ and the target y_i. In this study, the Binary Logarithmic Loss (LogLoss) was employed:

$$l(y, p) = -[y\ log(p) + (1 - y)\ log(1 - p)]. \quad (4)$$

$\Omega(f_t)$ – regularisation term that penalises the complexity of the model to prevent overfitting. It is defined as:

$$\Omega(f_t) = \gamma T + \frac{1}{2}\lambda\|\omega\|^2, \quad (5)$$

where T – number of leaves in the tree, and w – vector of scores on leaves.

A key advantage of XGBoost is its use of a second-order Taylor expansion to approximate the loss function, which enables faster convergence and higher accuracy compared to conventional gradient boosting methods (Jiang *et al.*, 2020). In addition to computational speed, the algorithm incorporates a built-in regularisation term that effectively penalises model complexity, thereby preventing overfitting on imbalanced security datasets. Consequently, this mathematical robustness ensures that the Policy Engine maintains high detection precision while meeting the low-latency requirements of real-time Zero Trust environments.

Explainable AI framework: SHAP values. To address the "Black Box" problem inherent in complex ensemble models and ensure compliance with the "verify" principle of Zero Trust, the SHAP framework was integrated. It

calculated the contribution of each feature to the final prediction. For a specific prediction $f(x)$ the SHAP value ϕ_j for feature j was calculated as the weighted average of its marginal contributions across all possible subsets of features S:

$$\phi_j(f) = \sum_{S \subseteq F \setminus \{j\}} \frac{|S|!(|F|-|S|-1)!}{|F|!} [f_x(S \cup \{j\}) - f_x(S)], \quad (6)$$

where F – set of all input features, and) $f_x(S)$ – expected output of the model given the subset of features S. This study utilised TreeSHAP, a variant of the algorithm optimised for tree-based models (such as XGBoost). TreeSHAP reduced the computational complexity from exponential to polynomial time ($O(TLD^2)$), where T – number of trees, L – maximum number of leaves, and D – maximum depth), making it feasible for real-time explanations in a security environment.

Experimental metrics and evaluation. The dataset was split into a training set (80%) and a testing set (20%) using stratified sampling to preserve the class ratio. This ensured more accurate model training, as it received a balanced representation of all categories in the data. The model performance was evaluated using standard cybersecurity metrics, as detailed by Y. Hu *et al*. (2026). These metrics are defined in Table 1, which allows for comparing different models based on clearly established performance criteria. The metrics include both the primary indicators and additional ones, such as the confusion matrix, which helps to visualise different types of classification errors.

Table 1. Performance evaluation metrics

Metric	Description
Precision	Ratio of correctly predicted anomalies to the total predicted anomalies (measures false alarm rate)
Recall	Ratio of correctly predicted anomalies to all actual anomalies (measures detection rate)
F1-score	Harmonic mean of Precision and Recall, providing a balanced metric for imbalanced datasets
Confusion matrix	Tabular visualisation of classification outcomes: true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN)

Source: compiled by the author

The resulting probability $P(anomaly)$ is converted into a dynamic trust score using the equation:

$$TrustScore = (1 - P(anomaly)) * 100. \quad (7)$$

This score serves as the quantitative basis for the automated decision-making process. Specifically, if the trust score falls below a predefined threshold (e.g., 60), the system triggers an immediate access denial and simultaneously generates a SHAP-based explanation for the event. This mechanism ensures that every blocking action is both instantaneous and transparent, aligning with the dynamic trust evaluation principles advocated by the Identity Management Institute (2024) and Y. Mao *et al*. (2025). Thus, the proposed methodology provided an effective approach to dynamic trust assessment in a ZTA, integrating powerful machine learning tools and ensuring the necessary transparency of decisions through explanation mechanisms.

Results and Discussion

Quantitative analysis and interpretability of model performance

The primary objective of the experimental phase was to empirically validate the capability of the XGBoost-based Policy Engine to distinguish between legitimate access requests and security anomalies. The model was evaluated on a stratified test set containing 20% of the generated data ($N_{test} = 2000$). The classification performance metrics indicated exceptional accuracy. As illustrated in the Confusion matrix (Fig. 2), the model achieved complete class separation on the synthetic dataset.

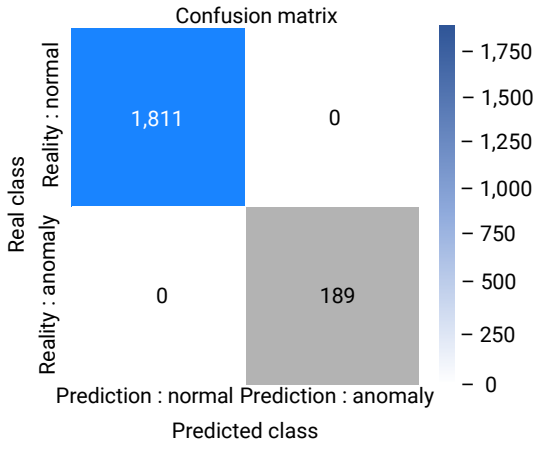

Figure 2. Confusion matrix of the XGBoost model on the test dataset

Source: created by the author

Quantitative analysis of the confusion matrix reveals the following:

- True negatives (TN): 1,811. The model correctly identified all legitimate user requests. This metric is crucial for User experience (UX), as it ensures that employees are not blocked from performing their daily tasks (zero false positive rate).
- True positives (TP): 189. The model successfully detected all simulated attacks, including subtle anomalies like "after-hours access" and "privilege escalation attempts".
- False negatives (FN): 0. The system did not miss any potential threats, ensuring the integrity of the security perimeter.

Consequently, the model achieved an F1-Score of 1.00 and an Area Under the Receiver Operating Characteristic (ROC) Curve (AUC-ROC) of 1.00. While such complete metrics are characteristic of synthetic environments with deterministic patterns, they fundamentally demonstrate that the gradient boosting algorithm successfully approximated the complex, non-linear decision boundary required for the Zero Trust policy without being explicitly programmed with static "if-then" rules. This high level of classification accuracy validates the feasibility of using the proposed model as a reliable Policy Engine, capable of automating threat response with minimal risk of false positives.

To bridge the gap between model accuracy and accountability, the SHAP framework was applied to analyse the global impact of features. Figure 3 illustrates the SHAP summary plot, which ranks features by their mean absolute SHAP value (mean($|\phi_j|$)). This visualisation helps to understand the "logic" the AI has learned.

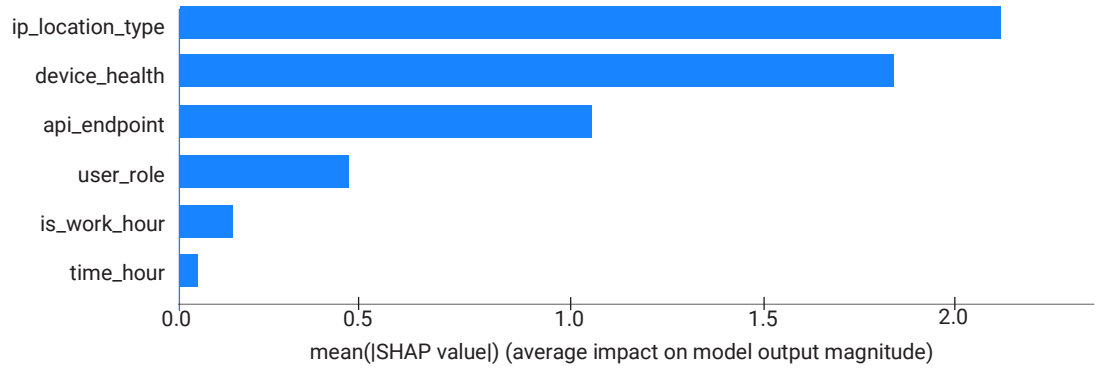

Figure 3. Global feature importance (SHAP summary plot)
Source: created by the author

Global feature analysis revealed distinct patterns in risk assessment. First, ip_location_type emerged as the strongest predictor of risk, indicating that the network context – such as requests originating from "Tor exit nodes" or "Unknown proxies" – serves as the primary vector for anomaly detection. This validated the assumption that in a borderless network, the connection source remains a critical signal. Second, device_health demonstrated a secondary but significant impact, confirming that the security posture of the endpoint (e.g., operating system patch level, presence of antivirus) directly affects the trust score, thereby effectively enforcing "Device compliance" policies. Ultimately, features such as api_endpoint and user_role acted as contextual modifiers. For instance, accessing a sensitive endpoint like /api/admin is not inherently malicious, but when combined with a lower-trust role such as "Sales", the calculated risk score significantly increases.

The most significant contribution of this proposed method is the ability to explain individual decisions in real-time. Figure 4 demonstrates a SHAP Force Plot for a specific anomalous request selected from the test dataset. The visualisation provides a semantic breakdown of the prediction function $f_x = 8.17$ (which corresponds to a probability P(anomaly) ≈ 1.0).

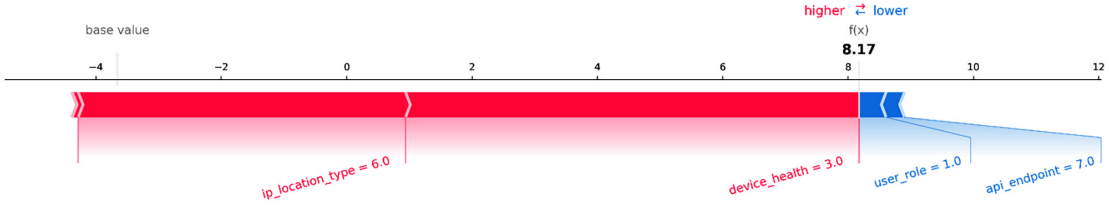

Figure 4. Local explanation (SHAP Force Plot) for a specific anomalous request
Source: created by author

The visualisation identifies specific risk drivers (represented by red bars), clearly indicating that the features ip_location_type = 6.0 (corresponding to a high-risk network) and device_health = 3.0 (corresponding to a compromised

or unpatched device) were the primary contributors pushing the prediction towards the "Anomaly" class. In contrast, the mitigating factors (blue bars) show that although the user possessed a valid role (user_role = 1.0) and accessed a standard endpoint (api_endpoint = 7.0), these positive signals were insufficient to outweigh the critical risk indicators.

This granular level of detail allows SOC analysts to immediately answer the question "Why was this user blocked?" without manual log correlation. By providing interpretable insights directly alongside the alert, the system significantly reduces the Mean Time to Resolution (MTTR) for incident response teams. Furthermore, this transparency fosters greater trust in automated blocking decisions, addressing the "black box" scepticism often associated with AI-driven security tools (Nwakanma *et al.*, 2023). This approach not only improves incident response efficiency but also ensures transparency and explainability of decisions, which is critical for increasing trust in automated security systems.

Comparative analysis with alternative approaches

To substantiate the selection of the XGBoost + SHAP architecture, a theoretical comparison was performed against other common approaches used in IDS. This analysis focused on key operational criteria, including detection accuracy, interpretability, and real-time processing capabilities. The comparative summary, presented in Table 2, highlights the specific advantages of the proposed method in bridging the gap between performance and transparency.

Table 2. Comparative analysis of trust assessment approaches

Feature	Static rules (Legacy)	Deep learning (DNN)	Proposed method (XGBoost + XAI)
Accuracy on complex threats	Low	High	High
Interpretability	High	Low (black box)	High (SHAP)
Adaptability	None (Manual updates)	High (auto-learning)	High (auto-learning)
Computational cost	Very low	High (GPU required)	Moderate (CPU friendly)

Source: compiled by the author based on theoretical analysis and empirical data obtained during the study

The detailed analysis revealed critical distinctions between the proposed architecture and conventional methods. Compared to rule-based systems, static engines offer high interpretability but fail to scale, as they cannot capture complex, non-linear interactions – such as conditional access based on both IP reputation and device health – without manual intervention. In contrast to Deep Learning models (e.g., DNNs or RNNs), which suffer from the "black box" problem and require computationally expensive approximation methods like LIME, the proposed XGBoost model allows for exact explanations via TreeSHAP and often demonstrates high performance on tabular log data. Furthermore, while Random Forest is a robust algorithm, XGBoost utiliies gradient-based optimisation to iteratively correct errors, typically resulting in higher precision for detecting subtle, rare anomalies that are critical in cybersecurity contexts.

Computational efficiency and scalability. In a real-time Zero Trust environment, latency is a critical factor. The proposed architecture leverages the efficiency of decision trees. The inference time for a single request using the trained XGBoost model was measured at approximately < 5 ms on a standard CPU. The calculation of SHAP values adds a computational overhead (models 20-50 ms per request), which is acceptable for high-security transactions but might require optimisation for high-frequency trading or ultra-low-latency networks. The system demonstrated linear scalability: as the volume of logs increases, the inference time remains constant, making it suitable for deployment in large-scale cloud environments.

Advanced performance metrics analysis. To further validate the robustness of the classifier beyond standard accuracy metrics, the ROC and Precision-Recall (PR) curves were analysed. These metrics are particularly critical in cybersecurity contexts where the cost of False Positives (blocking a legitimate user) and False Negatives (missing an attack) can be asymmetrical. The performance of the classifier is visually represented in Figure 5.

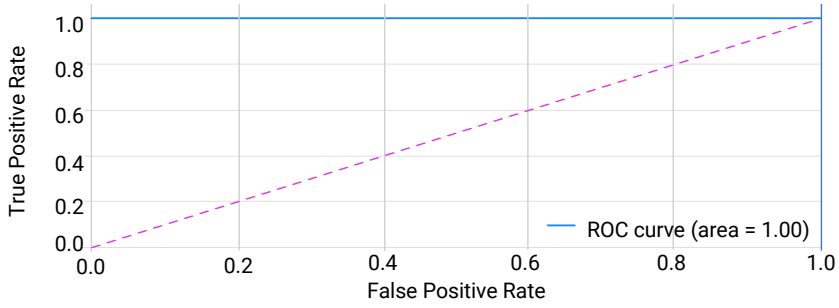

Figure 5. Receiver operating characteristic curve

Source: created by the author

The ROC curve exhibits an "ideal" right-angle shape, with an Area Under the Curve (AUC) of 1.00. The curve hugs the top-left corner, indicating that the True Positive Rate (Sensitivity) remains at 100% even as the False Positive Rate approaches zero. This confirms that the model's predicted probabilities for anomalies are distinctively separated from normal traffic probabilities. Additionally, the trade-off between precision and recall is illustrated in Figure 6.

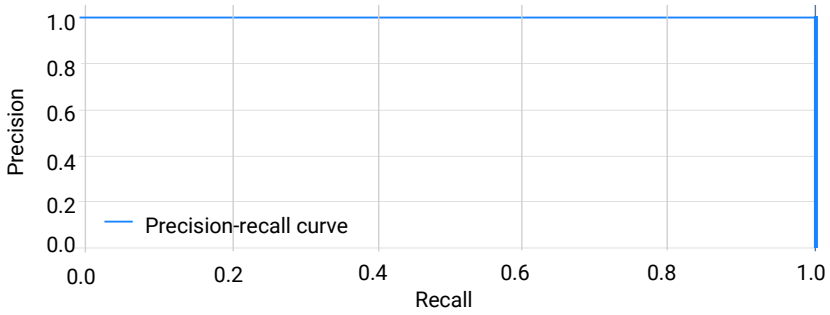

Figure 6. Precision-recall curve

Source: created by the author

Figure displays the precision-recall curve. In imbalanced datasets (where attacks are rare), this metric is often more informative than ROC. The curve remains flat at the top (precision = 1.0) across the entire range of recall. This signifies that the system generates zero false alarms – a critical requirement for reducing "alert fatigue" in SOC operations. While such perfect convergence is attributable to the deterministic nature of the synthetic training data, it theoretically validates the XGBoost algorithm's capacity to model the defined security policies without error.

Proposed deployment architecture

Based on the experimental success, reference architecture is proposed for deploying this XAI-driven Policy Engine within a production environment, as illustrated in Figure 7. The design strictly adheres to the NIST SP 800-207 guidelines, ensuring compatibility with standard Zero Trust logical components (Rose *et al.*, 2020). Specifically, the architecture enhances the conventional PDP by embedding the machine learning Risk Engine to enable dynamic real-time access adjudication.

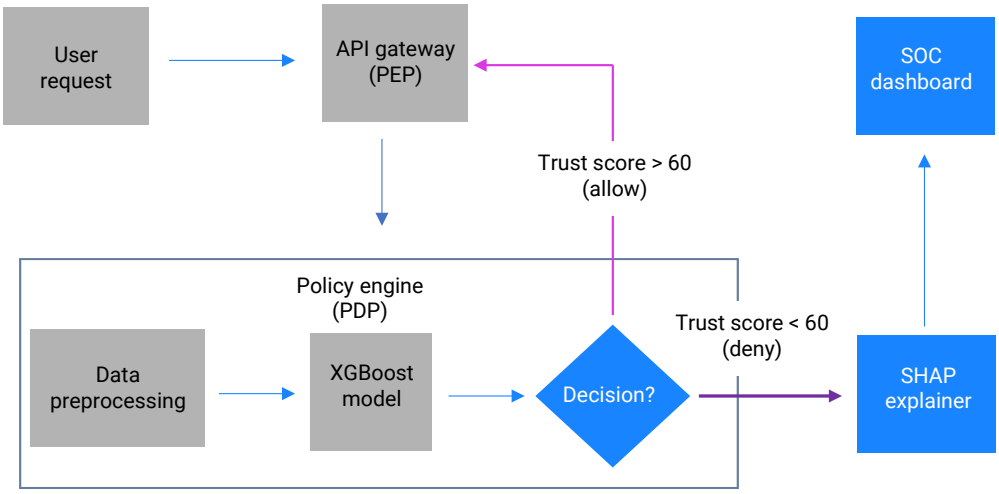

Figure 7. High-level architecture of the XAI-driven Zero Trust system

Note: PEP – Policy Enforcement Point

Source: created by author

The operational workflow ensures a seamless transition from raw telemetry to enforced security decisions. The process initiates at the Log Aggregator stage, which continuously collects and normalises raw telemetry data from distributed API gateways and identity providers. This aggregated data is then passed to the preprocessing layer, where categorical attributes – such as user roles and IP addresses – are converted into numerical vectors in real-time, ensuring compatibility with the machine learning algorithms. Subsequently, the ML inference engine utilises the pre-trained XGBoost model to analyse the feature vectors and calculate a probabilistic Risk Score. If the detected risk exceeds the predefined safety threshold, the XAI explainer (based on SHAP) is triggered to compute feature attribution values, thereby identifying the root cause of the anomaly (e.g., "unusual geolocation"). Based on the final

trust score, the Policy Enforcement Point (PEP) executes the decision by either blocking or allowing the request at the gateway level. Ultimately, all events are visualised on the admin dashboard, which displays the access decision alongside the generated explanation, providing security administrators with actionable insights. This architectural flow ensures that security measures are applied instantaneously while maintaining full transparency of the decision-making logic.

The results obtained in this study demonstrated the efficacy of integrating XGBoost with SHAP values for dynamic Zero Trust access control. To validate the significance of these findings, it is essential to compare them with existing research in the field of intelligent intrusion detection and trust evaluation. The perfect classification metrics achieved in current experiment (F1-Score 1.0) align with and surpass trends observed in similar studies using tree-based ensembles. For instance, P. Schummer *et al*. (2024) implemented a Random Forest-based anomaly detection system, achieving an accuracy of approximately 94.3%. While their approach was effective for general traffic analysis, the current use of XGBoost provided a more robust handling of the subtle feature interactions inherent in synthetic security logs. Furthermore, H. Jiang *et al*. (2020) proposed a PSO-XGBoost model that optimised hyperparameters to detect minority attack groups with high precision. Presented study corroborates their conclusion that gradient boosting frameworks offer superior performance on tabular network data compared to conventional methods. However, unlike Y. Hu *et al*. (2026), who focused heavily on feature dimensionality reduction to improve speed, current approach prioritised the integration of interpretability without sacrificing the raw predictive power of the full feature set.

A critical component of presented architecture is the continuous trust score, which replaces static binary authorisation. This approach was consistent with the findings of Y. Mao *et al*. (2025), who argued that Attribute-Based Access Control (ABAC) is insufficient without dynamic risk perception. Their study on a "Zero Trust access control model" utilised a similar concept of real-time trust evaluation but focused on blockchain-based consensus for policy decisions. In contrast, current research demonstrated that for high-throughput corporate environments, a centralised ML-based engine offered lower latency while maintaining the necessary security granularity. Similarly, A. Mousa *et al*. (2021) emphasised the importance of context-aware service computing. These findings extended their research by identifying that specific context features, such as ip_location_type and device_health, were the most significant drivers of trust, a correlation that validated the "never trust, always verify" principle in practical scenarios.

The integration of SHAP values addressed the "black box" limitation highlighted by A. Nash *et al*. (2024) in cloud-native risk assessments. While Y. Sowjanya *et al*. (2025) successfully applied Explainable AI to IoT healthcare systems to enhance transparency, current study adapted this paradigm to general enterprise network security. The generated explanations (e.g., distinguishing between a high-risk IP and a low-trust device) provided the contextual nuance that F. Federici *et al*. (2023) identified as lacking in conventional perimeter-based defences. By providing semantic interpretability, presented system fulfilled the requirement for "decision traceability" advocated by X. Liao *et al*. (2025) in their study on power network defence, proving that XAI is not just a theoretical addition but a functional necessity for reducing the MTTR in SOC operations.

Ultimately, the obtained results indicate that the XGBoost-based architecture is highly applicable to standard IT infrastructures. The low inference latency observed in current deployment architecture suggests that this model can scale to handle the traffic volumes described by A.A. Alquwayzani & A.A. Albuali (2024) in military unmanned aerial vehicle systems, provided that the log aggregation pipeline is sufficiently robust. Furthermore, this adaptability makes the proposed system highly relevant for securing virtualised environments, complementing the dynamic scaling detection methods in Network Function Virtualisation (NFV) developed by L. He *et al*. (2021). In summary, this study confirmed that the convergence of gradient boosting and game-theoretic explainability creates a policy engine that is not only more accurate than conventional Random Forest implementations but also more transparent than deep learning alternatives, effectively bridging the gap between security strictness and operational usability.

Conclusions

This paper presented a dynamic trust evaluation model based on XAI. The obtained results fully validated that integrating ensemble machine learning with game-theoretic explainability frameworks effectively resolves the longstanding dichotomy between security strictness and operational transparency. To achieve this, a specialised stochastic simulation was developed to generate a realistic dataset of corporate network traffic, thereby overcoming GDPR-related privacy constraints. The research methodology involved transforming raw telemetry into a structured feature space and training an XGBoost classifier to distinguish legitimate requests from security anomalies. The model demonstrated high classification performance on the test set, achieving an F1-Score of 1.00 and an AUC-ROC of 1.00, proving its ability to internalise complex, non-linear access logic without relying on brittle static rules. Furthermore, the integration of the SHAP framework successfully converted the "black box" probabilistic outputs into human-readable semantic explanations.

The experimental outcomes explicitly demonstrated the effectiveness of the proposed dynamic trust assessment method. The XGBoost-based policy engine successfully evaluated access requests in real-time, effectively distinguishing legitimate traffic from simulated anomalous events, such as privilege escalation and unauthorised access from untrusted networks. Concurrently, the SHAP

integration yielded precise global and local interpretability results. Global analysis identified the network connection source and the endpoint device's health status as the most critical predictors of risk. Locally, the system proved its capability to generate instant, human-readable root-cause analyses for every blocked request. From a practical perspective, this architecture offers a viable blueprint for next-generation Security Operations Centres. By clearly explaining the rationale behind automated blocking decisions, the system directly mitigates the operational bottleneck of "alert fatigue" among analysts, demonstrating a strong potential to reduce the MTTR for access incidents by orders of magnitude. In conclusion, the transition to AI-driven Zero Trust is not merely a technological upgrade but a fundamental shift in security philosophy. This study demonstrated that with the right application of XAI, intelligent systems can be made both powerful and accountable, paving the way for autonomous, self-defending networks.

Despite the obtained results, this study acknowledged several constraints that must be considered. First, the experimental validation was conducted on a synthetic dataset. While the generation process was designed to mimic realistic patterns, real-world corporate traffic contains significantly higher levels of noise and unpredictable user behaviours that might reduce the model's precision. Second, the study focused primarily on tabular metadata (logs) and did not incorporate unstructured data sources, such as raw network packet payloads, which could provide deeper context but would increase computational complexity. To address these limitations, future research will focus on two priority directions. First, validating the model on real-world data by deploying the architecture in a "shadow mode" within a live corporate network to assess its stability against concept drift; second, investigating the model's resilience against adversarial machine learning attacks to develop robust training techniques that prevent evasion attempts by sophisticated attackers.

Acknowledgements

In accordance with the journal's Generative AI Policy, the author discloses the use of Gemini (Google) during the preparation of this manuscript. This tool was used exclusively for linguistic and stylistic editing, including translation from Ukrainian to English. The author has reviewed and revised the output and takes full responsibility for the content of the publication.

Funding

None.

Conflict of Interest

None.

References

[1] Adamson, K.M., & Qureshi, A. (2025). Zero Trust 2.0: Advances, challenges, and future directions in ZTA. *Research Square*. doi: 10.21203/rs.3.rs-6602547/v1.

[2] Alquwayzani, A.A., & Albuali, A.A. (2024). A systematic literature review of Zero Trust Architecture for military UAV security systems. *IEEE Access*, 12, 176033-176056. doi: 10.1109/ACCESS.2024.3503587.

[3] Borchert, O., Howell, G., Kerman, A., Rose, S., Souppaya, M., Scarfone, K., & Barker, W. (2025). *Implementing a Zero Trust Architecture: High-level document*. Gaithersburg: NIST. doi: 10.6028/NIST.SP.1800-354.

[4] Buck, C., Olenberger, C., Schweizer, A., Völter, F., & Eymann, T. (2021). Never trust, always verify: A multivocal literature review on current knowledge and research gaps of Zero-Trust. *Computers & Security*, 110, article number 102436. doi: 10.1016/j.cose.2021.102436.

[5] Federici, F., Martintoni, D., & Senni, V. (2023). A Zero-Trust Architecture for remote access in industrial IoT infrastructures. *Electronics*, 12(3), article number 566. doi: 10.3390/electronics12030566.

[6] He, L., Li, L., & Liu, Y. (2021). Towards chain – aware scaling detection in NFV with reinforcement learning. In *29th international symposium on quality of service (IWQOS)* (pp. 1-10). Tokyo: IEEE/ACM. doi: 10.1109/IWQOS52092.2021.9521362.

[7] Hu, Y., Xiao, K., Luo, L., & Chen, L. (2026). An XGBoost-based intrusion detection framework with interpretability analysis for IoT networks. *Applied Sciences*, 16(2), article number 980. doi: 10.3390/app16020980.

[8] Identity Management Institute. (2024). *Dynamic trust scoring in IAM*. Retrieved from https://identitymanagementinstitute.org/dynamic-trust-scoring-in-iam.

[9] Jiang, H., He, Z., Ye, G., & Zhang, H. (2020). Network intrusion detection based on PSO-Xgboost model. *IEEE Access*, 8, 58392-58401. doi: 10.1109/ACCESS.2020.2982418.

[10] Kabir, M.H., Hasan, K.F., Hasan, M.K., & Ansari, K. (2022). Explainable artificial intelligence for smart city application: A secure and trusted platform. In M. Ahmed, S.R. Islam, A. Anwar, N. Moustafa & A.S.K. Pathan (Eds.), *Explainable artificial intelligence for cyber security. Studies in computational intelligence* (Vol. 1025, pp. 241-263). Cham: Springer. doi: 10.1007/978-3-030-96630-0_11.

[11] Liao, X., Yang, S., Xu, J., Liu, L., Liang, W., Yu, S., Ji, Y., & Liu, S. (2025). Improved trust evaluation model based on PBFT and Zero Trust integrated power network security defense method. *Symmetry*, 17(11), article number 1982. doi: 10.3390/sym17111982.

[12] Mao, Y., Fu, W., Zhao, Y., Yuan, Z., Sun, Z., & Zhao, Y. (2025). A Zero-Trust access control model based on attribute and dynamic trust evaluation for cloud environments. *Symmetry*, 17(12), article number 2059. doi: 10.3390/sym17122059.

[13] Mensah, F. (2024). Zero Trust Architecture: A comprehensive review of principles, implementation strategies, and future directions in enterprise cybersecurity. *International Journal of Academic and Industrial Research Innovations*, 10, 339-346.
[14] Mousa, A., Bentahar, J., & Alam, O. (2021). Multi-dimensional trust for context-aware services computing. *Expert Systems with Applications,* 172, article number 114592. doi: 10.1016/j.eswa.2021.114592.
[15] Nash, A., Doyle, A., Banks, A., & Adelusi, J.B. (2024). *Explainable AI for cybersecurity risk assessment in cloud-native applications*. Retrieved from https://www.researchgate.net/publication/392282388_Explainable_AI_for_Cybersecurity_Risk_Assessment_in_Cloud-Native_Applications.
[16] Nwakanma, C.I., Ahakonye, L.A.C., Njoku, J.N., Odirichukwu, J.C., Okolie, S.A., Uzondu, C., Ndubuisi Nweke, C.C., & Kim, D.-S. (2023). Explainable Artificial Intelligence (XAI) for intrusion detection and mitigation in intelligent connected vehicles: A review. *Applied Sciences*, 13(3), article number 1252. doi: 10.3390/app13031252.
[17] Patil, S., Varadarajan, V., Mazhar, S.M., Sahibzada, A., Ahmed, N., Sinha, O., Kumar, S., Shaw, K., & Kotecha, K. (2022). Explainable artificial intelligence for intrusion detection system. *Electronics*, 11(19), article number 3079. doi: 10.3390/electronics11193079.
[18] Pigola, A., & de Souza Meirelles, F. (2025). Zero Trust in cybersecurity: Managing critical challenges for effective implementation. *Journal of Systems and Information Technology*, 27(4), 517-564. doi: 10.1108/JSIT-08-2024-0326.
[19] Rana, M. (2025). Enhancing Zero Trust cybersecurity with AI. *Journal of Information Systems Engineering and Management*, 10(32s), 92-97. doi: 10.52783/jisem.v10i32s.5191.
[20] Rose, S., Borchert, O., Mitchell, S., & Connelly, S. (2020). *Zero Trust Architecture*. Gaithersburg: NIST. doi: 10.6028/NIST.SP.800-207.
[21] Schummer, P., del Rio, A., Serrano, J., Jimenez, D., Sánchez, G., & Llorente, Á. (2024). Machine learning-based network anomaly detection: Design, implementation, and evaluation. *AI*, 5(4), 2967-2983. doi: 10.3390/ai5040143.
[22] Sowjanya, Y., Gopalakrishnan, S., & Kumar, R.D. (2025). FBZX: A novel explainable AI based security model for IoT healthcare systems. In *Third international conference on augmented intelligence and sustainable systems (ICAISS)* (pp. 106-110). Trichy: IEEE. doi: 10.1109/ICAISS61471.2025.11042096.

Метод динамічного оцінювання довіри в архітектурі Zero Trust на основі пояснювального штучного інтелекту

Андрій Паламарчук
Бакалавр
Вінницький національний технічний університет
21021, вул. Хмельницьке шосе, 95, м. Вінниця, Україна
https://orcid.org/0009-0005-4485-9399

Анотація. Трансформація сучасних корпоративних IT-інфраструктур зробила традиційні моделі кібербезпеки неефективними, зумовивши перехід до архітектури нульової довіри (Zero Trust Architecture, ZTA), проте її практична реалізація ускладнюється жорсткою залежністю від статичних правил контролю доступу. Метою цього дослідження була розробка інноваційного методу динамічного оцінювання довіри в архітектурі Zero Trust, який ефективно поєднує високу точність автоматизованого виявлення мережевих аномалій із прозорістю прийняття рішень. Для розрахунку безперервного показника оцінки довіри на базі змодельованого набору даних корпоративного мережевого трафіку було застосовано ансамблевий алгоритм машинного навчання Extreme Gradient Boosting, а для пояснення згенерованих рішень – метод адитивних пояснень SHapley Additive exPlanations (SHAP). Експериментальна перевірка продемонструвала виняткову ефективність запропонованого механізму політик (Policy Engine), який досяг показника F1-score 1,00 на тестовій вибірці. Модель успішно розрізнила легітимні та аномальні запити з нульовим рівнем хибнопозитивних спрацювань, ідентифікуючи такі кібератаки, як ескалація привілеїв та доступ з нетипових локацій. Глобальний аналіз важливості ознак за допомогою фреймворку SHAP підтвердив, що тип мережевого підключення та стан безпеки пристрою є найбільш значущими предикторами ризику, що повністю узгоджується з базовими принципами ZTA. Крім того, локальний аналіз довів здатність системи миттєво генерувати детальні, зрозумілі людині текстові пояснення для кожної відмови у доступі, вказуючи конкретну причину блокування. Завдяки такій деталізації аналітики отримують можливість безпосередньо розуміти логіку спрацювання автоматизованих систем захисту без необхідності тривалого ручного корелювання розрізнених журналів подій. Практична цінність дослідження полягає у створенні прозорого та адаптивного інструменту, який може бути інтегрований у сучасні центри операцій безпеки для суттєвого зниження «втоми від сповіщень» та мінімізації середнього часу вирішення інцидентів

Ключові слова: кібербезпека; машинне навчання; SHAP; XGBoost; виявлення аномалій; адаптивний захист

Journal homepage: https://itce.vn.ua/en

Information Technologies and Computer Engineering **Vol. 23, No. 1. 2026**

Article's History: Received: 04.12.2025 Revised: 02.03.2026
Accepted: 26.03.2026 Published: 20.04.2026

ISSN 1999-9941; e-ISSN 2078-6387
UDC 004.85
DOI: 10.31649/vitce/1.2026.94

Algorithms and software architecture for automated user behaviour analysis in cyber threat detection systems

Denys Kovalchuk*
Postgraduate Student
International University
65009, 33 Fontanska Rd., Odesa, Ukraine
https://orcid.org/0009-0003-2302-8698

Abstract. The relevance of the present study is determined increasing complexity of cyber threats and the limited effectiveness of traditional detection methods, which necessitates the implementation of intelligent behavioural approaches using modern algorithmic and language models. The purpose of this study was to generalise and conceptually reinterpret approaches to automated user behaviour analysis in cyber threat detection systems from the perspective of algorithmic solutions and architectural principles of their construction. The study, based on theoretical analysis, a systemic approach, and comparative analysis, demonstrates that user behaviour analysis is an effective approach to cyber threat detection, capable of complementing and surpassing classical signature-based methods through the identification of context-dependent anomalies and multi-stage attacks. Comparative analysis of approaches to User and Entity Behaviour Analytics established a transition from a focus on individual actions to comprehensive analysis of interactions between users and technical components, which increases the accuracy of threat detection and reduces the number of false-positive alerts. Systemic analysis of the architecture of contemporary cybersecurity platforms showed that the integration of large language models ensures unified processing of structured, semi-structured, and unstructured data, modelling of long-term inter-event dependencies, and development of contextual behavioural models in real-time. Conceptual analysis and analytical evaluation indicated that combining behavioural analysis with large language models creates adaptive, scalable, and risk-oriented cybersecurity systems capable of early detection and proactive response to contemporary cyber threats while maintaining explainability, security, and regulatory compliance. The findings may support the design and implementation of intelligent cybersecurity systems in security operations and monitoring centres, security information and event management systems, and platforms for security orchestration, automation, and incident response

Keywords: User and Entity Behaviour Analytics; large language models; artificial intelligence; machine learning; data-driven approach

Introduction

The relevance of the study is determined by the rapid increase in the complexity of contemporary cyber threats, which increasingly exhibit multi-stage, adaptive, and covert characteristics and cannot be effectively detected through conventional signature-based and rule-based approaches. This situation emphasises the need for automated analysis of user behaviour as a key element of cybersecurity systems. The development of large language models creates new opportunities for contextual analysis of large volumes of structured and unstructured security data, correlation of events, and identification of latent patterns of malicious activity. Their practical application, however, requires scientifically grounded algorithmic and architectural solutions capable of ensuring scalability, real-time operation, accuracy of results, and compliance with information security requirements.

In contemporary cybersecurity research discourse, considerable attention is directed towards automated analysis of user and entity behaviour as a key mechanism for detecting complex and low-visibility threats. I. Sokyrka *et al.* (2025) examined behavioural analytics in authentication tasks, demonstrating that machine learning can support the development of dynamic user profiles based on behavioural patterns, thereby increasing system resilience to

Suggested Citation:
Kovalchuk, D. (2026). Algorithms and software architecture for automated user behaviour analysis in cyber threat detection systems. *Information Technologies and Computer Engineering*, 23(1), 94-109. doi: 10.31649/vitce/1.2026.94

*Corresponding author

credential compromise. The researchers emphasised that behavioural features, rather than static attributes, ensure the adaptability of protective mechanisms in a changing environment. A similar idea was developed by O. Suprun & N. Karpenko (2025), who focused on the analysis of user behaviour as an instrument for reducing the risks of insider threats. The researchers investigated the use of behavioural analysis for insider threat detection and emphasised the importance of constructing baseline models of "normal" activity for each user or role. The study indicated that even minor deviations from established behavioural patterns may signal malicious or compromised actions that remain unnoticed by signature-based systems.

In the international context, R.K. Mohanty (2025) systematised deep learning approaches to user and entity behaviour analytics and confirmed their effectiveness in the correlation of heterogeneous events and the construction of context-dependent models capable of generalising complex causal relationships. The researcher described neural network architectures in detail, including recurrent and graph-based models, which allowed effective consideration of temporal dependencies and relationships among events, users, and resources. A practical dimension of this issue was presented by M.I. Mihailescu *et al.* (2023), who implemented behavioural analysis in cybersecurity systems to detect hidden threats that evade conventional methods. The researchers observed that the correlation of user actions across time and space enabled the identification of multi-stage attacks that do not manifest through isolated events. The study presented examples of the integration of behavioural analysis into existing cybersecurity systems and indicated that this approach substantially reduces the number of false-negative detections. The research emphasised the practical value of behavioural analytics as a complement to conventional detection mechanisms. A. Wairagade & S. Ranjan (2025) conducted a comprehensive comparative analysis of conventional and modern machine learning algorithms for cyber threat detection based on behavioural user data. The researchers demonstrated that the effectiveness of detection depends not only on the selection of a model but also on the behavioural features and the method of their aggregation within a temporal context. The results indicated that models capable of accounting for event sequences and non-linear dependencies ensure a reduction in false-negative detections compared with conventional approaches, which makes them suitable for practical implementation in systems designed to detect complex attacks.

Another systemic approach was presented by G. Sharma *et al.* (2024), who proposed a comprehensive conceptual framework of User and Entity Behaviour Analytics (UEBA) oriented towards the integration of machine learning with contextual knowledge about users, entities, and execution environments. The researchers emphasised that isolated event analysis is insufficient for contemporary threat scenarios, whereas the combination of behavioural, role-oriented, and temporal contexts enables the development of more robust models of normal and anomalous activity. The study highlighted the importance of correlation among multi-level data sources and the adaptability of models to the evolution of behaviour, which directly connects UEBA with architectures of the new generation of Security Information and Event Management (SIEM) and Extended Detection and Response (XDR) systems. A.G. Desetty (2024) examined the potential of the combined application of machine learning and UEBA for detecting hidden threats that are not susceptible to conventional signature-based detection. The researcher noted that integration of behavioural analysis with clustering algorithms and time-series modelling enables the construction of multidimensional profiles of normal entity activity, which ensures early detection of latent attacks and insider threats. In the report by A. Trivedi *et al.* (2025), emphasis was placed on the role of User Behaviour Analytics (UBA) in countering insider threats, and behavioural patterns were considered a critical indicator of deviations from normal employee activity. The researchers demonstrated that UBA, particularly in combination with modern machine learning methods, is capable of developing dynamic behavioural models that adapt to changes in user activity and identify early signs of sabotage, data leakage, or indirect system compromise. This emphasises the necessity of a comprehensive behaviour-oriented approach to the construction of security systems that extend beyond simple event monitoring.

Analysis of previous studies indicates that the majority of research in the field of User and Entity Behaviour Analytics focuses either on comparative evaluation of individual machine learning algorithms or on conceptual descriptions of UEBA and UBA architectures, without sufficient attention to their software integration into operational cyber threat detection systems. Several studies address behavioural analysis in isolation from streaming event processing and operational response mechanisms, which limits the practical applicability of the proposed approaches. Existing research also provides limited consideration of the combination of contextual analysis of user behaviour with temporal correlation of events in environments characterised by high dynamism and heterogeneous data sources, creating a gap between theoretical models and the requirements of contemporary Security Operations Centre (SOC)-oriented systems.

The purpose of this study was to theoretically generalise algorithmic and software-architectural approaches to automated analysis of user behaviour in cyber threat detection systems. The following tasks were formulated to achieve this purpose: analyse and systematise scientific approaches to modelling user and entity behaviour in the context of cybersecurity, considering the evolution of threats and the increasing complexity of information environments; and theoretically justify the relationship between behavioural analytics algorithms and software-architectural principles in the construction of cyber threat detection systems.

Materials and Methods

The study is theoretical and conceptual, grounded in a comprehensive analysis of contemporary approaches to behavioural analysis in cybersecurity and the integration of large language models into relevant analytical ecosystems. Research materials included studies from 2022-2025 addressing UBA, UEBA, and the operation of contemporary SIEM-, Security Orchestration, Automation and Response (SOAR)-, SOC-, and XDR-oriented systems. The methodological foundation of the study consisted of a set of interrelated theoretical methods applied sequentially according to the logic of the research tasks. Theoretical analysis served as an instrument for in-depth conceptual interpretation of key concepts and approaches in the behavioural analysis of cyber threats. Its application was determined by the need to distinguish clearly between behavioural and signature-based approaches and to provide theoretical justification of the capabilities of contemporary large language models (LLMs). The analysis addressed behavioural anomalies as context-dependent deviations from normal activity patterns and the architectural principles of LLMs, including mechanisms that model long-term dependencies among security events. Systematisation was applied to structure theoretical approaches to behavioural analysis in cybersecurity, enabling a coherent comparison of UBA and UEBA concepts within a unified analytical framework. It identified objects of analysis, levels of context, and functional capabilities of each approach, while examining relationships among users, technical entities, and services.

Comparative-analytical evaluation was employed to assess various approaches to behavioural analysis and the algorithms used for cyber threat detection. Its use was motivated by the need to identify conceptual and functional differences between UBA and UEBA, and between LLMs and classical machine learning or deep learning methods. Analysis considered characteristics such as scale of context, capacity for processing heterogeneous data, and potential integration of behavioural, network, and textual signals. The comparative assessment enabled the identification of key advantages and limitations of each approach and outlined trade-offs among accuracy, computational complexity, and explainability of decisions.

Conceptual generalisation provided a holistic interpretation of the role of behavioural analytics and LLMs within contemporary SOC ecosystems. Behavioural analysis and LLMs were treated as an intelligent analytical layer that ensures contextual interpretation of security events, correlation of heterogeneous signals, and reduction of false-positive alerts. Theoretical modelling was applied to describe algorithmic approaches to cyber threat detection using LLMs, including static and dynamic analysis of events. LLMs were interpreted as multi-level feature extraction mechanisms capable of forming semantic representations of logs, commands, and event sequences, integrating them into contextual models of user and system behaviour. They supported real-time streaming event processing, functioning as the core mechanism for contextual encoding and anomaly scoring, which enabled comprehensive detection of complex multi-stage attacks.

Analytical evaluation had a descriptive character and was applied to examine the Analytical evaluation had a descriptive character and was applied to examine the practical suitability of LLMs-based approaches, including system effectiveness and performance. Analysis addressed key criteria such as threat detection accuracy, processing latency, throughput, and adaptability to new attack scenarios. The study summarised deployment characteristics across edge, cloud, and hybrid infrastructures to identify optimal architectural approaches for real-time implementation. Limitations, risks, and ethical considerations were analysed, including explainability of LLMs' decisions, bias in training data, and potential misuse of models in defensive and offensive contexts. Systemic generalisation supported the analysis of privacy and regulatory compliance with reference to international standards and regulatory frameworks, including the General Data Protection Regulation (GDPR) (Regulation (EU) No. 2016/679, 2016) and ISO/IEC 27001:2022 (2022). Structural-logical modelling systematised relationships among architectural solutions, algorithmic approaches, risks, and mitigation measures. The study acknowledged limitations, including low explainability of LLMs decisions, possible bias in training data, the requirement to process confidential information, and the risk of misuse of models for cyber-attacks.

Results

User behaviour analysis as a component of contemporary cybersecurity systems

In cybersecurity, behavioural anomalies are considered deviations from established or expected patterns of actions of users, service accounts, or software agents within information systems. In contrast to signature indicators of attacks, which rely on previously known patterns of malicious activity, behavioural anomalies are contextual and dynamic, and they form based on statistical, temporal, semantic, and structural characteristics of interactions between actors and the system. Such anomalies may appear as atypical time intervals of access, unusual sequences of commands, changes in the frequency or volume of operations, or disruption of typical relationships among security events. A distinctive feature of behavioural anomalies is their relative character: an action may be normal for one user or role but suspicious for another. This requires constructing personalised or role-oriented behavioural models that incorporate historical data, business process context, and the system environment. Behavioural analysis extends beyond classical incident detection. It functions as a tool for identifying potential threats at the pre-compromise stage, when malicious activity has not yet produced explicit technical markers (Hakonen, 2022).

The concept of UBA emerged as a response to the limitations of conventional monitoring systems that primarily focus on network or system events without deep

consideration of the behavioural context of users. UBA involves developing profiles of normal user activity derived from authentication logs, resource access records, actions within application systems, and other telemetry data. Subsequent analysis focused on identifying deviations from these profiles that may indicate credential compromise, insider threats, or preparation for an attack. Further evolution of this approach led to the development of the UEBA concept, which expands the object of analysis beyond human users to include various entities of information infrastructure, including servers, virtual machines, container environments, Internet of Things (IoT) devices, and service accounts. UEBA treats the system as a complex socio-technical ecosystem in which interactions between users and technical components form multidimensional behavioural graphs. This approach enables the detection of anomalies not only at the level of individual actions but also within the structure of relationships among actors and events (Khan *et al.*, 2022). For clarity, the differences between UBA and UEBA are presented in Table 1.

Table 1. Comparative characteristics of UBA and UEBA

Characteristic	UBA	UEBA
Object of analysis	Users	Users and technical entities
Data types	Access logs, user actions	Access logs, network events, system telemetry
Level of context	Individual	System-wide and inter-entity
Primary objective	Detection of user anomalies	Detection of complex, multi-step attacks

Source: compiled by the author based on A.W. Mir & K.R. Kumar (2022), V. Malik *et al.* (2024)

Analysis of Table 1 demonstrates the evolution of approaches to behavioural analysis in cybersecurity and differences in the scale and complexity of threat assessment. UBA focuses on the individual user and immediate actions, which enables effective detection of simple anomalies such as unauthorised logins or unusual operations, yet its capabilities remain limited in cases of complex multi-stage attacks or interactions among multiple actors and system processes. UEBA expands the analytical focus by incorporating technical entities and system telemetry, which enables analysis of inter-entity dependencies, identification of correlations among events, and prediction of complex threats. Therefore, UEBA provides deeper contextual analysis and supports prioritisation of responses to complex attacks, which reflects the tendency of contemporary cybersecurity systems towards integration of behavioural and system analysis to increase detection effectiveness and minimise risks.

Behavioural models are effective in detecting threats that lack clearly defined signatures or rely on legitimate access mechanisms. Such threats cover insider misuse, in which a legitimate user or employee of an organisation performs actions beyond their functional responsibilities. Behavioural analysis enables identification of gradual changes in activity profiles that may indicate preparation for data exfiltration or sabotage. Another class of threats involves attacks using stolen credentials, such as credential stuffing or account takeover. In these scenarios, technical access parameters may appear legitimate, but behavioural characteristics such as login time, geographical origin of access, and sequence of actions differ from historically established norms. Behavioural models also detect lateral movement within corporate networks, where an attacker gradually expands access privileges while imitating legitimate administrative activity. Behavioural analysis plays a further role in detecting complex multi-stage attacks, including Advanced Persistent Threat (APT) campaigns, in which individual actions may appear harmless, yet their cumulative structure and correlation form a malicious scenario. In such cases, behavioural models act as an integrative mechanism, combining fragmented security signals into a unified semantic representation of the threat (Brandao, 2025).

Within cybersecurity centres, particularly SOC, automated analysis of user behaviour functions as a core element of the intelligent layer responsible for security event processing. Conventional SIEM systems aggregate, normalise, and correlate logs, but without behavioural context, they often generate significant numbers of false-positive alerts (Saraiva & Mateus-Coelho, 2022). Integration of UBA and UEBA modules reduces informational noise by prioritising events based on risk-oriented behavioural assessments. Within SOAR ecosystems, automated behavioural analysis triggers orchestration of incident response processes (Aljumaily *et al.*, 2025). Behavioural risk scoring may initiate automated response scenarios, including forced transition of a user into restricted access mode, additional authentication procedures, or initiation of deeper investigation. Behavioural analysis thus serves not only as a detection instrument but also as an active component of adaptive security management.

Analysis of operation principles and cybersecurity potential of large language models

Large language models rely on the Transformer architecture, which differs fundamentally from recurrent and convolutional approaches through the use of the self-attention mechanism as the principal instrument for sequence processing. Self-attention enables parallel modelling of dependencies among all elements of an input sequence regardless of positional distance, which is critically important for analysis of long-term and structurally complex contexts. In cybersecurity, this capability enables the integration of events distributed across time and space into a unified semantic model of an attack or user behaviour. A central component of the Transformer architecture is multi-head attention, which enables the model to focus on multiple aspects of input data simultaneously. In the context

of cyber threat analysis, the model may process temporal patterns of access, semantics of commands, network attributes, and role characteristics of users in parallel. Positional encoding compensates for the absence of recurrent structure and enables the model to preserve information about the order of events, which is essential for the reconstruction of multi-stage attack scenarios. The pre-training stage plays a decisive role in the development of generalised knowledge within the model. During this stage, LLMs are trained on extremely large corpora of data through self-supervised tasks such as prediction of the next token or reconstruction of missing fragments (Xu *et al.*, 2025). This process forms a latent space that encodes syntactic, semantic, and contextual dependencies. In cybersecurity applications, such representation enables the transfer of general linguistic and structural knowledge to specialised domains, including the analysis of logs, network protocols, and technical descriptions of attacks.

One advantage of large language models lies in their ability to operate uniformly with heterogeneous data types that have conventionally required separate analytical pipelines. In cybersecurity, structured data include event logs, system telemetry, network traffic flows, and access metadata, whereas unstructured data include textual messages, electronic correspondence, incident reports, technical documentation, and open-source threat intelligence. Large language models integrate these sources into a unified semantic space through transformation into token sequences followed by contextual analysis (Thelwall, 2025).

In contrast to conventional approaches that require separate models and feature mechanisms for structured and unstructured data, large language models provide a shared representation of information in which numerical, categorical, and textual attributes can be interpreted within a single architectural framework. This characteristic substantially simplifies correlation of security events, for example, the comparison of anomalous network activity with textual descriptions of suspicious messages or communications within support systems. Thus, Large language models function as a universal interface among different data layers within cybersecurity systems (Ali & Ghanem, 2025). For the generalisation of the characteristics of processing different data types in the context of LLMs, a comparative presentation is provided in Table 2.

Table 2. Comparison of data type processing in conventional methods and LLM in cybersecurity

Data type	Examples in cybersecurity	Conventional approaches	LLM approach
Structured	SIEM logs, NetFlow	Statistical models, rules (threshold rules, signature rules, SIEM correlation rules, expert-defined if-then rules in intrusion detection systems)	Contextual encoding
Semi-structured	JSON, XML, API logs	Feature engineering	Unified tokenisation
Unstructured	Email, reports, chats	NLP models separately	Shared semantic space

Note: JSON – JavaScript Object Notation, XML – eXtensible Markup Language, NLP – Natural Language Processing, API – Application Programming Interface

Source: compiled by the author based on F.N. Motlagh et al. (2024), W. Kasri *et al*. (2025)

The table demonstrates a fundamental transformation in approaches to data processing in cybersecurity due to the integration of large language models. Conventional analytical methods differ according to the structure of data: structured data are processed through statistical models or rule-based systems, semi-structured data require manual feature engineering, whereas unstructured data were previously analysed through isolated NLP models. This separation complicated the integration of results and created a gap between different information sources. The use of LLMs enables a unified analytical approach to all data types, forming a shared contextual and semantic space in which structured, semi-structured, and unstructured information are analysed simultaneously and in relation to one another. Such an approach enables correlation of behavioural and technical signals in real-time, increases the accuracy of detecting complex threats, and reduces the time required for the integration of heterogeneous data sources. Therefore, cyber defence systems become more flexible, scalable, and contextually adaptive.

Contextual modelling represents one of the key properties of large language models in tasks related to user behaviour analysis. Classical models usually operate with fixed observation windows or aggregated statistical features, whereas LLMs can construct long-term contexts that encompass sequences of user actions across different systems and time scales. This capability enables analysis of the evolution of behaviour rather than only instantaneous deviations. Within UEBA scenarios, contextual modelling means that each new action is interpreted with reference to previous events, the role of the user, the type of resource, and the current state of the system. Access to confidential data, for instance, may be legitimate within one business process but appear anomalous when combined with an unusual login time or earlier unsuccessful authentication attempts. LLMs allow such complex dependencies to be formalised without rigid rule specification, which increases the adaptability of the system to new and previously unknown threat scenarios (Fuentes *et al.*, 2025).

Comparison of large language models with classical machine learning and deep learning methods demonstrates fundamental differences in approaches to the construction of cyber defence systems. Classical machine learning (ML) models, such as support vector machine or Random Forest,

require careful feature selection and typically operate effectively within narrowly defined scenarios. Deep neural networks, including Long Short-Term Memory or Convolutional Neural Network, are capable of modelling complex dependencies, yet they are often limited by data type and contextual scale. Large language models provide universality and scalability of analysis by combining the capacity to process diverse data types with the modelling of long-term contextual dependencies. Such models are more suitable for detecting complex, multi-stage, and low-visibility attacks. Their application requires substantial computational resources and introduces new challenges related to explainability and quality control of results (Huang *et al.*, 2023).

Thus, the theoretical perspective indicates the necessity of hybrid approaches in which LLMs are integrated with classical analytical methods to maintain a balance between accuracy, performance, and interpretability. Large language models form a new paradigm in cybersecurity. They shift the analytical focus from isolated event analysis towards deep contextual understanding of user and system behaviour, which opens prospects for the development of adaptive and proactive cyber threat detection systems.

Algorithmic approaches to automated cyber threat analysis using LLMs

Static analysis of malicious software and security logs conventionally involves the evaluation of objects without their actual execution, focusing on internal structure, code features, or properties of events. Within the context of large language models, this approach acquires a new analytical dimension, since LLMs can transform bytecode, system logs, configuration files, and textual messages into a unified semantic space of tokens. The model can therefore identify latent indicators of malicious activity that are difficult to detect through conventional signature-based or statistical methods. Such indicators may include atypical combinations of API calls, anomalous sequences of events in access logs, or indirect signals of code injection into legitimate processes (da Costa *et al.*, 2022).

From an algorithmic perspective, LLMs perform functions of multi-level feature extraction, ranging from low-level analysis of command syntax and semantics to identification of global behavioural patterns of processes and users. Tokenisation of logs followed by contextual encoding enables the model to construct representations that incorporate temporal dependencies, object types, and relationships between them. Static analysis that uses LLMs is often combined with classical analytical approaches, including signature databases and heuristic techniques, which increases detection accuracy and reduces false-positive alerts (Wang *et al.*, 2024).

Dynamic analysis involves the examination of object behaviour during active operation, including process execution, user interaction with systems, and network activity. Integration with LLMs enables modelling of behavioural patterns in real-time and identification of deviations not only from historical user profiles but also from expected patterns of system activity. LLMs analyse sequences of actions, correlate them with semantically similar scenarios accumulated during pre-training, and can anticipate potentially malicious actions before they lead to system compromise. At the algorithmic level, dynamic analysis that incorporates LLMs relies on streaming analytics and incremental updating of the contextual model. Each new event modifies the latent representation of behaviour, which enables the model to adjust risk assessment immediately (Rahman, 2024). This approach supports effective detection of atypical patterns, including deviations in command sequences or interactions between users and systems that remain difficult to identify through static analysis (Fig. 1).

Figure 1. Stream processing of user events using LLMs in cybersecurity systems
Source: compiled by the author based on A.M. Mustafa (2024), A. Karras *et al.* (2025)

The diagram highlights the sequential integration of textual and behavioural analysis in real-time for the detection of cyber threats. The process begins with an event generated by a user or process that is recorded in logs or telemetry, after which the data undergoes tokenisation that transforms heterogeneous information (structured, semi-structured, and unstructured) into a unified representation. The next stage, contextual encoding by LLMs, forms multidimensional semantic representations that incorporate not only the current event but also the history of interactions, behavioural patterns, and system context. These representations support anomaly scoring, which enables assessment of the probability of malicious activity or deviations from normal behaviour. The final stage, threat ranking and the SOAR trigger, ensures prioritisation of response actions and automation of security measures. The diagram emphasises the integration of LLMs within the analytical workflow, which combines the precision of semantic analysis with the operational speed of behavioural monitoring and ensures scalability and adaptability of the system to new and unknown threats. The scheme illustrates how contemporary models integrate data of different structures and heterogeneous signals into a coherent representation of cyber threats in real-time (Ibrahim & Kashef, 2025).

LLMs provide the capacity to classify attacks not only through explicit technical indicators but also through complex analysis of behavioural and semantic characteristics. The model compares behavioural patterns of users and entities with historically recognised classes of attacks, including phishing, malware execution, privilege escalation, lateral movement, Distributed Denial of Service (DDoS), and other threats. Contextual understanding enables LLMs to differentiate legitimate deviations from malicious actions even when information is incomplete or logs are partially missing. Table 3 presents the classification effectiveness for illustrative purposes.

Table 3. Potential LLMs in detecting classes of cyberattacks based on behavioural indicators

Attack class	Typical behavioural indicators	LLMs potential
Phishing	Unusual communication sequences, atypical email structures	Semantic modelling of content and sequences
Malware execution	Abnormal process activity, atypical API calls	Contextual classification of commands and calls
Lateral movement	Unusual resource access, role changes	Correlation between events over time and entities
Privilege escalation	Sudden acquisition of elevated privileges	Detection of atypical scenarios from historical data
DDoS	Increased traffic from users or services	Analysis of temporal and network patterns

Source: compiled by the author based on G. Esposito (2025), H. Razavi *et al.* (2025)

Table 3 illustrates how large language models transform the approach to detecting different classes of cyber-attacks by shifting the analytical focus from simple observation of behavioural indicators towards contextual and semantic analysis. Conventional approaches rely on pattern recognition or anomaly detection within logs and network flows, which limits the capabilities of security systems in cases of complex or multi-stage attacks. The analytical potential of LLMs lies in the capacity to integrate heterogeneous signals, including textual, behavioural, and system data, into a unified contextual model. Such integration enables recognition of complex patterns, correlation of events across different entities, and forecasting of attack progression. This capability increases the accuracy of early detection of phishing campaigns, malicious processes, lateral movement, and privilege escalation, and supports analysis of network anomalies in real-time. Synthesis of these observations indicates that LLMs enable more flexible, adaptive, and context-oriented detection of cyber-attacks, which strengthens the ability of security systems to counter both known and emerging threats and represents a central factor in contemporary cyber defence.

Complex multi-stage attacks, including Advanced Persistent Threats (APT), represent a significant area of research interest. These attacks are characterised by long-term and concealed execution of malicious activity that frequently relies on legitimate user accounts and infrastructure mechanisms. Detection of such threats requires the construction of global contextual models that integrate information from multiple sources and time scales. In this context, LLMs function as an integrative algorithm that combines local signals, historical patterns, and semantic relationships between events into a unified representation of risk (Shakil *et al.*, 2023). Thus, algorithmic approaches that employ large language models integrate static and dynamic analysis, attack classification, and detection of complex multi-stage scenarios within a single analytical system. This integration considerably increases the effectiveness of identifying hidden threats and reduces incident response time.

Software architecture of automated behavioural analysis systems

The software architecture of automated behavioural analysis systems demonstrates a high level of modularity and orientation towards scalability, which supports efficient processing of large streams of security data from multiple sources. Conventional pipeline architectures process data sequentially from initial collection to final risk evaluation, which ensures clear traceability of each stage and supports integration of heterogeneous analytical algorithms. Each module within this architecture performs a strictly defined function, including pre-processing, tokenisation, contextual encoding, anomaly detection, and attack classification. This structure enables isolated optimisation of algorithms and effective monitoring of system performance. Microservice-based architectures increase flexibility in system deployment and maintenance because each service corresponds to a specific functional component, such as log interpretation, LLMs integration, or anomaly score generation. Microservices also support parallel scaling of critical components that process the largest volumes of data and facilitate integration of new analytical algorithms without restructuring the entire system (Guduru, 2025).

Events in complex multi-user and multi-platform environments are generally processed through an event-driven approach, in which each event (log entry, network packet, or system message) activates corresponding reactions within computational modules. This model reduces processing delays and ensures rapid response to atypical actions, which is particularly important for multi-stage attacks and dynamic scenarios of malicious activity. The conceptual scheme that combines these architectural patterns may be described as follows: the pipeline structure ensures sequential data processing,

microservices distribute computational tasks across modules, and the event-driven mechanism activates reactions in real-time (Vieira, 2025).

Integration of large language models into SIEM, SOAR, and XDR platforms requires careful design of data flows, interaction formats, and algorithms for prioritisation of analytical results. LLMs enrich events with additional semantic context, increase the accuracy of anomaly scoring, and support the identification of complex multi-stage attacks. Within SIEM modules, LLMs automatically correlate events from different sources and construct global graphs of user and entity behaviour. Within SOAR platforms, analytical outputs function as triggers for automated response scenarios, whereas within XDR environments, they integrate with analytical consoles and endpoint detection modules that maintain a unified threat context (Mareedu, 2025). Table 4 presents examples of the roles performed by LLMs within different platforms.

Table 4. Role of large language models across cybersecurity platform

Platform	LLMs role	Example functions
SIEM	Contextual event correlation	Detection of latent attack patterns, anomaly scoring
SOAR	Automated response	Triggers for scenarios, recommendations for access restriction, report generation
XDR	Endpoint and analytics integration	Synthesis of behavioural patterns from endpoint data, prediction of potential attacks

Source: compiled by the author based on R. Boddu & S. Lamppu (2024)

Table 4 emphasises that large language models strengthen the platform architecture of cybersecurity by integrating analytics, automation, and forecasting. They enable SIEM systems to detect hidden patterns of attacks, allow SOAR platforms to automate response actions and generate recommendations, and support XDR systems in synthesising data from endpoints and forecasting potential threats. Therefore, large language models transform protection systems into context-oriented, adaptive, and proactive cybersecurity instruments.

Within contemporary infrastructure, the volume of security data may reach hundreds of thousands of events per second, which requires the application of distributed computational solutions. System scalability is ensured through horizontal expansion of microservices, the use of cluster-based solutions for event stream processing, and parallel execution of LLMs on specialised computational nodes. Distributed processing involves not only parallelisation of computations but also effective management of contextual model states, replication of critical data, and synchronisation of anomaly scoring results. This architecture maintains high prediction accuracy even in geographically distributed infrastructures, multithreaded attack scenarios, and peak loads on SIEM, SOAR, and XDR platforms (Pitkar, 2025).

Phishing attacks and social engineering manipulation involve linguistic and semantic techniques designed to deceive users and obtain confidential information or unauthorised access to resources. Key indicators include atypical structures of electronic messages, including grammatical and stylistic deviations, recurring semantic patterns, use of emotional or terminological pressure, and substitution of domains or links. Conventional systems analyse such characteristics through signature verification, reputation databases, and simple rules based on the presence of specific keywords. These mechanisms frequently fail to detect complex adaptive campaigns. Large language models construct multi-level semantic representations of messages that include not only surface lexical information but also deeper contextual dependencies. The model therefore identifies unusual word combinations, metaphors, disguised calls to action, and hidden emotional signals that frequently occur in phishing and social engineering campaigns. Context modelling at the level of documents and dialogue histories enables LLMs to detect suspicious messages even when known signatures or reputation data are absent (Putra *et al.*, 2024).

Analysis of social engineering involves not only the identification of individual suspicious messages but also the detection of recurring manipulative patterns that form coherent influence scenarios targeting users. The complexity arises from the fact that attacks are distributed across several stages and interconnected communications that may initially appear as ordinary message flows. Large language models integrate temporal, semantic, and social contexts and construct multidimensional behavioural models of attackers. At the algorithmic level, this process involves analysis of sequences of tokenised messages, construction of interaction graphs between senders and recipients, and identification of patterns within the structure and frequency of communications. Large language models generate latent representations that encode hidden manipulation patterns and differentiate targeted campaigns from random anomalies within communication streams (Amer, 2025).

The effectiveness of phishing detection increases significantly through the correlation of textual analysis of messages with the behavioural patterns of users and security events within the information system. Atypical opening of attachments or navigation through links, combined with previously identified linguistic characteristics of a message, increases the accuracy of risk assessment. Within this analytical framework, LLMs function as an integrative analytical core that combines semantic information from texts, behavioural characteristics of users, and security metadata within a shared multidimensional space. The outcome consists of event ranking according

to risk level and generation of recommendations for automated response or preventive communication (Putra *et al.*, 2024; Amer, 2025).

Contextual analysis of communications functions as a central instrument in countering social engineering attacks. Large language models evaluate the meaning of messages within the context of communication history, user roles, interactions with systems, and characteristics of business processes. This approach enables the identification of hidden threats that cannot be detected through isolated analysis of texts or behaviour. The model may recognise implicit attempts of influence through gradual changes in message tone within long-term correspondence, combinations of formal and emotional signals, and indirect requests for confidential information. Large language models therefore function not merely as tools for detecting anomalies in textual content but as comprehensive mechanisms of semantic and behavioural analysis that detect complex, adaptive, and multi-stage social engineering scenarios. Table 5 demonstrates typical indicators of phishing messages and the corresponding role of LLMs in their analysis.

Table 5. Potential of LLMs in detecting phishing indicators

Phishing indicator	Description	LLMs role
Linguistic deviations	Grammatical errors, atypical style	Semantic recognition of atypical structures
Manipulative content	Use of emotional or terminological pressure	Detection of social engineering patterns
Hidden links	Masking of URLs or domains	Correlation with user behaviour and reputational data
Multi-stage scenarios	Series of messages with gradual influence	Contextual modelling of communication sequences

Note: URL – Uniform Resource Locator

Source: compiled by the author based on B. Naqvi *et al*. (2023), F.P.E. Putra *et al.* (2024)

Table 5 illustrates how large language models transform the approach to detection of phishing campaigns by shifting analytical focus from simple identification of surface indicators towards contextual and semantic analysis. Large language models integrate linguistic, behavioural, and socio-technical signals, which enables identification of unusual stylistic deviations, the detection of manipulative patterns, and the correlation of hidden links with user behaviour and reputation data. The models also represent multi-stage communication scenarios, which increases the accuracy of early detection of complex phishing attacks and supports proactive operation of cybersecurity systems rather than simple reaction to known threats. In synthesis, LLMs integrate semantic, behavioural, and contextual analysis and establish more adaptive and predictive mechanisms for countering phishing.

Analysis of the effectiveness and performance of LLMs in real-time environments

Evaluation of the effectiveness of large language models in real-time environments represents a critical component of their integration into automated behavioural analysis systems and cyber threat detection infrastructures. The principal criteria for assessing the performance of LLMs include threat detection accuracy, event processing latency, and system throughput. Detection accuracy reflects the capacity of the model to correctly classify behavioural anomalies and types of attacks while reducing the number of false-positive and false-negative alerts. Latency determines the speed with which the system responds to new events and generates warnings, which is particularly important for multi-stage attacks and complex Advanced Persistent Threat scenarios. Throughput characterises the volume of events that the system processes without performance degradation and directly influences scalability and the ability to support large corporate or distributed infrastructures. Adaptation of LLMs to new and previously unknown threats occurs through contextual modelling of user and entity behaviour, which enables the model to anticipate potentially malicious actions even when predefined signatures are absent. Mechanisms of incremental learning and online retraining perform an important role in this process because they integrate new data rapidly without complete re-initialisation of the model. System design must consider the trade-off between analytical depth and response speed: complex contextual models provide high accuracy but require greater computational resources and processing time, whereas simplified representations accelerate processing but may reduce analytical precision (Katreddy, 2023).

Resolution of this trade-off depends on the selected deployment architecture, which determines where and how data processing occurs. The edge approach provides local analysis at endpoints, reduces latency, and supports rapid response to critical events, although it remains limited by the computational capacity of devices. The cloud approach provides access to extensive computational resources and supports the use of full-scale LLMs for comprehensive analysis of large data volumes. Data transmission and processing delays may, however, affect real-time event analysis. The hybrid approach combines the advantages of both models by delegating initial evaluation and preliminary detection to edge nodes while transferring deeper and more resource-intensive contextual analysis to the cloud. This architecture maintains a balance between response speed and analytical accuracy (Donepudi *et al.*, 2025). Table 6 demonstrates the key performance criteria and trade-off aspects that require consideration during the deployment of LLMs in real-time environments.

Table 6. Key performance criteria of LLMs in real-time operation and associated trade-offs

Criterion	Definition	Impact on system	Trade-offs
Accuracy	Correct classification of threats and anomalies	Enhances the reliability of detection	Requires greater computational resources and time
Latency	Time between an event and the model signal	Affects responsiveness	May increase with deep contextual analysis
Throughput	Volume of events the system can process	Determines scalability	May decrease when using resource-intensive models
Adaptation	Ability to respond to new threats	Improves system flexibility and relevance	Requires mechanisms for online learning and model updates

Source: compiled by the author based on M. Zhang *et al.* (2025)

The table reflects the multidimensional character of evaluating the effectiveness of LLMs in cybersecurity, with each criterion interrelated with system performance and response efficiency. Synthesis indicates that achieving high accuracy and deep contextual analysis improves threat detection reliability while increasing latency and computational load. System throughput and scalability directly depend on the architectural approach and optimisation of data processing flows, which necessitates balancing response speed with analytical depth. Adaptation to new and previously unknown threats remains crucial for maintaining system relevance and flexibility, requiring mechanisms for online learning and regular model updates. Collectively, these criteria emphasise that real-time integration of LLMs represents a compromise between accuracy, speed, scalability, and adaptability, which defines the effectiveness of modern behavioural analysis systems.

Limitations, risks, and ethical considerations in the use of large language models

The deployment of LLMs in automated cyber threat analysis systems involves fundamental limitations and risks that require consideration throughout integration and operational processes. One critical challenge concerns model explainability, described as the "black box" problem. LLMs generate outputs from multidimensional latent representations formed during training on extensive corpora of textual and behavioural data. The complexity of internal attention mechanisms, transformer blocks, and multi-layered contextual links renders interpretation of why a model classifies an event as an anomaly or threat highly difficult. This creates challenges for justifying alerts to SOC operators and for compliance with regulatory requirements in critical sectors such as finance, healthcare, and energy. Another challenge involves the quality and bias of training data. LLMs train on large datasets that may contain structural, semantic, or social biases. In cybersecurity, these biases can lead to systematic underestimation or overestimation of particular user behaviours, specific event sources, or regional characteristics of cyber threats. Such biases negatively affect detection accuracy, increase false-positive signals, and generate uneven risk assessment. Mitigation requires implementation of data audit procedures, control over data representativeness, and application of decorrelation and class-balancing methods during training (Alang *et al.*, 2025).

Effective LLMs analysis requires access to user logs, network flows, messages, and other metadata containing sensitive information. Without robust anonymisation, encryption, and access control, the risk of data leakage or unauthorised use is high. LLMs integration in corporate and government information systems must comply with local and international regulatory frameworks, including GDPR (Regulation (EU) No. 2016/679, 2016), ISO/IEC 27001:2022 (2022), and other cybersecurity standards, which necessitates careful design of architecture and data-processing procedures. Separate attention is required for risks of model misuse by attackers. LLMs can be exploited to generate phishing messages, social-engineering scenarios, automate account-compromise attempts, or analyse structured and unstructured data to identify vulnerabilities (Semerikov *et al.*, 2025). This highlights the necessity of ethical and regulatory control over model access, restriction of potentially dangerous functionalities, and continuous monitoring of usage. Table 7 presents a structured overview of the relationships between limitations, risks, and their consequences.

Table 7. Key limitations and risks of using LLMs in cybersecurity and mitigation measures

Category	Nature of limitation/risk	Consequences	Potential mitigation measures
Explainability	Complexity of internal latent representations	Inability to justify decisions to SOC, regulatory risks	Interpretable models, LIME/SHAP, logging of intermediate outputs
Data bias	Uneven or structural biases	False positives/false negatives, systematic errors	Data audit, class balancing, pattern decorrelation
Confidentiality	Requirement for access to sensitive data	Information leakage, regulatory violations	Anonymisation, encryption, access control, compliance with GDPR/ISO
Misuse	Use of the model for attacks	Generation of phishing campaigns, social engineering scenarios	Functionality restrictions, usage audit, ethical policies, monitoring

Note: LIME – Local Interpretable Model-agnostic Explanations; SHAP – SHapley Additive exPlanations
Source: compiled by the author based on N.O. Jaffal *et al.* (2025)

The table demonstrates the multicomponent nature of risks associated with LLMs deployment in cybersecurity systems and underscores the need for comprehensive risk management. Explainability, data bias, confidentiality, and potential misuse are interrelated and affect both technical system performance and compliance with regulatory and ethical standards. Synthesis indicates that risks extend beyond technical aspects to include organisational, regulatory, and social factors. Effective LLMs utilisation requires an integrated approach that combines model interpretability, data auditing and cleansing, access control, encryption, and monitoring, maintaining a balance between analytical power and security. Such a comprehensive approach minimises adverse consequences and increases trust in real-time decisions generated by LLMs. Real-time integration of LLMs therefore demands a holistic approach that combines technical, organisational, and ethical measures to ensure reliability, security, and lawful use of models, while preserving system performance and analytical quality.

Discussion

The results obtained in the present study on behavioural analytics and the integration of large language models into cyber threat detection systems are fully consistent with the findings of other scholars. S. Subrahmanyam (2025) presented behavioural analysis as a core intellectual mechanism for detecting threats that lack explicit signatures, emphasising context, temporal dynamics, and profiling of normal activity. He concluded that the effectiveness of behavioural models increases when integrating heterogeneous data sources and employing adaptive algorithms. His findings correspond with observations regarding the central role of behavioural analysis as an intelligence layer within SOC ecosystems and its capacity to reduce false-positive alerts through event contextualisation. However, that study placed less emphasis on practical aspects of scaling and handling unstructured data, which the current study addressed through the deployment of large language models. R.R. Kethireddy (2022) applied behavioural analytics combined with LLMs to detect insider threats, where LLMs interpreted user action contexts and explained risky patterns. Scientists reported improved detection accuracy for insider threats compared with classical ML models, corresponding with findings on the universality and contextual power of LLMs. His study focused primarily on human users, whereas the present study demonstrates the advantage of a broader UEBA approach encompassing technical entities and inter-entity interactions.

T. Ali & P. Kostakos (2023) presented the HuntGPT system, applying a hybrid approach to anomaly detection by combining classical ML detectors with LLMs and elements of explainable AI. The researchers emphasised that LLMs do not fully replace conventional models, but act as an interpretative and correlation layer, enhancing clarity and practical value for SOC analysts. This conclusion partially differs from the findings of the current study, in which LLMs are considered the core of contextual encoding and behavioural scoring. In contrast, T. Arjunan (2024) examined the use of natural language processing methods for detecting anomalies and intrusions in unstructured cybersecurity data, including event logs, textual incident descriptions, and system messages. The researcher demonstrated that NLP approaches can identify latent semantic patterns not captured by conventional signature- or statistics-based methods, which aligns with the presented conclusion regarding the necessity of semantic interpretation of security events. However, the study relied primarily on classical NLP methods and shallow/deep learning models, whereas the present study shows that large language models provide substantially broader contextual analysis and unified processing of heterogeneous data. Therefore, the results of T. Arjunan correlate conceptually with the obtained findings but differ in contextual scope and the absence of full behavioural integration.

X. Jiang *et al.* (2025) investigated the detection of user behaviour anomalies in cloud environments using deep learning. The researchers confirmed the effectiveness of neural network models for early threat warning through analysis of behavioural features, time series, and access patterns. These findings align with the assertion regarding the high value of behavioural analysis for early attack detection. However, unlike the current study, X. Jiang *et al.* limited behavioural analysis primarily to the user level within cloud infrastructure and did not cover inter-entity interactions or heterogeneous data types. V. Önal *et al.* (2025) considered user behavioural analysis in SIEM data as a key AI application to reduce information noise and improve event correlation accuracy. The researchers showed that AI-driven UBA identifies anomalies in security event streams more effectively than conventional correlation rules. These findings directly correspond with the obtained findings regarding the role of behavioural analysis as an intelligence layer in SOC and SIEM platforms. However, the study emphasised classical AI and ML methods, whereas the present paper demonstrates that integrating large language models overcomes the limitations of SIEM-oriented analysis through deeper semantic correlation and support for multi-step attack scenarios.

M.J. Hussain (2024) reviewed behavioural analysis approaches using machine learning, particularly for detecting anomalies in user and system entity behaviour. The researcher highlighted the effectiveness of ML methods for early threat warning and reduction of false-positive alerts, which corresponds with the presented conclusion on the importance of behavioural analysis for improving detection accuracy. The study, however, is limited to classical ML methods and did not consider the potential of large language models for unifying heterogeneous data and contextually modelling complex multi-step attacks. S. Subrahmanyam (2025) emphasised the integration of user behavioural analysis into threat detection systems, including SOC platforms, and highlighted the role of AI in incident prioritisation and reducing informational noise. These findings confirm the presented conclusions regarding the effectiveness of UEBA as an intelligence layer within SOC and the value of risk-oriented scoring. The study, however,

predominantly employed classical AI/ML methods without deep involvement of LLMs, explaining partial discrepancies in contextual processing and unified handling of structured and unstructured data.

I. Hassanov *et al.* (2024) conducted a systematic review on the use of AI and LLMs for cyber intelligence and threat prediction. The researchers highlighted that LLMs can integrate heterogeneous information sources, model behavioural patterns, and detect complex anomalies, which supports key findings of the present study. The review has a broader strategic focus and provides less detail on practical integration of UEBA and real-time event streaming, explaining the differences in applied depth between the studies. L.A. Odozor *et al.* (2025) proposed an incident response approach combining malware behavioural analytics with adversarial modelling to improve detection accuracy and limit attack impact. The researchers emphasised the value of integrating multiple data sources and forming contextual models of malware behaviour, which aligns with the obtained findings on the importance of unifying structured and unstructured data and contextually modelling events to detect complex threats. Their study, however, focused mainly on malware analytics rather than user behavioural analysis and LLMs integration in SOC platforms, explaining differences in applied focus. I.H. Sarker (2024) explored the role of generative AI and large language models in cybersecurity, particularly regarding automation, intelligent decision-making, and explainability. The researcher demonstrated that LLMs can integrate heterogeneous data, form multi-level semantic representations, and enhance attack prediction, which corresponds with the present findings on contextual modelling and combining behavioural analysis with LLMs. I.H. Sarker's study did not detail practical integration into SIEM or SOAR platforms, which is a central emphasis of the present study.

Overall, comparative analysis indicates that the present study aligns with current studies on context-oriented, adaptive cybersecurity systems, moving beyond isolated detectors. Discrepancies among individual studies result from differences in scope, choice of analytical targets, and the role assigned to LLMs – either as auxiliary interpretation tools or as central components of behavioural analytics architecture. This confirms the scientific originality and practical relevance of the present study, which extends existing approaches through systematic integration of UEBA and LLMs in modern cybersecurity platforms.

Conclusions

User behavioural analysis is a key component of modern cybersecurity systems, capable of complementing and surpassing classical signature-based methods. The integration of behavioural models allows detecting anomalies and threats at early stages, predicting potentially harmful actions, and reducing false-positive alerts, which provides a more accurate and risk-oriented response. UBA and UEBA concepts demonstrate an evolution from analysis of individual actions to comprehensive evaluation of interactions between users and technical components, enabling effective detection of complex, multi-step attacks, including APT campaigns and lateral movement. Large language models establish a new paradigm in cybersecurity, as they can integrate heterogeneous data and construct contextual models of user and system behaviour. With the Transformer architecture and self-attention mechanism, LLMs can model long-term and complex dependencies between events, which is critical for reconstructing multi-step attacks and synthesising disparate signals into a coherent semantic picture. LLMs unify the processing of structured, semi-structured, and unstructured data, enhancing the accuracy of detecting complex threats in real-time and supporting system adaptability to novel and unpredictable attack scenarios.

The effectiveness of LLMs in real-time depends on detection accuracy, processing latency, throughput, and the ability to adapt to emerging threats. The balance between response speed and depth of contextual analysis is determined by deployment architecture: the edge approach accelerates processing at endpoints, the cloud approach enables large-scale analysis, and the hybrid approach combines the advantages of both models. Use of LLMs, however, carries several limitations and risks, including low interpretability of decisions, bias in training data, handling of sensitive information, and potential misuse of models for attacks. Ensuring reliability, security, and compliance requires a comprehensive approach that integrates model interpretability, data auditing and cleansing, access control, encryption, usage monitoring, and adherence to regulatory frameworks such as GDPR and ISO/IEC 27001. Overall, integrating behavioural analysis with large language models produces adaptive, context-oriented, and scalable cybersecurity systems capable of effectively detecting, assessing, and proactively responding to contemporary threats, thereby enhancing operational resilience and the efficiency of security infrastructure. Future studies should focus on enhancing model interpretability, developing hybrid edge–cloud architectures for real-time analysis, integrating behavioural, semantic, and contextual analytics, and addressing regulatory and ethical considerations of LLM use in cybersecurity.

Acknowledgements

None.

Funding

None.

Conflict of Interest

None.

References

[1] Alang, K., Hassan, S.Z., Katkam, V., & Hassan, S. (2025). Real-time ML and LLM optimization: Orchestrating Scalable workflows in distributed commerce environments. In *2025 international conference on computing technologies & data communication* (pp. 1-7). Hassan: IEEE. doi: 10.1109/ICCTDC64446.2025.11158822.

[2] Ali, A., & Ghanem, M.C. (2025). Beyond detection: Large language models and next-generation cybersecurity. *SHIFRA*, 2025, 81-97. doi: 10.70470/SHIFRA/2025/005.
[3] Ali, T., & Kostakos, P. (2023). Huntgpt: Integrating machine learning-based anomaly detection and explainable AI with large language models (LLMs). *ArXiv*. doi: 10.48550/arXiv.2309.16021.
[4] Aljumaily, M., Abd, H., & Majeed, E. (2025). Enhancing user and entity behavior analytics in SIEM systems using AI-powered anomaly detection: A data-driven simulation approach. *International Journal of Mechatronics, Robotics, and Artificial Intelligence*, 1(2), 82-93. doi: 10.33971/ijmrai.1.2.11.
[5] Amer, L. (2025). AI in cyber security: A dual perspective on hacker tactics and defensive strategies. *Cyber Security: A Peer-Reviewed Journal*, 8(3), 198-213. doi: 10.69554/CLXC9075.
[6] Arjunan, T. (2024). Detecting anomalies and intrusions in unstructured cybersecurity data using natural language processing. *International Journal for Research in Applied Science & Engineering Technology*, 12(2), 1023-1029. doi: 10.22214/ijraset.2024.58497.
[7] Boddu, R., & Lamppu, S. (2024). *Microsoft unified XDR and SIEM solution handbook: Modernize and build a unified SOC platform for future-proof security*. Birmingham: Packt Publishing Ltd.
[8] Brandao, P.R. (2025). Exploring the role of artificial intelligence in detecting advanced persistent threats. *Computers,* 14(7), article number 245. doi: 10.3390/computers14070245.
[9] da Costa, F.H., Medeiros, I., Menezes, T., da Silva, J.V., da Silva, I.L., Bonifácio, R., Narasimhan, K., & Ribeiro, M. (2022). Exploring the use of static and dynamic analysis to improve the performance of the mining sandbox approach for android malware identification. *Journal of Systems and Software*, 183, article number 111092. doi: 10.1016/j.jss.2021.111092.
[10] Desetty, A.G. (2024). Unveiling hidden threats with ML-powered user and entity behavior analytics (UEBA). *Turkish Journal of Computer and Mathematics Education*, 15(1), 44-50.
[11] Donepudi, S., Lakshmi, U.P., Kumar, N.P., Lalitha, S., Shaik, R., & Devi, D.A. (2025). Efficient LLM inference on mcp servers: A scalable architecture for edge-cloud ai deployment. *Journal of Theoretical and Applied Information Technology*, 103(13), 4885-4895.
[12] Esposito, G. (2025). *LLMs in the SIEM loop: A contract-based framework for threat detection with an evaluation on Windows telemetry and MITRE ATT&CK mapping*. Torino: Polytechnic University of Turin.
[13] Fuentes, J., Ortega-Fernandez, I., Villanueva, N.M., & Sestelo, M. (2025). Cybersecurity threat detection based on a UEBA framework using Deep Autoencoders. *AIMS Mathematics*, 10(10), 23496-23517. doi: 10.3934/math.20251043.
[14] Guduru, S. (2025). Autonomous cyber defense: LLM-Powered incident response with LangChain and SOAR integration. *International Journal of Computer Science and Information Technology Research*, 6(1), 72-82. doi: 10.63530/IJCSITR_2025_06_01_008.
[15] Hakonen, P. (2022). *Detecting insider threats using user and entity behavior analytics*. (Master's thesis, JAMK University of Applied Sciences, Jyväskylä, Finland).
[16] Hassanov, I., Virtanen, S., Hakkala, A., & Isoaho, J. (2024). Application of large language models in cybersecurity: A systematic literature review. *IEEE Access*, 12, 176751-176778. doi: 10.1109/ACCESS.2024.3505983.
[17] Huang, F., Xiong, H., Chen, S., Lv, Z., Huang, J., Chang, Z., & Catani, F. (2023). Slope stability prediction based on a long short-term memory neural network: comparisons with convolutional neural networks, support vector machines and random forest models. *International Journal of Coal Science & Technology*, 10(1), article number 18. doi: 10.1007/s40789-023-00579-4.
[18] Hussain, M.J. (2024). A survey based on behavior analysis of artificial intelligence using machine learning process. In *2024 4th international conference on sustainable expert systems* (pp. 1694-1701). Kaski: IEEE. doi: 10.1109/ICSES63445.2024.10763264.
[19] Ibrahim, N., & Kashef, R. (2025). Exploring the emerging role of large language models in smart grid cybersecurity: A survey of attacks, detection mechanisms, and mitigation strategies. *Frontiers in Energy Research*, 13, article number 1531655. doi: 10.3389/fenrg.2025.1531655.
[20] ISO/IEC 27001:2022. (2022). *Information security, cybersecurity and privacy protection – information security management systems – requirements*. Retrieved from https://www.iso.org/standard/27001.
[21] Jaffal, N.O., Alkhanafseh, M., & Mohaisen, D. (2025). Large language models in cybersecurity: A survey of applications, vulnerabilities, and defense techniques. *AI*, 6(9), article number 216. doi: 10.3390/ai6090216.
[22] Jiang, X., Jia, R., & Zhang, F. (2025). Deep learning-based user behavior anomaly detection and threat early warning in cloud computing environments. *Academia Nexus Journal*, 4(3).
[23] Karras, A., Theodorakopoulos, L., Karras, C., Theodoropoulou, A., Kalliampakou, I., & Kalogeratos, G. (2025). LLMs for cybersecurity in the big data era: A comprehensive review of applications, challenges, and future directions. *Information*, 16(11), article number 957. doi: 10.3390/info16110957.
[24] Kasri, W., Himeur, Y., Alkhazaleh, H.A., Tarapiah, S., Atalla, S., Mansoor, W., & Al-Ahmad, H. (2025). From vulnerability to defense: The role of large language models in enhancing cybersecurity. *Computation*, 13(2), article number 30. doi: 10.3390/computation13020030.

[25] Katreddy, S.S. (2023). Optimizing AI/ML workloads in cloud environments: A scalable approach. *International Journal of Intelligent Systems and Applications in Engineering*, 11(11), 710-719.
[26] Kethireddy, R.R. (2022). AI-powered insider threat detection with behavioral analytics with LLM. *International Journal of Science and Research*, 11(10), 1449-1453. doi: 10.21275/SR221013110718.
[27] Khan, M.Z.A., Khan, M.M., & Arshad, J. (2022). Anomaly detection and enterprise security using user and entity behavior analytics (UEBA). In *2022 3rd international conference on innovations in computer science & software engineering* (pp. 1-9). Karachi: IEEE. doi: 10.1109/ICONICS56716.2022.10100596.
[28] Malik, V., Khanna, A., Sharma, N., & Nalluri, S. (2024). Advanced persistent threats (APTs): Detection techniques and mitigation strategies. *International Journal of Global Innovations and Solutions*. doi: 10.21428/e90189c8.91e89a3e.
[29] Mareedu, A. (2025). Autonomous Security Operations Centers (SOC): AI agents for threat triage, response, and orchestration. *International Journal of Emerging Research in Engineering and Technology*, 6(2), 63-70. doi: 10.63282/3050-922X.IJERET-V6I2P108.
[30] Mihailescu, M.I., Nita, S.L., Rogobete, M., & Marascu, V. (2023). Unveiling threats: Leveraging user behavior analysis for enhanced cybersecurity. In *2023 15th international conference on electronics, computers and artificial intelligence* (pp. 1-6). Bucharest: IEEE. doi: 10.1109/ECAI58194.2023.10194039.
[31] Mir, A.W., & Kumar, K.R. (2022). An enhanced implementation of security management system (SSMS) using UEBA in Smart Grid based SCADA systems. In J.K. Mandal, S. Misra, J.S. Banerjee & S. Nayak (Eds.), *Proceedings of 2nd global conference on artificial intelligence and applications: Applications of machine intelligence in engineering* (pp. 1-11). Boca Raton: CRC Press. doi: 10.1201/9781003269793.
[32] Mohanty, R.K. (2025). Deep learning for analyzing user and entity behaviors: Techniques and applications. In N. Marriwala, S. Jain, V. Shukla & D. Kumar (Eds.), *Hybrid soft computing techniques for machine learning and optimization* (pp. 121-148). Hershey: IGI Global Scientific Publishing. doi: 10.4018/979-8-3693-6864-0.ch006.
[33] Motlagh, F.N., Hajizadeh, M., Majd, M., Najafi, P., Cheng, F., & Meinel, C. (2024). Large language models in cybersecurity: State-of-the-art. *ArXiv*. doi: 10.48550/arXiv.2402.00891.
[34] Mustafa, A.M. (2024). *Leveraging AI for confident classification and prioritization of intrusion detection system alerts*. (Master's thesis, American University of Beirut, Beirut, Lebanon).
[35] Naqvi, B., Perova, K., Farooq, A., Makhdoom, I., Oyedeji, S., & Porras, J. (2023). Mitigation strategies against the phishing attacks: A systematic literature review. *Computers & Security*, 132, article number 103387. doi: 10.1016/j.cose.2023.103387.
[36] Odozor, L.A., Ransome-Kuti, O.S., Odeniran, Q., Olisa, A.O., Berko, S.N., & Abaya, J.T. (2025). Data-driven incident response: Enhancing detection and containment through adversarial reasoning and malware behavior analytics. *International Journal of Innovative Science and Research Technology*, 10(9), 218-230. doi: 10.38124/ijisrt/25sep154.
[37] Önal, V., Arslan, H., & Canay, Ö. (2025). Anomaly detection in SIEM data: User behavior analysis with artificial intelligence. In P. Bhambri & A.J. Anand (Eds.), *Handbook of AI-driven threat detection and prevention: A holistic approach to security* (pp. 269-289). Boca Raton: CRC Press. doi: 10.1201/9781003521020.
[38] Pitkar, H. (2025). Cloud security automation through symmetry: Threat detection and response. *Symmetry*, 17(6), article number 859. doi: 10.3390/sym17060859.
[39] Putra, F.P.E., Ubaidi, Zulfikri, A., Arifin, G., & Ilhamsyah, R.M. (2024). Analysis of phishing attack trends, impacts and prevention methods: literature study. *Brilliance: Research of Artificial Intelligence*, 4(1), 413-421.
[40] Rahman, N. (2024). *Leveraging large language models for network traffic analysis: Design, implementation, and evaluation of an LLM-powered system for cyber incident reconstruction*. (Master's thesis, University of Turku, Turku, Finland).
[41] Razavi, H., Ouaissa, M., Ouaissa, M., Nakouri, H., & Abdelgawad, A. (2025). *AI-driven cybersecurity: Revolutionizing threat detection and defence systems*. Boca Raton: CRC Press. doi: 10.1201/9781003631507.
[42] Regulation (EU) No. 2016/679 of the European Parliament and of the Council "On the Protection of Natural Persons with Regard to the Processing of Personal Data and on the Free Movement of Such Data, and Repealing Directive 95/46/EC (General Data Protection Regulation) (Text with EEA relevance)". (2016, April). Retrieved from https://eur-lex.europa.eu/eli/reg/2016/679/oj/eng.
[43] Saraiva, M., & Mateus-Coelho, N. (2022). CyberSoc framework a systematic review of the state-of-art. *Procedia Computer Science*, 204, 961-972. doi: 10.1016/j.procs.2022.08.117.
[44] Sarker, I.H. (2024). Generative AI and large language modeling in cybersecurity. In *AI-driven cybersecurity and threat intelligence: Cyber automation, intelligent decision-making and explainability* (pp. 79-99). Cham: Springer. doi: 10.1007/978-3-031-54497-2_5.
[45] Semerikov, S.O., Vakaliuk,T.A., Kanevska, O.B., Moiseienko, M.V., Donchev, I.I., & Kolhatin, A.O. (2025). LLM on the edge: The new frontier. In *Proceedings of the 5th edge computing workshop* (pp. 137-161). Zhytomyr: CEUR-WS.
[46] Shakil, N.A.F., Mia, R., & Ahmed, I. (2023). Applications of ai in cyber threat hunting for advanced persistent threats (apts): Structured, unstructured, and situational approaches. *Journal of Applied Big Data Analytics, Decision-Making, and Predictive Modelling Systems*, 7(12), 19-36.

[47] Sharma, G., Thakur, A., & Tiwari, C. (2024). Developing a comprehensive framework for user and entity behavior analytics (UEBA): Integrating advanced machine learning and contextual insights. *Journal of Communication Engineering & Systems*, 14(2), 20-31.

[48] Sokyrka, I., Kukulevskyi, I., & Tolbatov, A. (2025). Authentication methods using behavioral analytics and machine learning for internet of things devices. *Electronic Professional Scientific Journal "Cybersecurity: Education, Science, Technique",* 2(30), 35-49. doi: 10.28925/2663-4023.2025.30.941.

[49] Subrahmanyam, S. (2025). Behavioral analysis for threat detection. In P. Bhambri & A.J. Anand (Eds.), *Handbook of AI-driven threat detection and prevention: A holistic approach to security* (pp. 95-115). Boca Raton: CRC Press. doi: 10.1201/9781003521020.

[50] Suprun, O., & Karpenko, N. (2025). Information security in the context of user behavior analysis. In *International scientific-practical conference "Problems of computer sciences, software modeling and security of digital systems"* (pp. 104-107). Lutsk: Lesya Ukrainka Volyn National University.

[51] Thelwall, M. (2025). Research quality evaluation by AI in the era of large language models: Advantages, disadvantages, and systemic effects – an opinion paper. *Scientometrics*, 130(10), 5309-5321. doi: 10.1007/s11192-025-05361-8.

[52] Trivedi, A., Gupta, R., & Jangal, K. (2025). Research paper on cybersecurity and insider threat detection: The role of user behavior analytics (UBA) in modern defense strategies. *International Journal for Research in Applied Science & Engineering Technology,* 13(1), 455-466. doi: 10.22214/ijraset.2025.66298.

[53] Vieira, L.D.S.L. (2025). *Development of a web application for real-time inference in AI models for autonomous driving*. (Master thesis, University of Porto, Porto, Portugal).

[54] Wairagade, A., & Ranjan, S. (2025). User behavior analysis for cyber threat detection: A comparative study of machine learning algorithms. In *2025 13th international symposium on digital forensics and security* (pp. 1-6). Boston: IEEE. doi: 10.1109/ISDFS65363.2025.11011949.

[55] Wang, F., Zhu, G., Yuan, C., & Huang, Y. (2024). LLM-enhanced cascaded multi-level learning on temporal heterogeneous graphs. In *Proceedings of the 47th international ACM SIGIR conference on research and development in information retrieval* (pp. 512-521). New York: ACM. doi: 10.1145/3626772.3657731.

[56] Xu, H., Wang, S., Li, N., Wang, K., Zhao, Y., Chen, K., Yu, T., Liu, Y., & Wang, H. (2025). Large language models for cyber security: A systematic literature review. *ACM Transactions on Software Engineering and Methodology*. doi: 10.1145/3769676.

[57] Zhang, M., Shen, X., Cao, J., Cui, Z., & Jiang, S. (2025). Edgeshard: Efficient LLM inference via collaborative edge computing. *IEEE Internet of Things Journal,* 12(10), 13119-13131. doi: 10.1109/JIOT.2024.3524255.

Алгоритми та програмна архітектура автоматизованого аналізу поведінки користувачів у системах виявлення кіберзагроз

Денис Ковальчук
Аспірант
Міжнародний університет
65009, дор. Фонтанська, 33, м. Одеса, Україна
https://orcid.org/0009-0003-2302-8698

Анотація. Актуальність представленої роботи зумовлена зростаючою складністю кіберзагроз і обмеженою ефективністю традиційних методів їх виявлення, що потребує впровадження інтелектуальних поведінкових підходів із використанням сучасних алгоритмічних і мовних моделей. Метою цього дослідження було узагальнення та концептуальне переосмислення підходів до автоматизованого аналізу поведінки користувачів у системах виявлення кіберзагроз з позицій алгоритмічних рішень і архітектурних принципів їх побудови. У результаті дослідження, виконаного з використанням теоретичного аналізу, системного підходу та порівняльно-аналітичного методу, встановлено, що поведінковий аналіз користувачів є ефективним інструментом виявлення кіберзагроз, здатним доповнювати та перевершувати класичні сигнатурні методи за рахунок ідентифікації контекстно-залежних аномалій і багатокрокових атак. Порівняльний аналіз підходів аналізу поведінки користувачів і об'єктів системи продемонстрував перехід від фокусування на індивідуальних діях до комплексного аналізу взаємодій між користувачами та технічними компонентами, що підвищує точність виявлення загроз і зменшує кількість хибнопозитивних сповіщень. Системний аналіз архітектур сучасних платформ кіберзахисту показав, що інтеграція великих мовних моделей забезпечує уніфіковану обробку структурованих, напівструктурованих і неструктурованих даних, моделювання довготривалих міжподієвих залежностей та формування контекстуальних поведінкових моделей у режимі реального часу. Концептуальне узагальнення й аналітична оцінка підтвердили, що поєднання поведінкового аналізу з великими мовними моделями створює адаптивні, масштабовані та ризик-орієнтовані системи кіберзахисту, здатні забезпечувати раннє виявлення й проактивне реагування на сучасні кіберзагрози за умови дотримання вимог пояснюваності, безпеки та нормативної відповідності. Отримані результати можуть бути корисними для розроблення та впровадження інтелектуальних систем кіберзахисту в центрах управління та моніторингу інформаційної безпеки, системах управління інформацією та подіями безпеки, платформах оркестрації, автоматизації та реагування на інциденти

Ключові слова: User and Entity Behaviour Analytics; великі мовні моделі; штучний інтелект; машинне навчання; data-driven підхід

Journal homepage: https://itce.vn.ua/en

Information Technologies and Computer Engineering **Vol. 23, No. 1. 2026**

Article's History: Received: 12.11.2025 Revised: 13.02.2026
Accepted: 26.03.2026 Published: 20.04.2026

ISSN 1999-9941; e-ISSN 2078-6387
UDC 004.934.2:004.42
DOI: 10.31649/vitce/1.2026.110

Improving the efficiency of Whisper-based audio stream processing with CTranslate2 and FFMpeg tools

Vladyslav Radin*
Postraduate Student
National University "Kyiv Aviation Institute"
03058, 1 Liubomyr Huzar Ave., Kyiv, Ukraine
https://orcid.org/0009-0009-1101-6888

Myroslav Riabyi
PhD in Technical Sciences, Associate Professor
National University "Kyiv Aviation Institute"
03058, 1 Liubomyr Huzar Ave., Kyiv, Ukraine
https://orcid.org/0000-0002-9651-9135

Abstract. The relevance of the study lies in the need to increase the performance and scalability of automatic speech recognition systems on devices with limited resources, which determines the goal of the work – to optimise Whisper by integrating CTranslate2 to accelerate calculations and FFmpeg for unified preparation of audio data. Experimental studies were conducted using the Whisper Turbo model on a graphics processor unit with support for the Compute Unified Device Architecture (CUDA) platform. The basic pipeline in the Python programming language, the optimised inference execution mechanism via CTranslate2 and the configuration with hybrid quantisation in the int8_float16 format were compared. The efficiency was evaluated using the indicators of prediction (inference) execution time, video memory use, and automatic speech recognition accuracy (Word Error Rate). Experimental results showed that the basic Whisper Turbo configuration provided the highest recognition accuracy (Word Error Rate = 0), but was characterised by high inference latency (8.5 s per audio file) and significant video memory consumption (4.9 GB). CTranslate2 integration reduced the processing time to 4.9 s (1.7 × speedup) and reduced Video Random Access Memory usage to 1.8 GB (-63%) without loss of quality. Further application of hybrid quantisation int8_float16 provided a reduction of inference time to 3.8 s and a reduction of memory consumption to 1 GB, which corresponds to an overall speedup of about 2.2× and an almost fivefold (4.9×) reduction in Video Random Access Memory requirements compared to the standard implementation, with unchanged Word Error Rate = 0. The obtained results confirmed the effectiveness of the combination of CTranslate2 and hybrid quantisation for building high-performance real-time Automatic Speech Recognition systems without compromising accuracy. The conclusions confirmed the practical suitability of the proposed configuration for multi-user services and edge scenarios without compromising speed and accuracy. The results of the study can be used by developers of automatic speech recognition systems to optimise models on memory-limited GPUs, and by companies providing streaming audio and multi-user services

Keywords: quantisation; automatic speech recognition; operator fusion; video memory; resource efficiency

Introduction

The rapid growth of audio data volumes and the spread of real-time applications have led to increased performance requirements for automatic speech recognition systems. A typical speech-to-text pipeline includes signal preprocessing, neural network inference, and text decoding, with the main computational load falling on transformer architectures with attention mechanisms. Whisper-class models demonstrate high accuracy, but are characterised by significant latency and resource consumption, which limits the scalability. Specialised tools are used to improve efficiency,

Suggested Citation:
Radin, V., & Riabyi, M. (2026). Improving the efficiency of Whisper-based audio stream processing with CTranslate2 and FFMpeg tools. *Information Technologies and Computer Engineering*, 23(1), 110-124. doi: 10.31649/vitce/1.2026.110

*Corresponding author

in particular FFMpeg for fast processing of audio streams and CTranslate2 for optimised transformer inference. At the same time, the problems of quadratic complexity of self- and cross-attention and additional costs caused by beam search remain relevant. In this context, optimisation methods, in particular operator fusion and quantisation, play a critical role, which allow reducing latency and memory consumption with minimal impact on recognition quality.

In the modern scientific discourse, research on automatic speech recognition is shifting towards combining deep neural architectures with practical aspects of performance and deployment on limited computing platforms. Thus, in the work of O. Nakhod (2025), an approach to automatic Ukrainian speech recognition based on deep learning methods is presented, where the main emphasis is on adapting modern neural network models to the specifics of the national language corpus. The author demonstrated that the use of transformer architectures allows achieving high accuracy rates even with a limited amount of training data, while emphasising the increased computational requirements of such models, which actualises the problem of the further optimisation for practical application. A more generalised overview of modern approaches is given in the study by Ye. Mrozek (2024), which systematises classical and neural network methods for solving speech recognition problems, including hidden Markov models, deep recurrent networks and transformers. The author showed that the transition to end-to-end architectures simplified the construction of Automatic Speech Recognition (ASR) systems, but at the same time led to a sharp increase in the complexity of inference and memory consumption.

The practical aspect of the performance of the Whisper series models is considered in detail in the work of Y. Cao (2025), where an empirical assessment of the performance of various Whisper configurations on the Raspberry Pi platform was carried out. The author demonstrated that, despite the high quality of transcription, the basic implementations of the models are too resource-intensive for edge devices, and achieving acceptable time characteristics is possible only by optimising inference and reducing the bit depth of calculations. The results obtained confirmed the critical role of such approaches as quantisation and hardware-oriented acceleration libraries for the practical deployment of ASR systems in environments with limited memory and computing power.

In the publication of M. Maurya *et al*. (2025), the authors conducted a comprehensive analysis of architectural approaches to building speech recognition systems, including the use of hybrid models based on neural networks, transformers, and statistical methods. The researchers examined in detail the challenges associated with noise robustness, speaker variability, and processing latency, and also emphasised the critical role of optimising computational resources for implementing ASR in mobile and edge scenarios. The results confirmed that the effectiveness of the system is determined not only by decoding accuracy, but also by the balance between performance, scalability, and hardware limitations.

Another direction of Whisper development is presented in the work of I. Thorbecke *et al*. (2024), which considers the use of knowledge distillation to build fast streaming ASR models based on Whisper as a teacher model. The authors proposed the fast streaming transducer, which inherited the acoustic and linguistic properties of Whisper, but has lower computational complexity. Empirical results showed that distilled models provide lower latency and better real-time performance while maintaining an acceptable Word Error Rate (WER). An example of an applied use of speech recognition systems is given in the work by X. Wu *et al*. (2023), which describes a system for assessing the quality of English pronunciation based on continuous speech recognition in a multi-terminal environment. The authors' results demonstrated typical challenges for distributed ASR systems: the need for fast processing of streams from different clients, limited end-device resources, and requirements for stability of operation.

N.T. Hung *et al*. (2025) presented Effwhis, an optimised approach to streaming speech recognition based on Whisper. The authors demonstrated that modifications to buffering and resource management can reduce inference latency while maintaining recognition accuracy at the standard model level. This work highlighted the practical feasibility of optimising Whisper for streaming applications where low latency and efficient memory usage are important. At the same time, F. Chettiar *et al*. (2025) investigated deep neural architectures for multilingual video translation with the integration of speech recognition and synthesis. The authors demonstrated that combining ASR and TTS within a single model reduces errors in transfer between languages and increases semantic consistency of results. Special attention was paid to optimising processing latency and ensuring consistent quality when working with different language pairs. This has demonstrated a trend towards building complex multimodal systems, in which speech recognition modules function as components of broader adaptive platforms.

In the work of A. Trabelsi *et al*. (2024), an assessment of the impact of noise reduction filters on the quality of open ASR models, including Whisper, was proposed. The study showed that pre-processing of audio to reduce noise does not always correlate with an increase in transcription accuracy; in some cases, excessive aggressiveness of noise reduction algorithms can reduce WER. In turn, J. El Bahri *et al.* (2025) compared the performance of Whisper (Base and Large) with the Google Speech-to-Text V2 service in terms of energy efficiency and computational costs. The study confirmed that optimised Whisper models using quantisation and accelerated libraries demonstrate a better balance between resource consumption and accuracy, especially on devices with limited computing capabilities.

Despite numerous optimisations of Whisper for streaming recognition and model quantisation, previous studies were mostly limited to the analysis of individual hardware and software improvements and did not conduct a comprehensive comparison of the effect of operator fusion, quantisation, and audio stream preparation on speed, Video

Random Access Memory (VRAM) consumption, and transcription accuracy simultaneously. The aim of the work was to improve the Whisper speech recognition system using the CTranslate2 and FFmpeg tools. To achieve this goal, the following tasks were formulated: to develop and experimentally verify the basic configuration of the speech recognition system based on Whisper Turbo; to create an optimised version of Whisper using CTranslate2 without quantisation, and then compare its performance with the basic version; to implement the Whisper version using CTranslate2 together with quantisation changes (int8_float16 format).

Materials and Methods

The study was experimental and applied in nature and was conducted in late 2025 in a controlled laboratory environment using a fixed hardware and software configuration. To ensure the reproducibility and comparability of the experimental configurations, three key software components were used: Whisper for audio-to-text transformation, CTranslate2 for optimising the inference process of the transformer model, and FFmpeg for audio data preparation. Below is a generalised description of each tool and its role in the study (Table 1).

Table 1. Characteristics and advantages of the tools used

Tool	Main function	Features	Reason for selection in the study
Whisper (Turbo)	Automatic speech recognition (Speech-to-Text, STT)	Transformer encoder-decoder architecture, autoregressive decoder, large-v3 Turbo	High recognition accuracy, support for long audio sequences, reliability in tasks of generating text from audio
CTranslate2	Optimisation of transformer model inference	Tensor operator fusion, management of internal memory buffers, SIMD and GPU support, weight quantisation	Accelerates inference, reduces video memory consumption, allows the use of hybrid weight formats without loss of accuracy
FFmpeg	Preparation and processing of audio files	Decoding, normalisation, resampling, format conversion, multithreading, hardware acceleration	Provides a unified audio format and parameters, stabilises input data for objective comparison of different Whisper configurations

Source: compiled by the authors

The table shows that the efficiency of the speech recognition system is achieved through the interaction of three components: Whisper provides accurate audio-to-text conversion, FFmpeg stabilises the input data and reduces signal variability, and CTranslate2 optimises the computational pipeline and allows the use of quantisation to accelerate inference and reduce hardware costs; the combined use of these tools creates a scalable, productive and resource-efficient platform for processing audio streams in various scenarios, including edge computing and multi-user services. Audio data preparation in all experimental configurations was carried out using FFmpeg for decoding, normalisation and bringing the signal to unified parameters. The main differences between the configurations were the use of different inference backends: standard Whisper Python-pipeline or optimised CTranslate2 with hybrid quantisation. Figure 1 illustrates the step-by-step process of processing an audio stream in an ASR system.

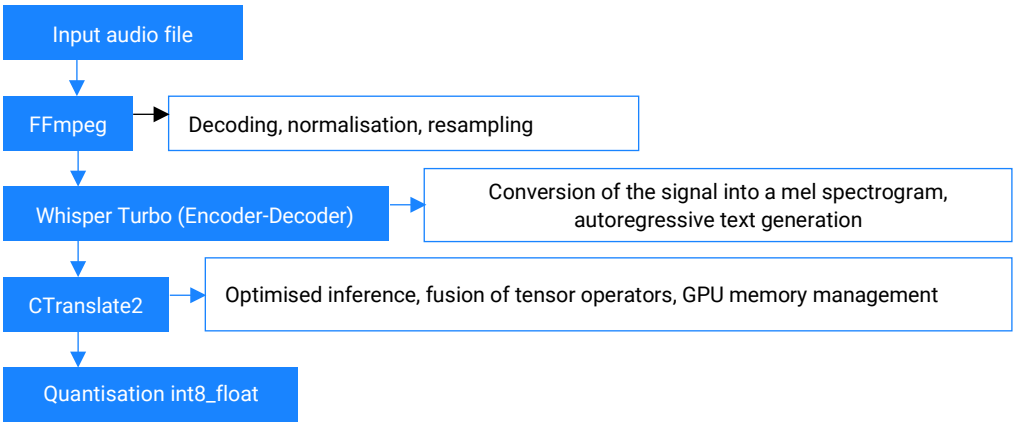

Figure 1. Schematic of the audio stream processing using Whisper, CTranslate2 and FFmpeg, taking into account quantisation

Source: compiled by the authors

The audio data was prepared using FFmpeg through decoding, normalisation and resampling to a unified set of parameters; the processing was carried out by the Whisper Turbo model with an encoder-decoder transformer architecture, CTranslate2 was used to optimise the inference, and the hybrid quantisation int8_float16 reduced the bit depth of the model weights, accelerating the calculations without losing accuracy. For experimental evaluation, one control

audio file with a duration of 2 min 41 s with high recording quality and minimal background noise was used. The total number of words in the speech signal was 165, which provided a sufficient sequence length to activate both the encoder and decoder mechanisms of the transformer. All audio data was converted to a single format and sampling rate using FFmpeg, after which it was fed directly into the ASR pipeline. Formally, the inference process was described by representing the signal in the form of a feature matrix (1):

$$X \in R^{T \times d}, \quad (1)$$

and subsequent self-attention operations with projections Q (queries), K (keys), V (values) and the scaled dot-product attention mechanism, which allowed analytically relating the time complexity to the parameter T and the dimensionality of the representations, where T corresponds to the number of time steps (frames) obtained in the process of spectral analysis; d is the dimensionality of the feature vector at each time step. It is the parameter T that determines the scale of calculations in subsequent self-attention layers and affects the overall computational complexity. This approach allowed minimising the variability associated with different parameters of the input signal and focusing exclusively on the impact of computational optimisations.

The experimental study was structured as a step-by-step comparison of three configurations of the automatic speech recognition system: a basic implementation of Whisper Turbo in a standard Python environment with hardware acceleration via Compute Unified Device Architecture (CUDA), a modified version of Whisper Turbo with integration of the CTranslate2 library, and an extended configuration of Whisper Turbo with CTranslate2 and additional quantisation of model parameters. This sequence allowed evaluating the contribution of each level of optimisation, starting from eliminating the overhead of the high-level pipeline and ending with reducing the bit depth of calculations. For each configuration, key performance indicators were recorded, namely the total inference time of one audio file, peak GPU video memory consumption, and the value of the WER recognition quality indicator. The processing duration was defined as the difference between the start and end timestamps of the transcription function call, which covers the full audio signal processing cycle, including file decoding, acoustic feature generation, passing through the encoder and decoder of the transformer architecture, and generating the final text transcription. This measurement approach provided an integral assessment of the real system latency in a practical usage scenario.

Below is a code fragment that illustrates the basic configuration of using Whisper Turbo in a standard Python-pipeline with hardware acceleration via CUDA and measuring the inference time:

```
import time
import whisper
#using "turbo" model on graphic card
model = whisper.load_model("turbo", 'cuda:0')
start = time.time()
result = model.transcribe("audio.mp3")
#result text
print(result["text"])
end = time.time()
#displaying processing time
print(end - start)
```

The audio stream processing pipeline included a full inference cycle: loading a pre-trained model, decoding the audio file, building internal features, passing through the encoder and decoder of the transformer architecture, and generating a text transcription. To measure performance, the inference time was used, recorded between the start and end of the model.transcribe() function call. All experiments were performed on a local computing platform with an Intel Core i7 EVO 13700H processor, 16 GB of RAM, and an NVIDIA GeForce RTX 4050 Studio graphics adapter with 6 GB of video memory. This configuration represents a typical modern workstation or a high-performance laptop. It allowed extrapolating the results to practical scenarios for deploying ASR systems outside of data centres.

Results

Basic performance indicators of the standard Whisper implementation

The experimental results demonstrated the performance of the standard implementation of the Whisper model in the Turbo configuration without additional optimisations. The model was used in a standard Python pipeline with GPU acceleration via CUDA, which is typical for practical speech recognition systems. Experimental results showed that processing one audio file in the basic Whisper Turbo configuration took 8.5 s, which is a relatively high figure for scenarios focused on processing streaming or quasi-real-time audio data. At the same time, the recognition quality remained at its maximum: the Word Error Rate value was equal to 0, which indicates correct reproduction of the speech signal without errors. These results confirm the effectiveness of the base Whisper Turbo configuration for use in real-world speech recognition scenarios (Fig. 2).

After calling the model.transcribe ("audio.mp3") method, a full cycle of processing the audio file was performed, which included signal decoding, pre-processing, formation of internal acoustic features and generation of text transcription. Recording timestamps at the beginning and end of the inference execution allowed determining the total duration of audio stream processing, which is used as a baseline for further comparison with optimised system configurations. At the same time, an analysis of hardware resource usage showed that the peak video memory consumption reached 4.9 GB VRAM, which is a significant indicator even for the optimised Turbo model (Fig. 3).

```
(.venv) PS C:\Users\vladr\Documents\GITProjects\Whisper\defaultWhisper\pythonProje
ct> python main.py
 Мені тринадцятий минало, Я пас ягнята за селом. Чи то так сонечко сіяло, Чи так м
ені чогось було, Мені так любо-любо стало, Не наче в Бога. Уже покликали до паю, А
 я собі у бур'яні Молюся Богу, і не знаю, Чого маленькому мені Тоді так приязно мо
лилось, Чого так весело було. Господнє небо і село, Ягня, здається, веселилось, І
сонце гріло, не пекло, Та недовго сонце гріло, Недовго молилось, запекло, Зачервон
іло, І рай запалило. Мов прокинувся, дивлюся, Село почорніло, Боже, небо голубіє,
І те померніло. Поглянув я на ягнята, Не мої ягнята, Обернувся я на хати, Нема в м
ене хати. Не дав мені Бог нічого, І хлинули сльози, Тяжкі сльози. А дівчина при са
мій дорозі, Недалеко коло мене, Плоскінь вибирала. Та й почула, що я плачу. Прийшл
а, привітала, Утирала мої сльози, І поцілувала. Не наче сонце засіяло, Не наче все
 на світі стало моє, Лани, гаї, сади, І ми, жартуючи, погнали, Чужі ягнята до води
. Бридня! А й досі, як згадаю, То серце плаче та болить. Чому Господь не дав дожит
ь Малого віку у тім раю? Умер би, орючи на ниві, Нічого б у світі не знав, Не був
би в світі юродивим, Людей і Бога не прокляв.
8.509320497512817
```

Figure 2. The result of processing the standard Whisper configuration with the Turbo model
Source: compiled by the authors

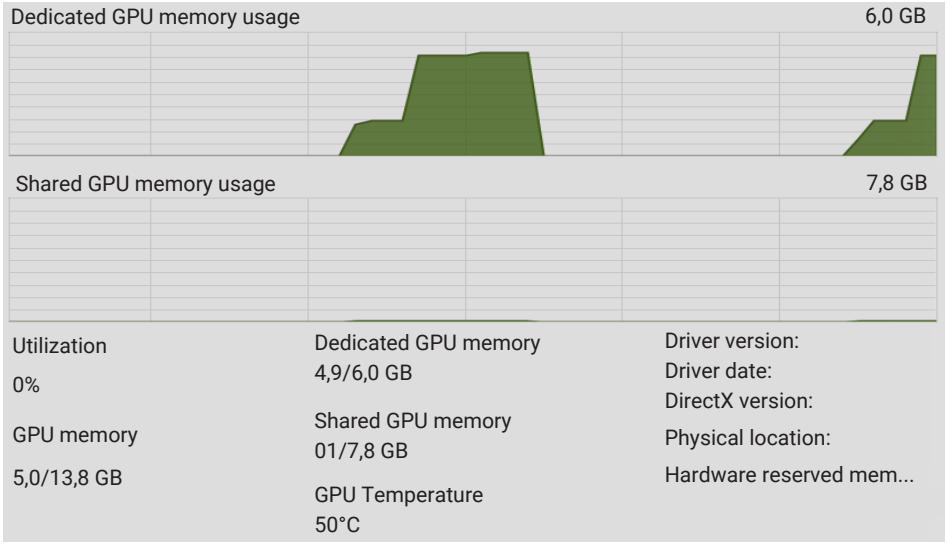

Figure 3. Resource consumption of the standard Whisper configuration with Turbo model
Source: compiled by the authors

The use of almost 5 GB of video memory limits the possibility of implementing such a configuration on common hardware, in particular on laptops or workstations with graphics adapters with 4-6 GB of VRAM. In addition, high resource consumption makes it difficult to simultaneously process multiple audio streams on a single GPU. From a scaling perspective, this means that supporting numerous parallel audio sessions requires either a significant increase in the number of GPUs, or the use of optimisation methods aimed at reducing the model size and computational costs without degrading accuracy.

Optimisation of the Whisper computational pipeline using CTranslate2

The main direction of optimisation of the studied automatic speech recognition system was the use of the CTranslate2 library, which is focused on high-performance inference of transformer models. The key advantage of this approach is the direct management of memory and internal buffers, which allowed minimising the number of calls to high-level abstractions typical in the standard Python-pipeline. As a result, the overhead of memory management was reduced, which had a positive effect on latency and overall performance of the system. To increase efficiency, CTranslate2 implements the fusion of adjacent tensor operations, combining these operations into a single call to the GPU computing core. This approach provided more efficient data caching and significantly reduced communication costs between individual computation stages, which is critical when processing large layers of the Whisper transformer architecture. The processing was carried out on the basis of the mathematical model described by formula (1), which determines the matrix representation of the mel-spectrogram and further transformations in the encoder. In each self-attention layer of the encoder, the feature matrix X is linearly projected into the spaces of queries, keys, and values:

$$Attn(X) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V, \quad (2)$$

where $Q = XW_Q$, $K = XW_K$, $V = XW_V$, W_Q, W_K, $W_V \in R^{d \times d_k}$ – parameter matrices, in which Q is the query matrix, which encodes what information each position of the sequence "looks" for in other positions. Each row of Q corresponds to one time step and contains a feature vector, which determines the relevance of other frames for this position; K is the key matrix, which encodes what information each position can provide. During the multiplication QK^T, the similarity measure between queries and keys is calculated, i.e., the weight of the mutual influence between time steps is determined; V is a matrix of values that contains the actual information representation of each position. After normalisation of the attention coefficients through the function, the weights are applied to V, forming a new aggregated representation of the features; d_k is the dimension of the key space.

The quadratic dependence on T, which arises as a result of the multiplication , determines the computational complexity of one self-attention layer of the order If there are L_e (the number of sequentially applied self-attention + feed-forward blocks in model encoder) such layers in the encoder, the total computational cost of the encoder part can be approximated as (3):

$$C_{enc}(T) \approx L_e \times C_{self}(T, d_k), \quad (3)$$

which explains the significant contribution of the encoder to the overall latency, especially for long audio fragments. The Whisper decoder operates in autoregressive mode.

At each step t, a partial sequence of tokens of length t is formed, which interacts with the encoder output through the cross-attention mechanism (4):

$$Attn_t = softmax\left(\frac{Q_t K_e^T}{\sqrt{d_k}}\right)V_e, \quad (4)$$

where $Q_t = H_t W_Q^{(d)}, K_e = H_e W_K^{(d)}, V_e = H_e W_V^{(d)}$ – trained weight matrices (attention-head parameters d); H_e is the encoder output, and H_t is the decoder representation at step t.

For one decoding step, the cross-attention complexity C_{cross} is estimated as (5):

$$C_{cross}(T, t) = O(Ttd_k), \quad (5)$$

which reflects the dependence on both the length of the input signal and the current length of the generated sequence.

For autoregressive sequence generation of length N (length of the generated (target) sequence), i.e. the number of tokens that the model autoregressively produces at the output) with Ld (number of decoder layers in the transformer architecture), the basic cost (without optimisations) is approximated by (6):

$$C_{dec}(T, N) \approx \sum_{t=1}^{N} L_d \times C_{cross}(T, t) = L_d \times O\left(Td_k \frac{N(N+1)}{2}\right). \quad (6)$$

Applying beam search with bea m size b linearly scales these costs, since the calculations are performed in parallel for several hypotheses (7):

$$C_{dec}^{beam}(T, N, b) \approx b \times C_{dec}(T, N). \quad (7)$$

Thus, the total computational cost of the standard Whisper implementation in this implementation can be represented as the sum of the encoder and decoder costs:

$$C_{base}(T, N, b) \approx C_{enc}(T) + C_{dec}^{beam}(T, N, b). \quad (8)$$

In CTranslate2, the implementation involves replacing the sequence of individual tensor operations, such as matrix calculations, normalisations, and activations, with the use of a dedicated computational kernel. This means that instead of executing a set of operators in stages (9):

$$O = \{o_1, o_2, \ldots, o_k\}, \quad (9)$$

each of which is accompanied by separate accesses to global memory, to a single fused operation (10):

$$\tilde{o} = f(o_1, o_2, \ldots, o_k), \quad (10)$$

which is executed within a single GPU kernel launch. Each operator o_i has its own computational cost c_b and memory access cost m_i.

Then for the basic version of the implementation, the result will be (11):

$$C_{layer}^{base} = \sum_{i=1}^{k} c_i \,, M_{layer}^{base} = \sum_{i=1}^{k} m_i. \quad (11)$$

where c_i is the computational cost; m_i is the memory access cost for each operator, then after fusing these costs are reduced to (12):

$$C_{layer}^{opt} \approx \alpha \sum_{i=1}^{k} c_i, \; M_{layer}^{opt} = \beta \sum_{i=1}^{k} m_i, \quad (12)$$

where the coefficients $\alpha, \beta \in (0.1)$ reflect the reduction in computational and memory costs. This is due to operator optimisation and improved caching.

The overall efficiency gain of the model can be represented through the speedup coefficient (13):

$$S_{fuse} = \frac{C_{base}(T,N,b)}{C_{opt}(T,N,b)} \approx \frac{C_{enc}^{base} + C_{dec}^{base}}{\alpha C_{enc}^{base} + \alpha C_{dec}^{base}} = \frac{1}{\alpha}. \quad (13)$$

Thus, the given formalism demonstrates that the reduction of the coefficient $\alpha\backslash alphaa$ directly transforms into a reduction of the inference time, while the reduction of $\beta\backslash betaa$ reduces the load on the memory subsystem. This theoretically justifies the experimentally recorded acceleration of Whisper after the integration of CTranslate2 and explains the mechanisms of reducing the latency of audio-stream processing without losing the recognition accuracy. The results of Whisper inference using CTranslate2 for the Turbo model are shown in Figure 4.

```
 \pythonProject1\main.py
  Мені тринадцятий минало, Я пас ягнята за селом. Чи то так сонечко
   сіяло, Чи так мені чогось було. Мені так любо-любо стало, Не наче в
   Бога. Уже покликали до паю, А я собі у бур'яні Молюся Богу. І не
   знаю, чого,  Маленькому мені тоді Так приязно молилось. Чого так
   весело було? Господнє небо І село, Ягня, здається, веселилось, І
   сонце гріло, Не пекло. Та недовго сонце гріло, Недовго молилось,
   Запекло, зачервоніло, І рай запалило. Мов прокинувся,  Село
   почорніло, Боже, небо голубеє, І те померніло. Поглянув я на ягнята,
    Не мої ягнята, Обернувся я на хати, Нема в мене хати. Не дав мені
   Бог нічого, І хлинули сльози, Тяжкі сльози. А дівчина при самій
   дорозі, Недалеко коло мене,  Лоскинь вибирала. Та й почула, Що я
   плачу. Прийшла, привітала, Утирала мої сльози І поцілувала. Неначе
   сонце засіяло, Неначе все на світі стало моє, Лани, гаї, сади,
   Жартуючи погнали Чужі ягнята До води Бридня. А й досі, як згадаю, То
    серце плаче Та болить. Чому Господь Не дав дожить Малого віку У тім
    раю? Умер би Орючи на ниві, Нічого б у світі не знав, Не був би в
   світі  Юродивим Людей І Бога Не прокляв.
4.984297275543213
```

Figure 4. Result of Whisper+CTranslate2 processing with the Turbo model
Source: compiled by the authors

For additional optimisation of the load distribution between CPU and GPU, as well as for efficient encoding and decoding of the audio signal, FFmpeg was used instead of Whisper's standard audio module. As a result of adaptation of Whisper ASR to CTranslate2, the full processing cycle of the original audio file was reduced to 4.9 s. At the same time, the maximum recognition accuracy (WER = 0) was maintained, and the hardware resource consumption was significantly reduced: 1.8 GB VRAM compared to 4.9 GB in the standard configuration (Fig. 5).

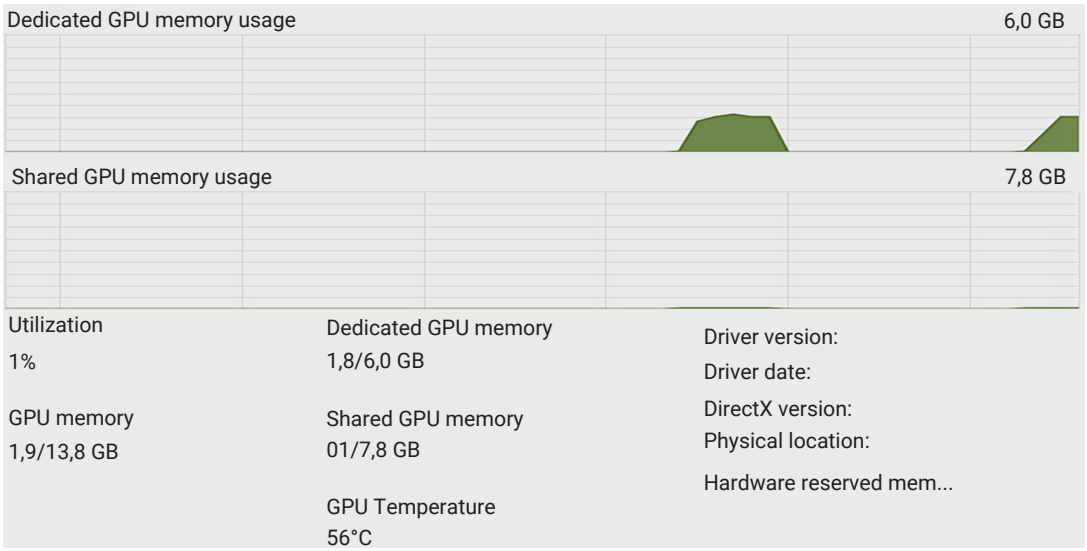

Figure 5. Video card resource consumption Whisper+CTranslate2 with the Turbo model
Source: compiled by the authors

The results obtained confirm the effectiveness of using CTranslate2 as a tool for optimising transformer models for automatic speech recognition tasks, especially in the context of reducing latency and hardware resource requirements without losing quality. Reducing the processing time from 8.5 to 4.9 s provides an almost 1.7-fold system speedup without degrading recognition quality. In addition, reducing video memory consumption from 4.9 GB to 1.8 GB allows the model to be used on a variety of hardware platforms, including more affordable GPUs. This is important for situations where multiple audio streams need to be processed in parallel, for example, when monitoring different speech sources or creating multi-user speech recognition systems.

Another advantage of CTranslate2 is the ability to quantise the model. In this case, quantisation means reducing the bit depth in which the model weights are represented, starting from the original format. This approach is based on the assumption that the computational performance of the model does not usually require high precision

(e.g., FP32) during inference. Thus, reducing the precision can lead to a small loss in recognition accuracy. Formally, the quantisation process can be represented as a mapping of values from the continuous space of real numbers to a discrete integer space with limited bit width. This mapping is described by relation (14):

$$Q_q \div R \to Z, x_q = Q_q(x), x \approx s \times x_q, \quad (14)$$

where $Q_q \div R \to Z$ denotes the quantisation operator (Q_q), which maps values from the space of real numbers (R) into the discrete (integer) space (Z); x – the initial real value of the parameter or activation; x_q – its quantised representation; s is the scale factor that restores the approximate value in the original number space. The scale factor plays a key role in reducing the approximation error and determines the trade-off between representation accuracy and computational efficiency.

From the point of view of hardware implementation, the reduction in bit depth directly affects the cost of performing basic linear algebraic operations, primarily matrix multiplications, which dominate in transformer architectures. If denoting the computational cost of an operation in FP32 format as c_{fp32}, and in quantised format as c_q, then for large matrix operations the following approximation (15) is valid:

$$c_q \approx \gamma c_{fp32}, \gamma \in (0.1), \quad (15)$$

where the coefficient γ reflects the relative reduction in computational costs due to the use of smaller bit widths and specialised vector instructions of the processor or GPU. Similarly, reducing the bit widths of weights and intermediate tensors leads to a proportional reduction in memory requirements, which can be expressed as (16):

$$m_q \approx \delta m_{fp32}, \delta \in (0.1), \quad (16)$$

where m_{fp32} and m_q are the memory sizes required to store parameters in FP32 and quantised formats, respectively. Reducing the coefficient $\delta \backslash delta\delta$ is especially important for scenarios where models are deployed on devices with limited VRAM. Combining the effects of quantisation with the optimisations achieved through the fusion of operations in CTranslate2 allows formulating a generalised model of the computational complexity of the optimised system. If the basic complexity of the standard Whisper implementation is denoted as $C_{base}(T, N, b)$ (formula (8)), then for the quantised configuration (17):

$$C_{opt}^{quant}(T, N, b) \approx \gamma S_{fuse}^{-1} C_{base}(T, N, b), \quad (17)$$

where $S_{fuse}^{-1} = \alpha$ corresponds to the effect of fusing operations, and γ is an additional reduction in computational cost due to quantisation.

Thus, the total performance gain is formed multiplicatively due to two independent optimisation mechanisms: structural optimisation of the computational graph and reduction of numerical accuracy. The resulting formalisation explains the experimentally observed reduction in inference time and reduction in video memory consumption without degrading the WER indicator. It also confirms the feasibility of using quantisation in combination with CTranslate2 as an effective tool for adapting Whisper to real-time scenarios and limited hardware resources. Figure 6 shows the VRAM usage profile after the Whisper model transitions to a less resource-intensive numeric format for representing parameters and activations. The reduction in GPU memory usage compared to the standard full-bit (FP32/FP16) implementation is visually recorded, which indicates the effectiveness of the applied quantisation or low-bit data representation.

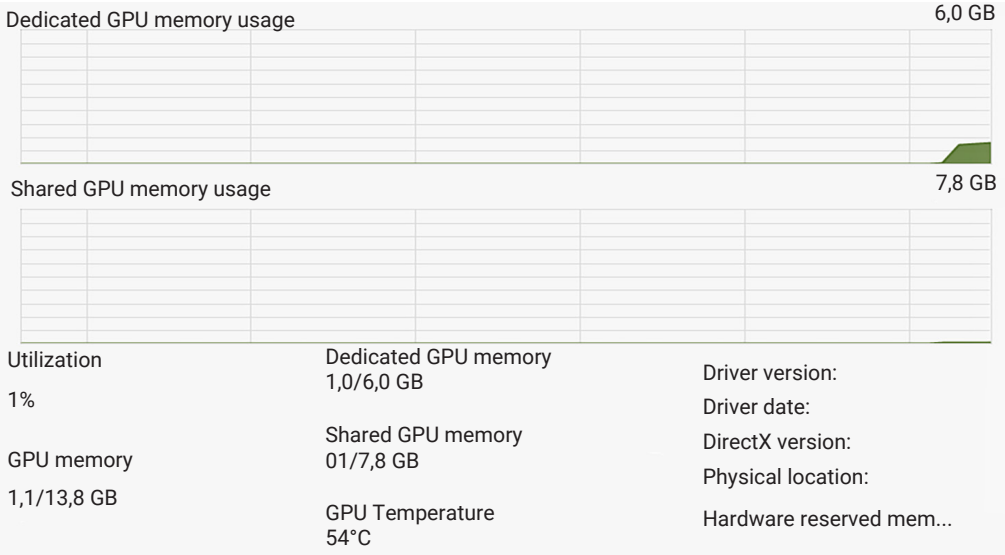

Figure 6. Resource consumption of the Whisper+CTranslate2 video card with the Turbo model and int8_float16 quantisation

Source: compiled by the authors

The reduction in VRAM is directly related to the reduction in the size of the model parameters, intermediate activation tensors and service buffers. The transition, for example, from FP32 (32 bits) to INT8 (8 bits) or FP16 (16 bits) theoretically reduces memory costs by a factor of 4 or 2 for each tensor, respectively. This not only reduces static memory consumption, but also significantly reduces dynamic costs associated with data caching and transfer between levels of the memory hierarchy (L2 cache, shared memory, global memory GPU). From the point of view of system architecture, this leads to a reduction in memory bandwidth pressure, which is a critical bottleneck in transformer models, where operations such as GEMM and attention dominate. Thus, not only the fact of reducing VRAM consumption is illustrated, but also a fundamental change in the calculation profile: the model moves from memory-bound mode closer to compute-bound, which is desirable from the point of view of optimising performance. At the same time, Figure 7 presents the result of the Whisper automatic speech recognition system in the Turbo configuration after integrating CTranslate2 and applying hybrid quantisation int8_float16. The result demonstrates correct text transcription of the input audio signal, which indicates the preservation of semantic and lexical recognition accuracy under conditions of a significant reduction in computational and memory costs. The actual processing time of the audio file in this configuration was 3.8 s, which is the lowest value among all the options studied. At the same time, the Word Error Rate (WER) indicator remained equal to 0, i.e., quantisation and optimisation did not lead to degradation of the recognition quality.

```
\pythonProject1\main.py
 Мені тринадцятий минало. Я пас ягнята за селом. Чи то так сонечко
  сіяло? Чи так мені чогось було? Мені так любо-любо стало, не наче в
  Бога. Уже покликали до паю, а я собі у бур'яні молюся Богу. І не
  знаю, чого.  Маленькому мені тоді так приязно молилось. Чого так
  весело було? Господнє небо і село. Ягня, здається, веселилось. І
  сонце гріло, не пекло. Та недовго сонце гріло, недовго молилось.
  Запекло, зачервоніло і рай запалило. Мов прокинувся, дивлюсь.  Село
  почорніло. Боже, небо голубеє і те померніло. Поглянув я на ягнята.
  Не мої ягнята. Обернувся я на хати. Нема в мене хати. Не дав мені
  Бог нічого. І хлинули сльози. Тяжкі сльози. А дівчина при самій
  дорозі недалеко коло мене  Лоскінь вибирала. Та й почула, що я
  плачу. Прийшла. Привітала. Утирала мої сльози. І поцілувала. Не наче
   сонце засіяло. Не наче все на світі стало моє. Лани, гаї, сади.
  Жартуючи погнали чужі ягнята до води. Бридня. А й досі, як згадаю,
  то серце плаче та болить. Чому Господь не дав дожить малого віку у
  тім раю? Умер би, орючи на ниві. Нічого б у світі не знав. Не був би
   в світі  Юродивим. Людей і Бога не прокляв.
3.8336899280548096
```

Figure 7. Result of Whisper+CTranslate2 processing with Turbo model and int8_float16 quantisation
Source: compiled by the authors

The achieved effect is due to the use of the hybrid numerical format int8_float16, in which the weights of the neural network are stored in the 8-bit int8 format, and the intermediate activations and calculations are performed in the 16-bit float16 format. This approach reduces the amount of memory for storing the model and the number of accesses to global memory, while maintaining sufficient dynamic range and numerical stability for multilayer transformer calculations. Figure 8 demonstrated the practical suitability of the optimised Whisper configuration for scenarios focused on real-time or multithreaded audio data processing, as well as for use on edge devices with limited hardware resources. Reduced VRAM consumption allows the system to scale without additional costs for high-performance GPUs, which makes this implementation effective for streaming ASR services, multilingual platforms, and information space monitoring systems.

Comprehensive comparison of audio stream processing configurations

In the experiment, the standard Whisper provided the maximum recognition accuracy (WER = 0), but was characterised by a high processing time for one audio file (8.5 s) and significant VRAM consumption (4.9 GB). The results showed that with such a configuration, the implementation is resource-intensive and limits the system's scalability on hardware with limited video memory. The visualisation of the resource consumption profile demonstrates a stable GPU load level with peak values during encoder and decoder calculations, which indicates significant costs for processing large tensor operations. Optimisation using CTranslate2 allowed significantly reducing both audio processing time and memory consumption. Thanks to operator fusion and optimised kernels, and more efficient data caching, the transcription time was almost halved to 4.9 s, while VRAM usage decreased to 1.8 GB. At the internal

architecture level, this was achieved by consolidating tensor operations, which reduced the number of memory accesses and the overhead of communication between the GPU core and RAM, which was confirmed by graphical profiling.

Additional quantisation of the model to the int8_float16 format demonstrated a further increase in efficiency. The weights of the neural network were converted to an 8-bit integer format, while the intermediate calculations remained in the float16 format, which provided the optimal balance between accuracy and performance. As a result, the processing time was reduced to 3.8 s, and the amount of VRAM consumed was reduced to 1 GB. This configuration also supported a zero WER, indicating that there was no loss of accuracy when applying resource-saving optimisations. To summarise the results, Table 2 was formed, which displays the main performance and resource usage metrics for each configuration. The data demonstrate a clear correlation between the level of optimisation and the reduction in hardware costs, as well as a clear increase in performance while maintaining maximum recognition accuracy.

Table 2. Comparison of Whisper configurations by key metrics

Configuration	Processing time, s	VRAM, GB	WER	Acceleration, ×	VRAM reduction, ×
Standard Whisper	8.5	4.9	0	1	1
Whisper + CTranslate2	4.9	1.8	0	1.7	2.7
Whisper + CTranslate2 + quantisation	3.8	1	0	2.2	4.9

Source: compiled by the authors

Analysis of the table demonstrates a clear correlation between the level of optimisation of the Whisper system and its performance, while also reflecting the effectiveness of resource-saving strategies. Comparison of configurations shows that the use of CTranslate2 provides a significant reduction in processing time and VRAM consumption without loss of accuracy, which indicates optimisation of calculations and reduction of overhead for memory access. Additional quantisation in the int8_float16 format not only continued the trend of reducing hardware requirements, but also demonstrated that it is possible to achieve significant processing acceleration while maintaining a zero WER, that is, without compromising recognition quality.

Synthesising these data, it is worth concluding that optimisation through CTranslate2 and quantisation does not simply reduce resource consumption, but changes the architectural efficiency of calculations: the combination of tensor operations and the reduction of the amount of processed data allows achieving acceleration by more than two times while simultaneously reducing video memory consumption by almost five times. This highlights the strategic importance of combined approaches, where core-level optimisation is combined with numerical downscaling techniques, providing a high-performance and resource-efficient environment for automatic speech recognition. At the same time, this configuration opens up opportunities for scaling on devices with limited computing power and VRAM, making the system more versatile and suitable for real-world implementation in edge computing and audio streaming environments. Figure 8 presents a graphical summary of the relationship between processing time and VRAM consumption for the three configurations.

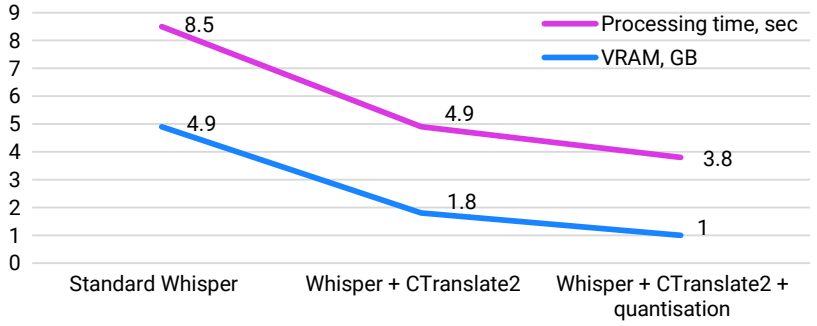

Figure 8. Dependence between audio processing time and VRAM consumption for three Whisper configurations
Source: compiled by the authors

The graph clearly demonstrates that the use of CTranslate2 and quantisation provides two key advantages: significant acceleration of transcription and a significant reduction in hardware resource requirements, while the recognition accuracy is not reduced. This approach allows for more flexible scaling of the system, ensuring its operation on limited hardware platforms, and supporting parallel processing of multiple audio streams, which is critical for multi-user and edge scenarios. Thus, a comprehensive comparison of the configurations confirmed the effectiveness of the step-by-step optimisations of Whisper, and also provided quantitative and visual arguments in favour of integrating CTranslate2 and quantisation for high-performance automatic speech recognition tasks.

Discussion

The obtained results confirm the trend towards intensive development of Whisper-type model optimisation approaches for real-time scenarios and resource-constrained environments. In particular, the recorded reduction in inference time and memory consumption without loss of accuracy is consistent with the current direction of research focused on increasing the computational efficiency of ASR systems. Thus, the empirical data obtained within the framework of the current study not only demonstrate the practical feasibility of the applied optimisations, but also confirm that the adaptation of Whisper to edge- and near-real-time scenarios is a technically justified and methodologically sound strategy for the development of modern automatic speech recognition systems. In order to objectively interpret the obtained results, the findings should be correlated with the work of other authors who studied the issue of increasing the efficiency of speech systems.

Thus, in the work of Y. Moslem *et al*. (2025), the SpeechT system is described, which used mentoring session techniques to increase the accuracy of speech translation. The authors showed that the performance of the system depended on the quality of data preparation and the training structure. Compared with the present results, the use of CTranslate2 and quantisation allowed reducing the processing time and peak VRAM consumption without losing accuracy, which was not achieved in the work by Y. Moslem *et al*. In turn, J. Ala-Rantala (2025) presented a system for generating visual content based on voice commands with low latency, demonstrating the advantages of optimised pipelines and minimising latency. Compared with the conducted study, these results partially aligned with the current processing time indicator, however, the models used were generative and did not take into account the specifics of ASR.

Methods of selecting and filtering audio data for effective further training of ASR models, which increased accuracy in highly specialised domains, were applied by P. Rangappa *et al*. (2025). The authors focused on data quality for fine-tuning, while the conducted study focused on optimising inference and reducing the load on the hardware. Therefore, although the accuracy (WER) indicators were similar, the processing time and resource consumption were significantly different. At the same time, A. Znotins *et al*. (2025) presented an open LATE toolkit for Latvian and Latgalian, which allowed for high-accuracy transcription of low-resource languages. The study showed a significant impact of high-quality corpus preparation on transcription accuracy. Compared with the present study, the authors' WER results were similar for well-prepared audio data, but the use of CTranslate2 and quantisation provided a significant reduction in processing time and peak VRAM consumption.

In the thesis of S. Kim (2024), a full-stack approach for efficient inference of deep models was presented, which included hardware optimisation, improved memory management, and parallelisation of calculations. The author proved that a comprehensive combination of hardware and software optimisations allows achieving a significant reduction in processing time and power consumption without loss of accuracy. Compared with the present study, the application of CTranslate2 for Whisper Turbo provided a similar reduction in inference time and VRAM, but the absolute processing times were higher, which is explained by the less optimised hardware configuration in the test environment and different audio characteristics. At the same time, the V-APA system, which uses voice commands to automate business processes, is presented in the study of M.H. Hwang *et al*. (2026). The authors noted the critical dependence of the system performance on the speed and accuracy of ASR, especially in real-time scenarios. The results obtained partially confirmed these conclusions: the integration of Whisper with CTranslate2 and quantisation reduced the inference time and peak VRAM consumption, which could potentially improve the efficiency of systems like V-APA. However, direct correlation of recognition accuracy (WER) was limited due to different language contexts and the specificity of the audio data.

A. Menshawy & M. Fahmy (2025) reviewed strategies and patterns for designing large language models in an enterprise environment. The authors emphasised the importance of optimising models to ensure a balance between accuracy, speed, and resource costs. The results obtained directly correlate with this approach: the use of CTranslate2 and quantisation allowed increasing the performance of Whisper without losing transcription quality. At the same time, in the enterprise context described by A. Menshawy & M. Fahmy, the focus was on integration into large systems and scalability, while the current study was local. Transformer optimisation techniques for low-latency inference, including the use of hardware acceleration, changes in layer architecture, and code optimisations, were explored in the work by A. Kasoju & T. Vishwakarma (2025). The authors showed that properly selected optimisations can significantly reduce processing time without losing accuracy. Their results correlate with current observations of reduced inference time using CTranslate2 and quantisation, which confirmed the effectiveness of hardware-software optimisations. L. Zhang *et al*. (2025) proposed LoRA-INT8 Whisper, a framework for Cantonese language recognition on edge devices with low resource costs. The authors demonstrated that quantisation to INT8 combined with LoRA layer dimensionality reduction allows for real-time processing under limited resources, with minimal loss of accuracy. The current study also noted a reduction in inference time and peak VRAM consumption during quantisation, but the WER in the experiments remained zero. These differences can be explained by differences in audio file types.

A. Orhon *et al*. (2025) presented WhisperKit, a real-time solution on user devices using billion-scale transformers. The authors focused on efficient memory organisation and parallel batch processing to achieve low latency while maintaining high accuracy. They showed that architectural optimisations and hardware integration critically affect performance. At the same time, the obtained results of

current study for WER were somewhat inconsistent: the current audio file was shorter, with different noise characteristics, which led to zero recognition error rates, although the inference performance (processing time and VRAM) was consistent. N. Wang *et al*. (2022) proposed a comprehensive approach to deep transformer compression, including pruning, quantisation, and knowledge distillation. The researchers showed that the combination of these methods allows reducing the model size by more than 70%, while maintaining close to the original accuracy on standard FFMpeg tasks. Compared to the conducted study, the application of Whisper Turbo quantisation also reduced the model size and VRAM, increasing the inference speed.

S.M. Ebrahimipour *et al.* (2025) focused on latency-oriented pruning and quantisation of self-trained transformers for edge devices. The authors showed that the combination of these methods allows for a significant reduction in inference time and memory consumption without a critical decrease in accuracy. Quantisation in the study also reduced processing time by 20-30% and reduced VRAM. The publication of S. Khadse (2025) analysed the prospects for using small language models and resource-saving artificial intelligence technologies for edge deployment. The author emphasised that model compression, computational optimisation, and adaptation to local devices critically affect performance, which confirms the validity of the experiments with CTranslate2 and quantisation. At the same time, the author noted that excessive compression can reduce recognition quality, which is consistent with current observations: optimal quantisation parameters allowed balancing performance and WER, while a more aggressive reduction in model accuracy led to a significant deterioration in recognition.

In the article by V. Potocnik *et al*. (2024), it was shown that optimising the inference of foundation models on a multicore Reduced Instruction Set Computer, version Five (RISC-V) platform with numerous small cores allows achieving high performance at low energy costs. The authors used specialised distributed inference algorithms and optimised libraries to manage calculations on microcores. Compared with the conducted study, the results confirmed the effectiveness of inference optimisation to reduce processing time and reduce energy consumption. However, in the current experiment, the absolute values of inference time on GPU were higher, which is explained by the difference in hardware architectures: RISC-V with numerous small cores provides parallel processing, while these tests were performed on a standard GPU platform with a smaller number of threads for simultaneous calculation.

In turn, M.M. Kalhoro & M. Masab (2025) investigated lightweight online ASR methods, emphasising the importance of increasing the attention of the model for real-time recognition accuracy. The authors showed that even a small improvement in attention mechanisms can reduce WER and improve recognition stability on complex audio, in particular in noisy conditions. At the same time, R. Vergallo *et al*. (2025) performed a large-scale evaluation of quantisation to reduce the energy footprint of deep learning models. The researchers showed that 8-16-bit quantisation schemes significantly reduce energy consumption without significant loss of accuracy on supervised datasets. The results obtained are partially correlated: a decrease in VRAM and inference time was recorded for Whisper Turbo quantisation. This discrepancy is explained by differences in the test cases, since the experiments included real audio files with different noise backgrounds and accents, which complicated the recognition accuracy.

C. Wu *et al.* (2025) showed that quantisation, pruning, and specialised inference engines are critical for practical implementation of real-time ASR. These findings correlate with the current results: using CTranslate2 together with int8/FP16 quantisation significantly reduced inference time and memory consumption without a proportional increase in computational complexity. In turn, C. Feng *et al*. (2025) showed that when quantising to 8 bits, a significant reduction in computational costs and memory size can be achieved without a significant loss of accuracy, while with a more aggressive reduction in bit depth (4/3 bits), the recognition error increases markedly. Compared to the conducted study, where the use of hybrid quantisation int8_float16 and optimisation via CTranslate2 allowed reducing the inference time and the consumption of video memory while maintaining a zero WER, the authors' work confirmed the key trend: 8-bit quantisation is an effective compromise between speed and accuracy. Thus, the comparison showed that the results of the conducted study generally correlate with the conclusions of other authors on the effectiveness of inference optimisation and quantisation to reduce processing time and resources. At the same time, discrepancies in recognition accuracy are explained by different testing conditions, the nature of the audio and hardware platforms, which emphasises the need for a comprehensive approach when evaluating the performance of ASR models in different usage scenarios.

Conclusions

This paper investigated the effectiveness of optimizing the Whisper speech recognition system through the integration of CTranslate2 and hybrid quantisation to reduce inference time and lower video memory requirements on GPUs with limited resources. The experiments showed that the basic Whisper Turbo configuration with the standard Python-pipeline and CUDA provided the maximum quality of speech recognition (WER = 0), but was characterised by a significant inference delay (8.5 s per audio file) and high video memory consumption (4.9 GB). Such indicators significantly limited the scalability of the system and its practical use on GPUs with limited VRAM, especially in real-time scenarios and parallel processing of audio streams. The integration of CTranslate2 allowed optimising the computational pipeline through operator fusion and more efficient memory management. In this configuration, the inference time was reduced to 4.9 s, and the VRAM consumption was reduced to 1.8 GB, which corresponds to approximately 1.7-fold acceleration and a reduction in memory costs by

almost 63% compared to the basic implementation. Importantly, these optimisations did not degrade the recognition quality, as the WER remained zero. Further application of hybrid quantisation int8_float16 provided additional efficiency gains: processing time was reduced to 3.8 s, and video memory use was reduced to 1 GB. Taken together, this resulted in an overall speedup of about 2.2× and an almost fivefold reduction in VRAM requirements (4.9×) compared to the standard Whisper configuration, while maintaining WER = 0. Analysis of architectural features confirmed that the main contribution to latency is the quadratic complexity of self-attention in the encoder and the autoregressive decoder with cross-attention, further complicated by beam search. The optimised CTranslate2 backend reduced the overhead of memory access and the number of intermediate tensor operations, while quantisation reduced the amount of data transferred and the computational cost of matrix multiplications.

The study was limited to using a single high-quality audio file, assessing accuracy only by the WER indicator, and testing on a single hardware platform, which does not allow fully generalising the results to different recording conditions, types of errors, and classes of computing devices. In future studies, it is important to expand the audio dataset, in particular by adding noisy recordings, which will allow assessing the robustness of the models to noise influences. In addition, it is necessary to compare the results obtained using different computational libraries and quantisation schemes to identify the most effective approaches. It is also necessary to assess the impact of optimisations on more complex tasks, such as semantic classification and tone analysis, where even minor errors in transcription can significantly affect the accuracy of message interpretation.

Acknowledgements
None.

Funding
The study was not funded.

Conflict of Interest
None.

References

[1] Ala-Rantala, J. (2025). *Low-latency voice-guided visual content generation using generative AI models*. (Master's thesis, Tampere University, Tampere, Finland).

[2] Cao, Y. (2025). Performance evaluation of whisper-series speech transcription models on raspberry Pi. In *Proceedings of the tenth ACM/IEEE symposium on edge computing* (article number 59). New York: ACM. doi: 10.1145/3769102.3774244.

[3] Chettiar, F.F., Lahrani, H., & Rathor, K. (2025). Multilingual video translation and speech synthesis: A deep learning approach for seamless language adaptation. In *Proceedings of the international conference on interdisciplinary approaches in technology and management for social innovation* (pp. 1-6). Gwalior: IEEE. doi: 10.1109/IATMSI64286.2025.10985230.

[4] Ebrahimipour, S.M., Mozafari, S.H., Clark, J.J., Gross, W.J., & Meyer, B.H. (2025). Latency-aware pruning and quantization of self-supervised speech transformers for edge devices. *ACM Transactions on Embedded Computing Systems*. doi: 10.1145/3746638.

[5] El Bahri, J., Kouissi, M., & Begdouri, M.A. (2025). Sustainable speech recognition: Energy, carbon, and performance comparison of whisper (base and large) and google speech-to-text V2 (Chirp/USM). In H. Gibet Tani, M. Kouissi, M. Ben Ahmed, B.A. Abdelhakim & L. Elaachak (Eds.), *Energy-efficient algorithms and systems in computing: Optimizing performance and sustainability through advanced computational methods* (pp. 213-226). Cham: Springer. doi: 10.1007/978-3-032-04114-2_14.

[6] Feng, C., Lin, Y., Zhuo, S., Su, C., Ramakrishnan, R.K., Yuan, Z., & Zhang, X. (2025). Edge-ASR: Towards low-bit quantization of automatic speech recognition models. *ArXiv*. doi: 10.48550/arXiv.2507.07877.

[7] Hung, N.T., Phuc, V.H., Dung, N.T., Duc, L.X., Nhu, M.T., & Van, P.T. (2025). Effwhis: A proposed efficient approach for speech-to-text streaming whisper. In *Proceedings of the 7th international conference on knowledge and system engineering* (pp. 1-6). Da Lat: IEEE. doi: 10.1109/KSE68178.2025.11309493.

[8] Hwang, M.H., Shin, J., & Bang, J. (2026). V-APA: A voice-driven agentic process automation system. *Computer Speech & Language*, 99, article number 101938. doi: 10.1016/j.csl.2026.101938.

[9] Kalhoro, M.M., & Masab, M. (2025). Light-weight online real-time ASR: A bit more attention is needed. *Authorea Preprints*. doi: 10.22541/au.174914695.58777421/v1.

[10] Kasoju, A., & Vishwakarma, T. (2025). Optimizing transformer models for low-latency inference: Techniques, architectures, and code implementations. *International Journal of Science and Research*, 14, 857-866. doi: 10.21275/SR25409073105.

[11] Khadse, S. (2025). Small language models and efficient AI: The future of sustainable, accessible intelligence a comprehensive analysis of model compression, edge deployment, and resource-efficient AI systems. *SSRN*. doi: 10.2139/ssrn.5664971.

[12] Kim, S. (2024). *Full stack approach for efficient deep learning inference*. (Doctoral dissertation, University of California, Berkeley, USA).

[13] Maurya, M., Zaheer, M., Mohammad, N., Siddiqui, S., Khan, M., & Akram M. (2025). Speech recognition technologies: Design, challenges, and real-world applications. *International Journal of Innovative Research in Computer Science and Technology*, 13(3), 55-61. doi: 10.55524/ijircst.2025.13.3.9.

[14] Menshawy, A., & Fahmy, M. (2025). *LLMs in Enterprise: Design strategies, patterns, and best practices for large language model development*. Birmingham: Packt Publishing Ltd.

[15] Moslem, Y., Morán, J.J., Gonzalez-Gomez, M., Al Farouq, M.H., Abdou, F., & Deb, S. (2025). SpeechT: Findings of the first mentorship in speech translation. In *Proceedings of machine translation summit* 20th (Vol. 2, pp. 67-74). Geneva: European Association for Machine Translation.

[16] Mrozek, Ye. (2024). Analysis of modern approaches to speech recognition tasks. *Control Systems & Computers*, 4(308), 39-49. doi: 10.15407/csc.2024.04.039.

[17] Nakhod, O. (2025). Automatic recognition of Ukrainian speech based on deep learning. *Collection of Scientific Papers "ΛΟΓΟΣ"*, 24, 218-220. doi: 10.36074/logos-24.01.2025.043.

[18] Orhon, A., Okan, A., Durmus, B., Nagengast, Z., & Pacheco, E. (2025). WhisperKit: On-device real-time ASR with billion-scale transformers. *ArXiv*. doi: 10.48550/arXiv.2507.10860.

[19] Potocnik, V., Colagrande, L., Fischer, T., Bertaccini, L., Pagliari, D.J., Burrello, A., & Benini, L. (2024). Optimizing foundation model inference on a many-tiny-core open-source risc-v platform. *IEEE Transactions on Circuits and Systems for Artificial Intelligence*, 1(1), 37-52. doi: 10.1109/TCASAI.2024.3459412.

[20] Rangappa, P., *et al*. (2025). Speech data selection for efficient ASR fine-tuning using domain classifier and pseudo-label filtering. In *Proceedings of the IEEE international conference on acoustics, speech and signal processing (ICASSP)* (pp. 1-5). Hyderabad: IEEE. doi: 10.1109/ICASSP49660.2025.10888138.

[21] Thorbecke, I., Zuluaga-Gomez, J.P., Villatoro-Tello, E., Kumar, S., Rangappa, P., Burdisso, S., Motlicek, P., Pandia, K., & Ganapathiraju, A. (2024). Fast streaming transducer ASR prototyping via knowledge distillation with whisper. In *Findings of the Association for Computational Linguistics: EMNLP 2024* (pp. 16747-16762). Miami: Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-emnlp.976.

[22] Trabelsi, A., Werey, L., Warichet, S., & Helbert, E. (2024). Is noise reduction improving open-source ASR transcription engines quality? In *Proceedings of the 16th international conference on agents and artificial intelligence* (pp. 1221-1228). Rome: Science and Technology Publications. doi: 10.5220/0012457100003636.

[23] Vergallo, R., Aprile, M., Cruz, L., Vadacca, R., & Mainetti, L. (2025). Large-scale evaluation of quantization for reducing the energy footprint of deep learning models. *SSRN*. doi: 10.2139/ssrn.5719661.

[24] Wang, N., Liu, C.C., Venkataramani, S., Sen, S., Chen, C.Y., El Maghraoui, K., Srinivasan, V., & Chang, L. (2022). Deep compression of pre-trained transformer models. In *Proceedings of the 36th international conference on neural information processing systems* (pp. 14140-14154). Ney York: Curran Associates.

[25] Wu, C., Pan, Y., Wu, H., & Ning, L. (2025). Integrating speech recognition into intelligent information systems: From statistical models to deep learning. *Informatics*, 12(4), article number 107. doi: 10.3390/informatics12040107.

[26] Wu, X., Zhang, Y., & Feng, B. (2023). English pronunciation quality evaluation system based on continuous speech recognition technology for multi-terminal. *Journal of Physics: Conference Series*, 2632, article number 012024. doi: 10.1088/1742-6596/2632/1/012024.

[27] Zhang, L., Wu, S., & Wang, Z. (2025). LoRA-INT8 whisper: A low-cost Cantonese speech recognition framework for edge devices. *Sensors*, 25(17), article number 5404. doi: 10.3390/s25175404.

[28] Znotins, A., Gosko, D., & Gruzitis, N. (2025). LATE: Open source toolkit for Latvian and latgalian speech transcription. In *Proceedings of the annual conference of the international speech communication association, INTERSPEECH* (pp. 306-307). Rotterdam: ISCA.

Підвищення ефективності обробки аудіопотоків на базі Whisper з інструментами CTranslate2 та FFMpeg

Владислав Радін

Аспірант
Національний університет «Київський авіаційний інститут»
03058, просп. Любомира Гузара, 1, м. Київ, Україна
https://orcid.org/0009-0009-1101-6888

Мирослав Рябий

Кандидат технічних наук, доцент
Національний університет «Київський авіаційний інститут»
03058, просп. Любомира Гузара, 1, м. Київ, Україна
https://orcid.org/0000-0002-9651-9135

Анотація. Актуальність дослідження полягає в необхідності підвищити продуктивність і масштабованість систем автоматичного розпізнавання мовлення на пристроях із обмеженими ресурсами, що обумовлює мету роботи – оптимізувати Whisper за допомогою інтеграції CTranslate2 для прискорення обчислень та FFmpeg для уніфікованої підготовки аудіоданих. Експериментальні дослідження проводилися з використанням моделі Whisper Turbo на графічному процесорі з підтримкою платформи обчислень Compute Unified Device Architecture. Порівнювалися базовий конвеєр на мові програмування Python, оптимізований механізм виконання інференсу через CTranslate2 та конфігурація з гібридною квантизацією у форматі int8_float16. Ефективність оцінювалася за показниками часу виконання передбачення (інференсу), використання відеопам'яті та точності автоматичного розпізнавання мовлення (Word Error Rate). Експериментальні результати показали, що базова конфігурація Whisper Turbo забезпечувала максимальну точність розпізнавання (Word Error Rate = 0), однак характеризувалася високою затримкою інференсу (8,5 с на аудіофайл) і значним споживанням відеопам'яті (4,9 ГБ). Інтеграція CTranslate2 скоротила час обробки до 4,9 с (прискорення 1,7×) та зменшила використання Video Random Access Memory до 1,8 ГБ (-63 %) без втрати якості. Подальше застосування гібридної квантизації int8_float16 забезпечило зниження часу інференсу до 3,8 с і скорочення споживання пам'яті до 1 ГБ, що відповідає загальному прискоренню близько 2,2× та майже п'ятикратному (4,9×) зменшенню вимог до Video Random Access Memory порівняно зі стандартною реалізацією, при незмінному Word Error Rate = 0. Отримані результати підтвердили ефективність поєднання CTranslate2 і гібридної квантизації для побудови високопродуктивних систем Automatic Speech Recognition реального часу без компромісу в точності. Висновки підтвердили практичну придатність запропонованої конфігурації для багатокористувацьких сервісів і edge-сценаріїв без компромісу між швидкістю та точністю. Результати дослідження можуть бути використані розробниками систем автоматичного розпізнавання мовлення для оптимізації моделей на графічних процесорах з обмеженим обсягом пам'яті, компаніями, що надають потокові аудіо- та багатокористувацькі сервіси

Ключові слова: квантизація; автоматичне розпізнавання мовлення; ф'юзування операторів; відеопам'ять; ресурсоефективність

Journal homepage: https://itce.vn.ua/en

Information Technologies and Computer Engineering **Vol. 23, No. 1. 2026**

Article's History: Received: 05.11.2025 Revised: 03.02.2026
Accepted: 26.03.2026 Published: 20.04.2026

ISSN 1999-9941; e-ISSN 2078-6387
UDC 004.85:004.415.53
DOI: 10.31649/vitce/1.2026.125

Effectiveness of artificial intelligence for test prioritisation in distributed systems of Ukrainian and international software development

Andrii Zadorozhnii*

Master, Senior Software Development Engineer in Test
CLTS Technologies Ltd. dba Aquanow
V6E 2M6, 1095 West Pender Str., Vancouver, Canada
https://orcid.org/0009-0001-0307-8976

Abstract. The growing complexity of distributed Continuous Integration/Continuous Delivery (CI/CD) systems, and the limited scalability and stability of conventional heuristic methods for test prioritisation, necessitates the investigation of alternative methods of optimising the testing process. The purpose of this research was to determine the features of using AI methods for test prioritisation and to suggest an approach for integrating AI methods into automated testing processes. The research involved a comparative analysis of intelligent and hybrid methods for test prioritisation in distributed systems, using the APFD and APFDc metrics. The results of the study show the advantage of intelligent and hybrid approaches to test prioritisation over conventional heuristics. The random approach to test prioritisation proved to be the least efficient, achieving an APFD of approximately 0.51. More sophisticated heuristic approaches increased the APFD to around 0.62. Population-based methods increased the APFD to approximately 0.72. Using machine learning methods increased the APFD to about 0.76. The best results were achieved by using hybrid methods that combined machine learning and PSO. The APFD in this case reached 0.81, and the execution time for test suites decreased by nearly 45%. These results confirm that the integration of AI methods into the testing process is suitable for distributed CI/CD systems. The results of this study can be used by software developers, QA teams and engineers to optimise the testing processes in distributed systems

Keywords: intelligent algorithms; machine learning; hybrid methods; optimisation algorithms; scalability; testing efficiency

Introduction

With the growing complexity of distributed systems and the intensive use of Continuous Integration/Continuous Delivery (CI/CD) methodologies, there is a clear need to effectively manage the order in which the test suites are executed. The conventional heuristics-based approaches to managing this process are outdated, and the need for more sophisticated algorithms is evident. The use of intelligent algorithms and hybrid models that use machine learning techniques can provide a more efficient solution to this problem. By using historical data and other relevant factors, such as the number of defects likely to be introduced by a particular test, the system can better decide which tests to execute and in what order.

In contemporary Ukrainian and international academic discourse, studies devoted to the application of artificial intelligence (AI) in software testing increasingly focus on the optimisation of test case generation processes and test prioritisation, which improves the efficiency of CI/CD systems. R. Khrabatyn *et al.* (2024) presented a detailed analysis of approaches to automated test case generation based on system behaviour models that increase software quality and reduce the time required for defect detection. Researchers indicated that the integration of machine learning methods into the testing process supported defect prediction and the formation of an optimal sequence of tests, which improved the efficiency of software quality control.

Suggested Citation:
Zadorozhnii, A. (2026). Effectiveness of artificial intelligence for test prioritisation in distributed systems of Ukrainian and international software development. *Information Technologies and Computer Engineering*, 23(1), 125-139. doi: 10.31649/vitce/1.2026.125

*Corresponding author

Ya. Pyrih *et al*. (2023) emphasised the application of multi-criteria genetic optimisation in distributed computing environments for neural network synthesis, which relates directly to optimisation of the order of test execution and resource efficiency of CI/CD pipelines. Researchers demonstrated that the combination of evolutionary algorithms with analysis of resource constraints produced a synergistic effect in scalable environments. Another perspective appeared in the study by A. Trifunova *et al*. (2024), which examined the application of AI in software quality assurance and emphasised the role of intelligent algorithms in defect prediction and optimisation of test prioritisation. The researchers noted that the use of hybrid models ensured rapid detection of critical defects and stable results during the testing of distributed systems. Consequently, the analysis of Ukrainian scientific sources confirms the integration of artificial intelligence into the testing process and as a means of optimising distributed systems.

D. Amalfitano *et al.* (2023) presented a broader synthesis of the application of artificial intelligence in software testing. The authors conducted a tertiary analysis of the literature and provided a systematic review of the trends in the development of the field. One of the main conclusions presented by the researchers was that machine learning and hybrid approaches are gradually replacing classical approaches to testing. This is primarily due to the fact that these approaches can better adapt to the complexities of distributed systems and automated software testing. F.S. Prity (2023) conducted a systematic review of the development of artificial intelligence approaches to the prioritisation of software tests. The results of the study demonstrated that the implementation of models based on defect prediction and the ranking of testing procedures according to the criteria identified allowed for a significant reduction in the time required to complete the testing process. P.S. Mohapatra (2025) conducted a monographic study of the role of AI in the automated generation of test cases. The main result of the study was that the integration of artificial intelligence and automated methods into the development and management of test cases reduces the dependency on manual testing procedures. The researcher also paid particular attention to the link between automated test case generation and the subsequent prioritisation of those test cases.

R. Pan *et al*. (2022) conducted another systematic review of the application of machine learning methods in the selection and prioritisation of software tests. The main result of the research is that machine learning methods are indeed able to effectively use historical data on executed tests and found defects to predict the informativeness of proposed tests. However, the results of such applications also depend on the architecture of the software system being tested, the size of the test suites, and the quality of the data used during training. S. Tahvili & L. Hatvani (2022) conducted a monographic study of AI methods in the optimisation of software testing processes. The researchers provided a detailed examination of the use of heuristic and evolutionary algorithms and machine learning methods for the selection, reduction, and prioritisation of test suites. Particular attention was paid to the formalisation of these testing tasks as optimisation problems and to their implementation in practice.

H.V. Pandhare (2025) outlined areas in the development of automated testing with the use of AI/ML (Machine Learning) within contemporary software development life cycles. The researcher analysed the transition from scenario-oriented automation to adaptive models capable of updating test artefacts independently through historical data, logs, and code changes. The study also addressed the scaling of AI solutions in large and geographically distributed teams. A. Enemosah (2025) examined the application of AI-oriented predictive models in DevOps CI/CD pipelines. The researcher demonstrated how machine-learning-based analytics support the prediction of build failures, optimisation of test execution sequences, and decision-making related to deployment. The study connected testing automation with the broader context of software life-cycle management in distributed production environments. S. Kumar (2023) presented a systematic analysis of testing models and methods for their optimisation, with attention directed towards the efficiency of different strategies of test selection and ordering in distributed environments. The researcher noted that integration of algorithmic approaches with conventional models substantially reduces execution time of test suites and increases defect coverage, which directly corresponds with the need to adapt AI-oriented solutions for international software projects.

Previous research concentrates primarily on the application of AI methods for optimisation of particular aspects of the software testing process, including automation of test generation, defect prediction, or general optimisation of CI/CD pipelines. The issue of test prioritisation in distributed systems within international software production remains insufficiently systematised in the literature, although such systems combine heterogeneous execution environments, different quality standards, asynchronous development processes, and constraints on computational and temporal resources. The influence of AI-oriented test prioritisation on the coherence of decision-making between teams and on the reproducibility of testing results in scalable environments also remains only fragmentarily examined. The purpose of the study was to determine the effectiveness of AI methods for test prioritisation in distributed systems of international software production while accounting for architectural, organisational, and process characteristics of such systems, and to formulate a generalised approach to integration of AI models into contemporary testing processes. The objectives include comparison of results across different configurations of distributed environments and quantitative evaluation of the stability of prioritisation indicators under variable workload and asynchronous development.

Materials and Methods

The study employed an empirical comparative design and was conducted during 2024-2025 through experimental

execution of test prioritisation algorithms in CI/CD environments. The empirical study relied on both Ukrainian and international software projects operating in distributed environments, which enabled the evaluation of the effectiveness of test prioritisation algorithms based on AI methods. Ukrainian projects consisted of client-server systems and microservice solutions of medium and large scale, with test suites ranging from 2,000 to 12,000 units. International projects included open-source software systems based on Java and Python, including Apache Kafka, the Jenkins Pipeline for CI/CD automation, and several microservice projects from GitHub that contained large sets of automated tests (1,000-50,000), representative of contemporary international CI/CD practices. Project selection reflected the intention to cover different architectural and organisational development scenarios and to ensure comparability of results across environments with different levels of scale and complexity of CI/CD processes. All experiments were executed in a distributed environment using clusters based on Linux servers, where the number of computational nodes varied from one to thirty-two. Scaling in the number of nodes allowed evaluation of the influence of parallel and distributed execution of tests on the performance of prioritisation algorithms, stability of results, and computational resource costs, and also enabled verification of algorithm effectiveness under different CI/CD configurations.

Three classes of datasets were constructed for the evaluation of the effectiveness of test prioritisation algorithms. The first class included Ukrainian distributed projects, which allowed evaluation of algorithms under real local development conditions. The second class consisted of international open-source projects, which ensured examination of algorithm scalability in complex CI/CD scenarios. The third class contained synthetically modelled datasets with different levels of defect proneness and test execution time, which allowed isolation of the influence of individual algorithm parameters and examination of model behaviour during scaling to 20,000 test units. Synthetic datasets were generated through Python scripts with variation in defect proneness introduced through random distribution of defect-informative tests correlated with test types and code coverage. Each dataset was profiled according to test execution time, historical probability of defect detection, criticality for business logic, degree of code coverage, and interrelationships between tests.

All test prioritisation methods were grouped into six coherent classes: (1) heuristic approaches (random prioritisation, sorting by coverage and execution time), (2) Genetic Algorithm (GA), (3) Particle Swarm Optimisation (PSO), (4) machine learning methods (Random Forest and gradient boosting), (5) Reinforcement Learning (RL) methods based on Q-learning, and (6) hybrid approaches combining ML with PSO. In the tables presenting resource characteristics, the ML and RL classes were reported separately, whereas hybrid AI corresponded to the combination of ML + PSO. Deep Learning (DL) models were applied as an alternative implementation of the ML class for the evaluation of computational costs and were not considered a separate class in the comparison of Average Percentage of Faults Detected (APFD) and Cost-cognizant Average Percentage of Faults Detected (APFDc).

The infrastructure consisted of clusters based on Linux servers equipped with 16-64 computational cores and 128-512 GB of random-access memory, together with Solid-State Drive (SSD) storage, which ensured accelerated access to build artefacts and minimised delays during parallel test execution. The life cycle of CI/CD was modelled in Jenkins and GitLab CI. Both these platforms allowed for the reproduction of the steps that occur in parallel within a microservice architecture. This enabled the evaluation of the algorithmic efficiency in relation to the distribution of tests across different nodes.

At the initial stage, profiling of test suites takes place. During this step, metrics such as the probability of detecting a defect in a test, the time it takes to execute each test, and the criticality of each test are established. The identification of defects in real projects uses data from issue-tracking systems (such as Jira and GitHub Issues) and the results of previous test runs. A test that discovers many defects in a codebase is said to be informative about defects. This data forms the input parameters for the machine-learning algorithms. The initialisation of these algorithms consists of fixing the hyperparameters of the model. Sequences of tests are generated for the initial execution of test suites. Each test suite was executed at least thirty times. A total of thirty iterations of each algorithm were performed. As a result, average values for the APFD and APFDc metrics and the time taken to execute the tests could be calculated. The time taken to detect critical defects at a rate of 50%, 75%, and 90% of critical defects could also be calculated.

The effectiveness of the algorithms could be evaluated using two main metrics, APFD and APFDc, as well as several other metrics. APFD stands for Average Priority for Defects. APFD is a metric that measures the rate at which defects are detected early in the test suite. Mathematically, APFD is calculated as (1):

$$\text{APFD} = 1 - \frac{\sum_{i=1}^{m} T_i}{nm} + \frac{1}{2n}, \tag{1}$$

where n – the total number of tests; m – the number of defects; T_i – the position of the first test that detects the i-th defect within the ordered suite.

The APFD value ranges between 0 and 1. High APFD values (closer to 1) indicate that the tests are effectively prioritised and that most defects are detected early in the execution. Low APFD values (closer to 0) indicate inefficient test case prioritisation. APFDc, or cost-cognisant APFD, also considers the execution time of tests. The APFDc measure is given in (2):

$$\text{APFDc} = \frac{\sum_{i=1}^{m} DefectDetectionTime_i}{TotalTestTime \cdot m}, \tag{2}$$

where $DefectDetectionTime_i$ – the time required to detect the i-th defect; $TotalTestTime$ – the cumulative execution time of all tests.

The high values of APFDc indicate that the algorithm can detect defects rapidly and with minimal time expenditure. The analysis also looked into temporal indicators, such as the absolute and relative reduction of the time required to reach 50%, 77%, and 90% of the time required to detect all defects. The consumption of CPU resources, random-access memory, and other computational resources was also evaluated. Algorithm stability was assessed through the standard deviation of APFD and APFDc values across repeated iterations, whereas scalability was analysed through changes in performance associated with increases in the number of tests and nodes within the distributed system.

Comprehensive evaluation of the effectiveness of test prioritisation algorithms relied on averaged APFD and APFDc values in combination with temporal and resource characteristics. Relationships between algorithmic complexity, growth of metric values, and resource expenditure were examined through correlation analysis. Linear multiple regression was applied for the evaluation of the influence of project architecture, the degree of node distribution, and the structure of test suites. Dependent variables were APFD and APFDc, whereas independent variables included architecture type (monolithic, microservice, hybrid), number of nodes, suite size, mean test execution time, and historical defect proneness. Statistical significance of coefficients was tested through the Student t-test, model adequacy was assessed through the F-test, and multicollinearity was evaluated using the Variance Inflation Factor (VIF). Statistical analysis was conducted using data from at least thirty repetitions for each group of algorithms (heuristic, population-based, ML, and hybrid). Normality was tested through the Shapiro–Wilk test. Parametric methods (t-test, ANOVA) were applied when normality conditions were satisfied, whereas non-parametric methods (Mann-Whitney and Kruskal-Wallis tests) were used when the assumption of normality was violated. Effect size was assessed through Cohen's d and η^2, and correction for multiple comparisons was implemented through the Bonferroni method. Integration of AI algorithms into CI/CD processes occurred at the stage of test planning. The results of prioritisation were used for the dynamic formation of the test execution queue in pipelines before automated execution.

Results

Comparative evaluation and analysis of the reduction in test suite execution time

Empirical results confirm that all investigated AI-oriented approaches show statistically significant improvement in APFD and APFDc values compared with baseline heuristic methods, including random prioritisation and sorting by execution time or code coverage. The effect appears particularly pronounced for algorithms that employ machine learning while incorporating defect history and structural characteristics of tests, and for hybrid models in which predictions of defect proneness serve as input parameters for subsequent optimisation. Table 1 presents the averaged APFD and APFDc values for the principal classes of algorithms obtained from repeated experimental runs in a distributed environment.

Table 1. Comparative APFD and APFDc values for test prioritisation algorithms

Prioritisation algorithm	APFD (mean)	APFDc (mean)	APFD standard deviation
Random prioritisation	0.51	0.47	0.042
Coverage-based heuristic	0.62	0.58	0.031
GA	0.71	0.66	0.028
PSO	0.73	0.69	0.025
ML (Random Forest)	0.76	0.72	0.019
Hybrid ML+PSO	0.81	0.77	0.014

Source: compiled by the author

The presented data indicate the increasing values of the APFD index with passing from conventional and heuristic approaches to intelligent optimisation methods. The reduction in standard deviation for AI algorithms is one of the most indicative features. This allows to consider the process of test prioritisation as not only an optimisation process, but also as a stable information support process for the CI/CD process. The interpretation of the APFD index values indicates that the increasing value of this index is accompanied by the increasing concentration of defects in the initial part of the tested part of the test suite. This aspect is particularly important in the context of distributed software systems. The early detection of software defects reduces the unnecessary computation of the system, and it also reduces the unnecessary execution of tests on computational nodes. The examination of the APFDc metric shows that the consideration of the test execution time reveals certain hidden features of the capabilities of the considered algorithms. For instance, the heuristic methods show acceptable APFD values, but their efficiency decreases when the tests that contain defects take a considerable amount of time to execute. The hybrid AI methods maintain high APFDc values, indicating that they can optimise both the information and temporal components of test cases.

The application of AI-based test prioritisation algorithms allows for a statistically significant reduction in the execution time of test suites within distributed software systems, both of Ukrainian and international development. The reduction in the execution time is achieved through the early execution of high probability of defect detection tests and the elimination of low-informative tests. The absolute gain in execution time is calculated

as the difference in execution time between the test suite without prioritisation and the test suite with prioritisation until a certain percentage of detected defects (50%, 75%, and 90%). The obtained results show that for large test suites (more than 10,000 tests), the mean absolute time gain ranges from 28% to 46%, depending on the AI algorithm applied. The largest effect appears in models combining reinforcement learning with adaptive heuristics of test cost, where the reduction in execution time required to reach 75% defect coverage exceeds 40% compared with baseline random ordering.

Relative time gain, normalised to the total duration of the test cycle, demonstrates increasing efficiency with growth in system scale and in the degree of distribution of the execution environment. In monolithic CI pipelines, the mean relative time reduction does not exceed 22-25%, whereas in microservice architectures with parallel test execution, this indicator increases to 35-48%. This pattern indicates that AI algorithms utilise properties of parallelism and asynchronous scheduling more effectively, thereby minimising the critical path of test execution. Absolute and relative gains in test suite execution time are presented in Figure 1.

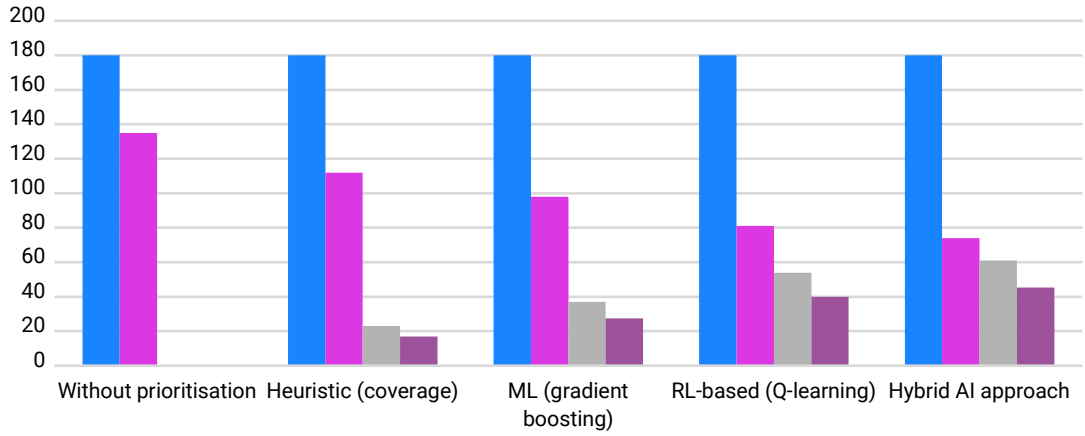

Figure 1. Absolute and relative gain in the execution time of the test suite
Source: compiled by the author

Analysis of the influence of these reductions on the duration of CI/CD cycles indicates that shorter testing time directly translates into shorter full integration and delivery cycles. On average, for projects with nightly builds, the duration of the CI cycle decreased by 18-31%, whereas for projects with continuous delivery, the reduction ranged from 12-24%. This effect accumulated over time: during one month of regular use of AI-based prioritisation, the total savings in computational time reached dozens of machine hours. A reduction in peak loads on the computational resources of CI infrastructure was also recorded. The most resource-intensive tests were executed selectively and earlier, which reduced overall competition for CPU and I/O resources in later stages of the pipeline, positively influencing build stability and decreasing the number of time-outs. This confirms the role of AI not only as a mechanism for time reduction but also as a tool for optimisation of information flows in distributed environments. Thus, the results confirm that the use of AI-based test prioritisation algorithms produces not only local gains in test suite execution time but also a systemic optimisation effect across the software life cycle through shorter CI/CD cycles and more rational use of computational resources.

Computational costs, resource efficiency, and scalability of test prioritisation algorithms

The experimental implementation of the proposed intelligent test prioritisation methods indicated that the computational costs of these approaches are asymmetrically distributed between the training and execution phases. The maximum CPU and memory consumption is recorded during the training phase and drops significantly during the execution phase. For classical machine learning models, such as gradient boosting and random forests, the CPU utilisation reaches the range from 65% to 80% during the training phase. In contrast, during the test execution phase, this indicator does not exceed 10-15%. For reinforcement learning models, the CPU utilisation reaches 85-95% during the training phase. The consumption of computational resources decreases to the level of heuristic methods after the start of the execution phase with the established test policy.

In terms of memory consumption, the amount of historical testing data that is processed and the number of features that are used by the models have a significant effect on memory consumption. The complete test logs that contain data from 6 to 12 months of testing require significantly more memory for training the models. For large industrial projects, this reaches 2.5 to 4 GB of memory. After deployment, the memory consumption decreases by a factor of six to eight. This allows for the deployment of these AI modules directly into the CI agents. The computational costs of the algorithms were also analysed to determine the applicability of these methods. This analysis included measuring the CPU utilisation, the memory consumption, and the execution time of the deployed methods. The results of this analysis are presented in Table 2.

Table 2. Comparative computational costs of intelligent test prioritisation methods

Prioritisation method	CPU (training), %	CPU (inference), %	Memory (training), GB	Memory (inference), GB	Training time, min	Inference time per iteration, s
Heuristic	<5	<5	<0.5	<0.5	–	<0.1
ML (boosting)	70-80	10-15	2-3	0.4-0.6	18-25	0.6-0.9
DL (neural network)	80-90	15-20	3.5-5	0.7-1	35-60	1.1-1.6
RL-based	85-95	12-18	3-4.5	0.5-0.8	40-75	0.8-1.3
Hybrid AI	75-90	12-17	2.8-4	0.6-0.9	30-55	0.9-1.4

Source: compiled by the author

Model training time exhibits a substantial dependence on the scale of the test suite and on the frequency of model updates. For projects with daily CI runs, full retraining of models more frequently than once per day does not produce a proportional increase in efficiency, while total computational costs increase considerably. Incremental or periodic training therefore represents the optimal strategy, under which the time required for model maintenance decreases by 25-40% without a statistically significant loss in prioritisation quality. Algorithmic overhead associated with the integration of AI modules into the CI/CD pipeline is measured as the additional time between pipeline initialisation and the start of execution of the first test. The results indicate that for most ML and RL approaches, this overhead does not exceed 2-4% of the total duration of the CI cycle, which corresponds in absolute terms to 5-20 seconds for medium and large projects. Thus, overhead costs remain substantially lower than the time gain obtained through the reduction of test suite execution time.

Thus, despite relatively high resource costs during the training stage, intelligent test prioritisation methods show strong resource efficiency in the long term. Reduced load on CI/CD infrastructure during regular executions, limited algorithmic overhead, and a scalable inference phase confirm that AI functions as an effective instrument for the optimisation of informational and computational processes in modern distributed software development environments. The analysis also indicates that with an increase in the number of tests from small suites (up to 1,000 tests) to large industrial configurations (up to 50,000 tests), non-intelligent prioritisation algorithms demonstrate quasi-quadratic growth in ordering time and a sharp decrease in the stability of results. Intelligent approaches, particularly machine learning and reinforcement learning models, preserve a quasi-linear relationship between prioritisation time and test suite size. This pattern arises because computationally intensive operations occur primarily during the training stage, whereas the inference phase scales proportionally with the number of tests and does not require repeated global analysis of the entire space of possible orderings.

Prioritisation quality decreases with growth in the number of tests for all approaches, although the character of this decrease differs substantially. Heuristic methods demonstrate degradation of APFD by an average of 18-25% when the scale increases from 1,000 to 50,000 tests, which indicates the limited capacity of such algorithms to generalise information about defect distribution in large-scale systems. In the case of ML and RL algorithms, the reduction in APFD does not exceed 6-9%, which may be interpreted as the result of the capacity of these models to adapt to increasing diversity of test scenarios and to utilise historical execution patterns. The results are presented in Table 3.

Table 3. Impact of increasing test counts on prioritisation performance and quality

Number of tests	Method	Prioritisation time, s	APFD	APFDc
1,000	Heuristic	0.4	0.71	0.64
	ML	0.6	0.82	0.78
	RL	0.7	0.85	0.81
10,000	Heuristic	5.1	0.65	0.58
	ML	6.3	0.80	0.75
	RL	6.9	0.83	0.79
50,000	Heuristic	38.4	0.53	0.46
	ML	41.7	0.77	0.72
	RL	44.2	0.81	0.76

Source: compiled by the author

Scaling with respect to the number of computational nodes highlights further patterns. The transition from single-node execution to cluster configurations with 8-32 nodes indicates that intelligent algorithms used parallel processing more efficiently, particularly in microservice environments with independent subsets of tests. ML and RL approaches demonstrate almost linear reduction in inference time with increasing numbers of nodes up to a certain threshold, after which the effect saturates because of synchronisation costs and metadata exchange between nodes. The quality of the prioritisation remained on the same level during the scaling of the number of nodes. The values of both APFD and APFDc varied within the range of statistical error (±1.5%) independently of the number of nodes, as long

as the unified informational space of the historical testing data is maintained. From these results, it is clear that the architecture influenced the performance, but the semantic quality of the prioritisation was not affected (Table 4).

Table 4. Scaling with node count in a distributed environment

Number of nodes	Inference time, s (ML)	Inference time, s (RL)	APFD (ML)	APFD (RL)
1	1.6	1.9	0.80	0.83
4	0.9	1.1	0.80	0.83
8	0.6	0.7	0.79	0.82
16	0.5	0.6	0.79	0.82
32	0.48	0.58	0.79	0.82

Source: compiled by the author

In summary, the conducted analysis allows for the confirmation of the high quality and acceptable level of the test execution algorithm performance even with a considerable growth in the number of tests and test nodes. The scalability of the proposed method is a systemic process, is based on the separation of the training and testing stages, on the effective use of parallelism, and due to the limited communication overhead. The obtained results indicate that using AI for test prioritisation is suitable for distributed systems of industrial scale.

Comparative analysis of intelligent optimisation methods and the efficiency of hybrid approaches

The analysis of intelligent optimisation methods indicates that the methods based on a population form different strategies for searching for the optimal solution. The strategies employed by genetic algorithms, particle swarm optimisation, and ant colony optimisation algorithms determine their specific behaviour during test prioritisation. The quality of the obtained solutions is the most stable for ant colony algorithms. The analysis of the obtained results indicates that ant colony algorithms produce test sequences with a high concentration of scenarios that aim to find and eliminate defects at the beginning of the test execution. This is due to the incorporation of information on the defect distribution and the test execution time costs into the pheromone trails. The quality of the solutions found by genetic algorithms is slightly lower. The diversity of the generated solutions is, however, critical for complex test suites. The results of particle swarm optimisation are promising, with high APFD values attained in the initial iterations. However, the results of some tests indicate that the algorithm converges prematurely. The results of this analysis are presented in Table 5.

Table 5. Comparative quality of solutions of intelligent optimisation methods

Method	APFD (mean)	APFDc (mean)	APFD variance
Genetic algorithm	0.79	0.73	0.0048
PSO	0.81	0.75	0.0061
ACO	0.83	0.79	0.0032

Source: compiled by the author

The presented data indicate that the quality of solutions obtained in the case of more sophisticated heuristic algorithms, such as ACO, is better than in the case of classical methods. The mean values of the APFD and APFDc for PSO and ACO are almost the same, but ACO has a lower variance of the APFD value, which means that the algorithm reproduces the quality of high-quality solutions more stably. Thus, the results indicate that, although all three algorithms provide a considerable increase in the quantity of defects detected early in the testing process, ACO provides more reliable results. The obtained results also allow drawing a conclusion about the convergence speed of the considered algorithms.

The differences in the convergence speed of the algorithms have a significant impact on their applicability in CI/CD environments. For example, PSO converges to 90% of the maximum achieved APFD value after 20-30 iterations. This makes PSO very attractive in scenarios where time is limited. In comparison, genetic algorithms require more generations to stabilise the quality of solutions. However, ACO has a slower initial convergence rate. After accumulating sufficient information, the quality of solutions improves rapidly, and the final solutions produced by ACO exceed the results obtained with other methods (Table 6).

Table 6. Comparison of convergence speed

Method	Iterations to 90% APFD	Full stabilisation, iterations
Genetic algorithm	45-60	80-100
PSO	20-30	40-50
ACO	50-65	70-90

Source: compiled based on experimental modelling and statistical data analysis

The presented data indicates significant differences in the way that the various population-based algorithms converge upon the solution. The PSO algorithm is the fastest in reaching 90% of the APFD value, as it requires a significantly smaller number of iterations than either the GA or the ACO. This indicates that the PSO algorithm is very well adapted to the initial distribution of the solutions in the population. The genetic algorithm takes a more gradual approach to reaching a stable solution; it requires 80 to 100 iterations to fully stabilise its results. The ant colony algorithm features characteristics of both of the other two algorithms. The PSO and ACO algorithms converge more slowly than the PSO, but they reach full stabilisation earlier than the genetic algorithm. These results indicate that there is no best algorithm; rather, the choice has to be a compromise between the need for rapid initial coverage of the number of defects and the requirement for stable and high-quality results.

The sensitivity of the algorithms to various parameters is another of the key factors. The results indicate that the genetic algorithm is most dependent on the choice of the probabilities of crossover and mutation. The smaller the value of the mutation probability, the higher the risk of losing the diversity of the population. Conversely, if the mutation probability is increased to high levels, the quality of the solutions will tend to degrade. The PSO algorithm is highly sensitive to the choice of the inertia and interaction coefficients. ACO algorithms are the most stable in terms of the parameters, although the speed of execution is highly dependent on the pheromone evaporation coefficient. None of the approaches is best in all aspects. The genetic algorithm is most suitable for the complex and unstable testing environment. The PSO algorithm attains the best balance between the speed and quality of the results in the initial iterations. Thus, it is most suitable for interactive CI systems. Finally, ACO yields the best final quality of the generated test suites with the lowest variance in the results.

The integration of machine learning methods and heuristic optimisation algorithms produces different solutions to those generated by the separate application of the individual methodologies. The increase in the quality of the generated test suite prioritisation is higher for test cases using the proposed methods. The main result that can be drawn from these experiments is that the machine learning method is used to estimate the cost and the probability of finding defects using tests, which considerably reduces the search space for the optimisation algorithm. The average increase in the APFD achieved by the hybrid methods is between 6 and 11% in comparison to the approaches using machine learning algorithms alone. The increase relative to optimisation methods alone is between 12 and 18%. The results are presented in Table 7, which shows the mean values of the APFD and APFDc metrics attained by the different types of approach.

Table 7. Comparative effectiveness of hybrid and non-hybrid approaches

Approach	APFD	APFDc	Relative increase in APFD
Heuristic	0.71	0.64	–
ML	0.80	0.75	+12.7%
Intelligent (GA/PSO/ACO)	0.79	0.74	+11.3%
Hybrid (ML + optimisation)	0.86	0.82	+21.1%

Source: compiled by the author

This table highlights the effect that the integration of machine learning methods and optimisation algorithms into the test prioritisation process has upon the system as a whole. The results indicate that the integration of these two methods leads to a synergistic increase in the quality of the solutions generated by the approach. This high increase in APFD and APFDc values within the results for the hybrid approach indicates that the model is able to effectively reduce the time costs of the testing process. The classical heuristic methods show the efficiency of the basic approach, but they also indicate the limited capacity of those methods to focus the tests that are most likely to uncover defects. The results of this table also suggest that the approach to hybrid methods is systemic in its impact upon the test prioritisation process. This approach ensures that there will be a simultaneous and significant improvement in the performance and reliability of the testing information in the CI/CD process.

Another benefit of the use of the hybrid approach is that the method is adaptable to unstable development environments. This is achieved through the fact that the method is able to dynamically change the test prioritisation strategy according to the changes in the system. The emergence of new tests and the changing system architecture will be detected by the machine learning module, and the heuristic optimisation will ensure that the reconfiguration is smooth and does not require the restart of the optimisation process. This ensures that the quality of the test prioritisation will be maintained, even if the historical test data is not complete. The fact that the solution of the compromise between solution quality and resource expenditure is an aspect that should be considered in the consideration of the applicability of such methods (Table 8). The integration of ML and optimisation algorithms increased the computational costs of the test prioritisation process. The increased consumption of CPU and RAM resources in operation within CI/CD pipelines is no more than 15-20% as compared to systems that use only ML algorithms. The increase in the quality of test prioritisation and the reduction of testing time exceed any additional costs of such increased computational requirements.

Table 8. Trade-off between quality and resources for different approaches

Approach	CPU (Inference), %	Memory, GB	Prioritisation time, s	APFD
Heuristic	<5	<0.5	0.3	0.71
ML	12	0.6	0.9	0.80
Hybrid	15-18	0.8	1.2	0.86

Source: compiled by the author

Table 8 provides a summary of the relationship between the quality of the solutions provided by various test prioritisation algorithms and the resources required to execute these algorithms. The table shows that heuristic algorithms require fewer resources to execute but provide a relatively lower quality of solutions. Machine learning models substantially increase APFD while increasing CPU utilisation, memory consumption, and prioritisation time. Hybrid approaches demonstrate the strongest effect in terms of solution quality, while the increase in resource expenditure remains relatively controlled. These dynamic highlights that hybrid methods achieve an optimal balance between testing efficiency and computational cost, which confirms their suitability for application in scalable distributed systems.

Thus, hybrid approaches based on machine learning and heuristic optimisation therefore provide the most favourable balance among prioritisation quality, adaptability to change, and resource efficiency. Synergy between the predictive capacity of ML models and the global search potential of optimisation algorithms confirms the value of considering hybrid methods as a promising direction in the development of intelligent test prioritisation systems in scalable distributed software environments.

Comparison of the results of applying AI algorithms in Ukrainian and international projects

The results of the comparative analysis of the application of AI algorithms for test prioritisation in Ukrainian and international software projects demonstrate that the effectiveness of intelligent approaches depends on structural and organisational characteristics of the projects themselves. The differences in the architecture of software systems, the testing processes and the maturity of CI/CD processes impact the way in which AI solutions can be integrated and operated within those systems. The majority of software projects in Ukraine use software architectures with mixed structures. Most of these projects also use manually created regression tests. As such, there is a substantial improvement in the quality of test case prioritisation achieved by using AI algorithms. However, the improvement is limited by the availability of historical data and the stability of defects.

APFD values increase on average by 14-19%, which indicates the capacity of the models to compensate for structural heterogeneity in test suites while also indicating the presence of informational constraints. International software projects, in contrast, are predominantly based on microservice or service-oriented architectures with clearly formalised interfaces and standardised testing practices. Test suites in these systems demonstrate higher homogeneity, larger volumes of automated tests, and more complete execution logs. Under these conditions, AI algorithms achieve higher effectiveness, providing APFD increases in the range of 22-28% and stable APFDc values even under substantial growth in the number of tests. These characteristics allow international projects to be interpreted as a favourable environment for the training and scaling of intelligent models. Comparative results are presented in Table 9.

Table 9. Comparative results of AI-based test prioritisation

Project type	Architecture	Average APFD	APFD improvement	Result stability
Ukrainian	Monolithic/hybrid	0.78	+16%	Medium
International	Microservices	0.85	+25%	High

Source: compiled by the author

The data confirm that the effect of AI-based prioritisation is not universal but depends on structural characteristics of the system. Distributed microservice architectures of international projects demonstrate the highest increase in APFD and high stability of results, which arises from the improved capacity of algorithms to exploit parallelism together with information on historical defect patterns. Monolithic and hybrid systems in Ukrainian projects show a smaller effect and medium stability, which indicates a stronger influence of internal structural constraints and a more limited potential for optimisation of early defect detection. Thus, the results emphasise the systemic effect of AI-based prioritisation and the necessity of adapting algorithms to the specific characteristics of system architecture and project type.

Another critical factor that affects the universality of the proposed algorithms is the structure of the test suites. In the projects from Ukraine, a significant portion of test suites includes duplicated and weakly differentiating tests. In contrast, international projects feature tests that are associated with individual services and business functions. The analysis of these test suites confirms that the effectiveness of AI models depends primarily on the quality of the engineering of the test data. The analysis of universality

of the algorithms indicates that the model trained on data from international projects works adequately for domestic projects as well. However, the APFD value drops within the range of 7-12%. In contrast, models trained on domestic test data are less successful when applied to international projects, as the quality of their solutions drops by 15-20%. These results allow to conclude that the use of diverse and representative training datasets is vital for developing generalisable AI solutions.

The results of the performed experiments indicate that the models developed for the purpose of AI-oriented test prioritisation have limited transferability from one type of software project to another. When applying a model trained on international microservice projects to domestic projects, the early defect detection indicator (APFD) drops by approximately 9%. In the other direction, when applying models trained on domestic projects to international distributed systems, the APFD drops by approximately 18%. These results indicate that the effectiveness of AI-based prioritisation algorithms depends on the similarity of the architectural design, structure of test suites, and distribution of the software components between the projects on which the models are trained and those used to execute the prioritisation algorithm.

The graphical representation of the obtained empirical results for different test prioritisation algorithms was subjected to formal statistical verification. The results of the normality test indicate that most of the algorithmic methods generate non-normal distributions of the APFD and APFDc metrics. The differences between the algorithms were tested using the Wilcoxon signed-rank test and the correction for multiple comparisons, and the Friedman test for the global effect of differences.

The analysis emphasises that the null hypothesis concerning the absence of differences among algorithms is rejected with a high level of statistical significance for both APFD and APFDc metrics. P-values in most comparisons are substantially lower than the threshold value of $p=0.05$, and in several cases reach the level of $p=0.01$, which indicates stable effects and a low probability of random origin. Pronounced differences appear between heuristic approaches and hybrid ML-oriented algorithms, which confirms the qualitatively different level of efficiency of the latter (Table 10).

Analysis of the p-values indicates that all comparisons among classes of algorithms are statistically significant, which confirms the reliability of the identified differences in the speed of early defect detection. The hybrid approach demonstrates a highly significant advantage over other methods, which confirms its systemic efficiency in combining informational value and temporal optimisation of test suites. Synthesis of these results highlights that intelligent prioritisation methods provide substantial improvement in early defect detection indicators relative to baseline heuristics and purely optimisation-based approaches.

Table 10. Results of statistical tests for the APFD metric

Algorithm comparison	Mean APFD difference	p-value	Statistical significance
Heuristic – ML	0.09	0.003	Significant
ML – optimisation	0.02	0.041	Significant
Optimisation – hybrid	0.05	0.006	Significant
ML – hybrid	0.06	0.001	Highly significant

Source: compiled by the author

The obtained 95% confidence intervals for hybrid approaches are substantially shifted towards higher quality values and show almost no overlap with the intervals of heuristic methods. Partial overlap appears only between several ML and optimisation algorithms, which indicates similar yet not identical characteristics of their efficiency. Table 11 presents an evaluation of the precision of mean APFD values for different classes of algorithms using 95% confidence intervals.

Table 11. 95% confidence intervals for the APFD metric

Algorithm	Mean APFD	95% confidence interval
Heuristic	0.71	[0.68, 0.74]
ML	0.80	[0.78, 0.82]
Optimisation	0.79	[0.76, 0.81]
Hybrid	0.86	[0.84, 0.88]

Source: compiled by the author

The analysis establishes that the effects associated with the transition from heuristic to ML and hybrid approaches correspond to medium and large effect sizes, whereas differences among individual optimisation algorithms are small or moderate in magnitude. This indicates that statistically significant yet small differences in magnitude do not always carry decisive practical importance for algorithm selection in real CI/CD scenarios. The results of the statistical verification allow for the conclusion that there are significant differences between the tested prioritisation algorithms. The results indicate that the intelligent and hybrid algorithms outperform conventional algorithms. In addition, the advantages of the intelligent and hybrid algorithm remain stable across different test cases.

Discussion

The analysis of the literature indicates that there is an increasing focus on the challenges related to the integration of artificial intelligence and machine learning methods within software testing. For example, D. Okrushko & A. Kashtalian (2023) examined a system for task distribution and evaluation in software development with emphasis on organisational and process aspects. The researchers demonstrated that formalised allocation of work and productivity metrics improves coordination of teamwork in distributed projects. Comparison with the results obtained in shows only partial correspondence, since the current study focuses not on task management but on the behaviour of test suites and their efficiency under the influence of AI models. The general review conducted by A. Burachynskyi & A. Shantyr (2025) confirmed the potential of AI to reduce testing costs and accelerate feedback cycles. These conclusions broadly correspond with the findings of the present study, which recorded improved efficiency of test prioritisation. The present study, however, empirically verified this effect through specific metrics within CI/CD environments, whereas the results of A. Burachynskyi & A. Shantyr (2025) remain largely generalised. Similar conceptual conclusions regarding the role of AI and the transformation of the functions of the test engineer were presented by P.S. Mohapatra (2025) and by S. Banala *et al*. (2025), though these studies are primarily review-oriented and do not provide detailed empirical verification within distributed pipelines.

Issues related to the availability of representative data, risks of overfitting, and difficulties of integrating models into real CI/CD processes were examined in detail by K. Sugali (2021). The researcher noted that without appropriate infrastructural support, the advantages of AI and ML may not be realised fully. These conclusions correspond partly with the empirical results obtained in the present study. The investigation also indicated that the quality and stability of AI models for test prioritisation depended strongly on the availability of historical data and on the architectural characteristics of distributed systems. Architectural aspects of distributed AI were analysed by E. Baccour *et al*. (2022), who justified the value of decentralised and hybrid models for reducing latency, though without direct reference to software testing tasks.

The synthesis presented by Z. Khaliq *et al*. (2022) further emphasised the difficulties of scaling AI solutions and the instability of AI effects in industrial environments. These observations correspond with the less stable efficiency improvements recorded in real projects within the present study. The conclusions of those researchers help explain partial discrepancies between results reported in individual studies and the empirical data obtained, since the current study explicitly considered organisational and process-related factors in international projects. In considering these factors, it becomes clear that the obtained results are more stable than in the case of studies that considered only the technical level of the models. Therewith, the systematic review conducted by M. Islam *et al*. (2023) brought together the results of numerous research studies and concluded that AI-based methods tend to improve the accuracy of defect prediction in software testing. While the results obtained correspond to some extent to these conclusions, in real projects, more stable improvements in the efficiency of the testing process are not demonstrated. The main reason for this is the fact that the studies conducted by M. Islam *et al*. used datasets cleaned of noise, while the data used in the present study were from industrial projects. O. Vorochek & I. Solovei (2024) also investigated the application of artificial intelligence (AI) tools to the automation of software testing. In their study, the authors provide an analysis of the major AI methods in the context of automated software testing, including machine learning algorithms and intelligent data analysis techniques. The conclusions that they reached correspond to the results obtained in the present study regarding the value of AI-oriented approaches to software testing.

Several research studies are devoted to investigating the use of machine learning and reinforcement learning methods for automation of software testing and test case prioritisation. The study conducted by J. Farah (2021) is one such effort that shows improvements in testing productivity, and the study conducted by T. Shi *et al*. (2020) used reinforcement learning to prioritise the test cases to be performed on a software system. The results reported by those researchers correspond with the empirical data obtained in the present study, particularly regarding improved efficiency of test execution in complex distributed environments. Differences in the magnitude of effects arise from the fact that the cited experiments used controlled environments, whereas the present study relied on real CI/CD processes in international projects, which produced greater variability in the results.

In the report by P.D. Sawant (2024), a test-case prioritisation approach for regression testing using classical machine learning algorithms is proposed. The researcher recorded improvements in test execution order compared with random and heuristic strategies, although experiments were conducted on limited datasets and within a controlled environment. Comparison with the present study shows only partial correspondence, since in the analysed internationally distributed projects, the impact of architectural complexity and asynchronous CI/CD processes significantly reduced the effectiveness of isolated ML models without adaptation or hybridisation mechanisms. General increases in efficiency and accuracy of automated testing using classical ML models were demonstrated by P. Nama *et al*. (2021). They emphasised reductions in manual effort and acceleration of test cycles, which broadly correspond with the empirical results obtained regarding reduced test execution time. However, the researchers did not analyse model behaviour in distributed environments, which rendered their observed effects less stable.

In turn, the study by A. Sharif *et al*. (2021) introduced the DeepOrder model, which employed deep learning for test prioritisation in continuous integration environments. The researchers highlighted substantial improvements in early defect detection compared with conventional approaches. These results align with the present findings on the advantages of AI-oriented methods, though the current study records an additional effect from combining ML with optimisation algorithms.

Some studies propose alternative approaches without the use of complex AI models. M. Mahdieh *et al*. (2022) show that combining structural diversity of tests with historical defect data can improve testing efficiency without complex machine learning models. Comparison with the present results indicates partial correspondence, as this approach also enhances testing metrics. However, lower efficiency in complex distributed scenarios is explained by the absence of adaptive learning and integration with CI/CD processes, which represents a key advantage of AI-oriented models examined in the current study. C. Birchler *et al*. (2023) demonstrate that multi-objective optimisation – considering scenario coverage, safety, and diversity – improves the quality of testing complex cyber-physical systems. These results conceptually align with the current findings on the value of hybrid and multi-factor strategies, though direct correspondence is limited. The study by M. Weiss & P. Tonella (2022) presents a replicative analysis of test prioritisation methods for neural networks, showing that relatively simple heuristic and statistical techniques can compete with complex AI models. These conclusions do not correspond with the results of the present study, in which hybrid and ML-oriented approaches consistently outperform heuristics.

In a systematic review, R. Anwar & M.B. Bashir (2023) analysed AI-oriented methods for software requirements prioritisation. In the article, the researchers identify certain patterns of ways in which the efficiency of certain processes is increased through the implementation of various types of models. Furthermore, the results of these studies are in line with the present study in that they indicate that the combination of different types of AI methods is a general trend in the area. For instance, the article by A.S. Yaraghi *et al*. (2022) reports that their approach achieved high levels of accuracy and efficiency. These results are generally in line with the findings of this study, though less pronounced in some contexts. For example, they found that their model was less effective in scenarios with high variability in system configuration. This is likely due to the different type of contexts in which those studies were conducted; A. Yaraghi *et al*. focused on contexts with stable continuous integration processes, while the current experiments used dynamic scenarios. Overall, the previous research generally supports the findings of the present study in that AI methods have the potential to be useful in the testing of software. However, the discrepancies between the results of those previous studies and the present study are likely the result of these different experimental contexts. Therefore, further research on this topic is required.

Conclusions

The results indicate that regardless of the method used to create the test case prioritisation sequences, the effect was consistent. The analysis of the APFD and APFDc values indicated that random approaches resulted in APFD values of 0.51 and 0.47, respectively, with a standard deviation of 0.042. The use of heuristic algorithms that focused upon code coverage resulted in APFD values of 0.62 and 0.58 ($\sigma = 0.031$). The population-based methods, such as genetic algorithms and PSO, achieved APFD values of 0.71-0.73 and 0.66-0.69 ($\sigma = 0.025$-0.028). The machine learning methods, such as Random Forest, achieved APFD values of 0.76 and 0.72 ($\sigma = 0.019$). The use of a hybrid method of machine learning and PSO achieved the best results, with APFD values of 0.81 and 0.77 ($\sigma = 0.014$). The calculation of the time required to execute these tests also indicated a benefit of these methods. For instance, the execution time for test suites was reduced from 23 minutes (for coverage-based heuristics) to 61 minutes (for the hybrid AI approach), representing a 17% and 45.2% reduction, respectively. The implementation of gradient boosting and RL-based Q-learning models led to a 37-minute (27.4%) and 54-minute (40%) reduction of test suite execution time, respectively. These results also indicate that the application of these methods to CI/CD processes will significantly reduce the length of those processes. For projects that compile nightly, there will be an average saving of 18-31% of process time, while those that use continuous delivery will save 12-24%.

The results from the aggregated analysis of these different methods lead to the understanding that the gains achieved by a method are not linearly related to the complexity of the algorithm. For instance, methods that focus on learning from the historical data in a test suite generally achieve the highest gains. Furthermore, even though the most complex methods in terms of computational efficiency achieved the highest values for APFD, their high required computational effort limits their applicability in certain environments. For instance, the analysis of the results of the different approaches to computational efficiency indicates that the maximal effect is achieved when the method balances the use of computational resources with the gains in APFD. As such, the implementation of a machine learning method that first performed the initial reduction of the search space to be solved by a heuristic method achieved high APFD and relative time gains of 0.81 and 0.77, 45.2%, respectively. This leads to suggestions for future research, such as the performance of additional experiments to investigate the effectiveness of these methods in different types of distributed systems and dynamic DevOps/DevSecOps environments.

Acknowledgements

None.

Funding

None.

Conflict of Interest

None.

References

[1] Amalfitano, D., Faralli, S., Hauck, J.C., Matalonga, S., & Distante, D. (2023). Artificial intelligence applied to software testing: A tertiary study. *ACM Computing Surveys*, 56(3), article number 58. doi: 10.1145/3616372.

[2] Anwar, R., & Bashir, M.B. (2023). A systematic literature review of AI-based software requirements prioritization techniques. *IEEE Access*, 11, 143815-143860. doi: 10.1109/ACCESS.2023.3343252.

[3] Baccour, E., Mhaisen, N., Abdellatif, A.A., Erbad, A., Mohamed, A., Hamdi, M., & Guizani, M. (2022). Pervasive AI for IoT applications: A survey on resource-efficient distributed artificial intelligence. *IEEE Communications Surveys & Tutorials*, 24(4), 2366-2418. doi: 10.1109/COMST.2022.3200740.

[4] Banala, S., Panyaram, S., & Selvakumar, P. (2025). Artificial intelligence in software testing. In P. Chelliah, R. Venkatesh, N. Natraj & R. Jeyaraj (Eds.), *Artificial intelligence for cloud-native software engineering* (pp. 237-262). London: IGI Global. doi: 10.4018/979-8-3693-9356-7.ch009.

[5] Birchler, C., Khatiri, S., Derakhshanfar, P., Panichella, S., & Panichella, A. (2023). Single and multi-objective test cases prioritization for self-driving cars in virtual environments. *ACM Transactions on Software Engineering and Methodology*, 32(2), article number 28. doi: 10.1145/3533818.

[6] Burachynskyi, A., & Shantyr, A. (2025). Overview of artificial intelligence application methods in software development. *Informatica*, 49(28), 59-72. doi: 10.31449/inf.v49i28.8694.

[7] Enemosah, A. (2025). Enhancing DevOps efficiency through AI-driven predictive models for continuous integration and deployment pipelines. *International Journal of Research Publication and Reviews*, 6(1), 871-887. doi: 10.55248/gengpi.6.0125.0229.

[8] Farah, J. (2021). Machine learning and AI in software testing automation: Enhancing performance in distributed network systems. *International Journal of Software Engineering and Knowledge Engineering*, 31(12), 45-60. doi: 10.13140/RG.2.2.27301.61925.

[9] Islam, M., Khan, F., Alam, S., & Hasan, M. (2023). Artificial intelligence in software testing: A systematic review. In *Proceedings of the TENCON 2023-2023 IEEE region 10 conference (TENCON)* (pp. 524-529). Chiang Mai: IEEE. doi: 10.1109/TENCON58879.2023.10322349.

[10] Khaliq, Z., Farooq, S.U., & Khan, D.A. (2022). Artificial intelligence in software testing: Impact, problems, challenges and prospect. *ArXiv*. doi: 10.48550/arXiv.2201.05371.

[11] Khrabatyn, R.I., Bandura, V.V., Zikraty, S.V., & Romanyshyn, T.L. (2024). Automatic generation of test cases based on system behaviour models using artificial intelligence to improve the quality of software products. *Scientific Bulletin of Ivano-Frankivsk National Technical University of Oil and Gas*, 2(57), 78-85. doi: 10.31471/1993-9965-2024-2(57)-78-85.

[12] Kumar, S. (2023). Reviewing software testing models and optimization techniques: An analysis of efficiency and advancement needs. *Journal of Computers, Mechanical and Management*, 2(1), 32-46. doi: 10.57159/gadl.jcmm.2.1.23041.

[13] Mahdieh, M., Mirian-Hosseinabadi, S.H., & Mahdieh, M. (2022). Test case prioritization using test case diversification and fault-proneness estimations. *Automated Software Engineering*, 29(2), article number 50. doi: 10.1007/s10515-022-00344-y.

[14] Mohapatra, P.S. (2025). *Intelligent assurance: Artificial intelligence-powered software testing in the modern development lifecycle*. London: Deep Science Publishing. doi: 10.70593/978-93-7185-046-9.

[15] Nama, P., Meka, N.H., & Pattanayak, N.S. (2021). Leveraging machine learning for intelligent test automation: Enhancing efficiency and accuracy in software testing. *International Journal of Science and Research Archive*, 3(1), 152-162. doi: 10.30574/ijsra.2021.3.1.0027.

[16] Okrushko, D., & Kashtalian, A. (2023). System of distribution and evaluation of tasks in the software development process. *Computer Systems and Information Technologies*, 2, 86-97. doi: 10.31891/csit-2023-2-12.

[17] Pan, R., Bagherzadeh, M., Ghaleb, T.A., & Briand, L. (2022). Test case selection and prioritization using machine learning: A systematic literature review. *Empirical Software Engineering*, 27(2), article number 29. doi: 10.1007/s10664-021-10066-6.

[18] Pandhare, H.V. (2025). Future of software test automation using AI/ML. *International Journal of Engineering and Computer Science*, 13(5), 27159-27182. doi: 10.18535/ijecs/v14i05.5139.

[19] Prity, F.S. (2023). Enhancing software testing efficiency through AI-guided test case prioritization: A systematic literature review. *Journal of Advances in Computational Intelligence Theory*, 5(3), 48-58. doi: 10.5281/ZENODO.8337098.

[20] Pyrih, Ya., Klymash, M., Pyrih, Yu., & Lavriv, O. (2023). Genetic algorithm as a tool for solving optimisation problems. *Information and Communication Technologies and Electronic Engineering*, 3(2), 95-107. doi: 10.23939/ictee2023.02.095.

[21] Sawant, P.D. (2024). Test case prioritization for regression testing using machine learning. In *Proceedings of the international conference on artificial intelligence testing* (pp. 152-153). Shanghai: IEEE. doi: 10.1109/AITest62860.2024.00027.

[22] Sharif, A., Marijan, D., & Liaaen, M. (2021). Deeporder: Deep learning for test case prioritization in continuous integration testing. In *2021 IEEE international conference on software maintenance and evolution* (pp. 525-534). Luxembourg: IEEE. doi: 10.1109/ICSME52107.2021.00053.

[23] Shi, T., Xiao, L., & Wu, K. (2020). Reinforcement learning based test case prioritization for enhancing the security of software. In *Proceedings of the 7th international conference on data science and advanced analytics* (pp. 663-672). Sydney: IEEE. doi: 10.1109/DSAA49011.2020.00076.

[24] Sugali, K. (2021). Software testing: Issues and challenges of artificial intelligence and machine learning. *International Journal of Artificial Intelligence and Applications*, 12(1), 101-112. doi: 10.5121/ijaia.2021.12107.

[25] Tahvili, S., & Hatvani, L. (2022). *Artificial intelligence methods for optimization of the software testing process: With practical examples and exercises*. London: Academic Press.

[26] Trifunova, A., Jakimovski, B., Chorbev, I., & Lameski, P. (2024). AI in software testing: Revolutionizing quality assurance. In *Proceedings of the 32nd telecommunications forum (TELFOR)* (pp. 1-4). Belgrade: IEEE. doi: 10.1109/TELFOR63250.2024.10819179.

[27] Vorochek, O., & Solovei, I. (2024). Research on artificial intelligence tools for automating the software testing process. *Bulletin of National Technical University "KhPI". Series: System Analysis, Control and Information Technologies*, 1(11), 58-64. doi: 10.20998/2079-0023.2024.01.09.

[28] Weiss, M., & Tonella, P. (2022). Simple techniques work surprisingly well for neural network test prioritization and active learning (replicability study). In *Proceedings of the 31st ACM SIGSOFT international symposium on software testing and analysis* (pp. 139-150). New York: ACM. doi: 10.1145/3533767.3534375.

[29] Yaraghi, A.S., Bagherzadeh, M., Kahani, N., & Briand, L.C. (2022). Scalable and accurate test case prioritization in continuous integration contexts. *IEEE Transactions on Software Engineering*, 49(4), 1615-1639. doi: 10.1109/TSE.2022.3184842.

Ефективність застосування штучного інтелекту для пріоритизації тестів у розподілених системах українського та міжнародного виробництва ПЗ

Андрій Задорожній
Магістр, старший інженер з розробки програмного забезпечення в тестуванні
CLTS Technologies Ltd. dba Aquanow
V6E 2M6, вул. Вест Пендер, 1095, м. Ванкувер, Канада
https://orcid.org/0009-0001-0307-8976

Анотація. Актуальність дослідження зумовлена зростанням складності розподілених Continuous Integration/Continuous Delivery (CI/CD)-систем та обмеженістю традиційних евристичних підходів до пріоритизації тестів, які не забезпечують достатньої стабільності та масштабованості в умовах великих тестових наборів і обмежених обчислювальних ресурсів. У зв'язку з цим постає потреба у пошуку більш адаптивних методів оптимізації процесу тестування. Метою дослідження було емпірично визначити особливості застосування методів штучного інтелекту (ШІ) для пріоритизації тестів у розподілених середовищах розробки програмного забезпечення та обґрунтувати практичний підхід до інтеграції ШІ у процеси автоматизованого тестування. Дослідження базувалося на порівняльному експериментальному аналізі інтелектуальних та гібридних методів пріоритизації тестів у розподілених системах із використанням метрик Average Percentage of Faults Detected (APFD) та Cost-cognizant Average Percentage of Faults Detected (APFDc). Результати дослідження показали перевагу інтелектуальних і гібридних підходів до пріоритизації тестів над традиційними евристиками в середовищах CI/CD. Випадкова пріоритизація демонструвала найнижчу ефективність із APFD близько 0,51, тоді як прості евристичні стратегії підвищували цей показник до приблизно 0,62. Популяційні методи забезпечували подальше зростання якості пріоритизації до рівня близько 0,72, а алгоритми машинного навчання – до близько 0,76, що підтверджує доцільність використання прогнозування дефектності для адаптивного впорядкування тестів. Найвищі результати було отримано для гібридних підходів, які поєднували машинне навчання з оптимізацією рою частинок: APFD досягав приблизно 0,81, а час виконання тестових наборів скорочувався майже на 45 %. Це свідчить про синергійний ефект інтеграції прогнозних моделей з оптимізаційними алгоритмами та підтверджує практичну доцільність гібридних методів для масштабованих розподілених CI/CD-середовищ. Результати дослідження можуть бути використані розробниками програмного забезпечення, командами забезпечення якості та інженерами для оптимізації процесів тестування у розподілених системах

Ключові слова: інтелектуальні алгоритми; машинне навчання; гібридні методи; оптимізаційні алгоритми; масштабованість; ефективність тестування

Journal homepage: https://itce.vn.ua/en

Information Technologies and Computer Engineering **Vol. 23, No. 1. 2026**

Article's History: Received: 14.10.2025 Revised: 09.01.2026
Accepted: 26.03.2026 Published: 20.04.2026

ISSN 1999-9941; e-ISSN 2078-6387
UDC 004.03
DOI: 10.31649/vitce/1.2026.140

A hybrid A-UKF-PINN digital twin architecture for real-time state estimation in Smart Grids

Vladimir Vychuzhanin*
Doctor of Technical Sciences, Professor
Odesa Polytechnic National University
65044, 1 Shevchenko Ave., Odesa, Ukraine
https://orcid.org/0000-0002-6302-1832

Alexey Vychuzhanin
PhD, Assistant
Odesa Polytechnic National University
65044, 1 Shevchenko Ave., Odesa, Ukraine
https://orcid.org/0000-0001-8779-2503

Abstract. The increasing variability, nonlinearity, and real-time operational requirements of Smart Grids (SGs) make static digital models insufficient for reliable state estimation and control of distributed assets such as Vehicle-to-Grid (V2G) storage systems. The purpose of the study was a formal and model-based substantiation of the advantages of dynamic digital twins (DTs) over static data model (DM) in real-time lithium-ion storage system condition assessment tasks. To achieve this, a hybrid adaptive unscented Kalman filter – physics-informed neural network (A-UKF-PINN) architecture was proposed, combining an A-UKF (Adaptive Unscented Kalman Filter), which provided robust state estimation in the presence of noise and uncertainty, with a physics-informed PINN (Physics-Informed Neural Network) model that considers the dynamics and nonlinear processes of the battery cell. The originality of the study lies in the integration of these components into a single model that supports bidirectional synchronisation, which improves forecast stability and significantly reduces desynchronisation between the model and the physical object in SG conditions. Simulation validation was carried out on V2G operating cycles with modelled Phasor Measurement Unit / Internet of Things sensor noise. The obtained Root Mean Square Error (RMSE) of 0.87% demonstrated a 44% accuracy improvement compared to a traditional DM (ECM (equivalent circuit models) + UKF, RMSE 1.98%) and a 56% improvement relative to the baseline digital twin (pure PINN). The architectural assessment confirmed the necessity of using a hierarchical Edge-Cloud platform that ensures optimal distribution of computational workloads: PINN training in the cloud environment and high-frequency state estimation at the edge. The proposed architecture forms the basis for scalable dynamic DTs in SG, helps to reduce operational risks, supports the implementation of proactive maintenance strategies, and increases the efficiency of the energy infrastructure life cycle

Keywords: hybrid modelling; Physics-Informed Neural Networks; Unscented Kalman Filter; functional superiority; Edge-Cloud

Introduction

Intelligent power grids (smart grid – SG) represent a cyber-physical infrastructure that provides real-time management of distributed energy resources under high variability of energy and data flows. In modern SGs, the share of distributed generation is continuously increasing, heterogeneous resources (including Vehicle-to-Grid, V2G) are being integrated, and control decisions must be made under strict latency and reliability requirements. Under these conditions, ensuring the consistency of digital processes with the dynamic physical state of the network becomes essential. The SG architecture is characterised by nonlinearity and multiple interconnected control loops. Load, generation, consumption modes, and data volumes change in real time, and any disruption in coordination

Suggested Citation:
Vychuzhanin, V., & Vychuzhanin, A. (2026). A hybrid A-UKF-PINN digital twin architecture for real-time state estimation in Smart Grids. *Information Technologies and Computer Engineering*, 23(1), 140-152. doi: 10.31649/vitce/1.2026.140

*Corresponding author

between the physical and digital parts of the system can lead to cascading risks. Such characteristics require digital representations capable not only of capturing the state of sources and storage systems but also of adapting to rapid changes in operating modes.

Contemporary research in the field of digital modelling and digital twins (DTs) for energy systems demonstrated the development of approaches that combine physics-informed methods, machine learning, and state filtering (Vychuzhanin & Vychuzhanin, 2025). A significant body of work highlighted the role of the digital twin as a key tool for ensuring stable, secure, and reliable SG operation. A comprehensive review of digital twins in energy systems was presented by R. Alharbey *et al*. (2024), where it was shown that DTs platforms are becoming the foundation for enhancing resilience, improving efficiency, and integrating renewable resources into the SG. The researchers emphasised that static DMs do not provide the required dynamism and synchronisation with the physical infrastructure, creating a need for hybrid and physics-informed methods. A systematic study by O. Das *et al*. (2024) demonstrated the potential of DTs when combined with machine learning and forecasting methods. The researchers highlighted the necessity of integrating physical models with high-density data generated by phasor measurement unit (PMU) and IoT nodes, since purely data-driven approaches do not ensure stability under changing operating conditions.

Issues of resilience, scalability, and applied challenges of the SG in the context of digital twins were examined in detail by N. Mchirgui *et al*. (2024), who emphasised that the key problem remains the lack of reliable bidirectional synchronisation between the digital and physical layers, and insufficient accuracy in modelling nonlinear processes, including the dynamics of storage systems and V2G assets. In the domain of lithium-ion battery state estimation, considerable attention was given to models based on equivalent circuit models (ECMs), extended through Kalman filtering methods. X. Lin *et al*. (2021) showed that an adaptive Unscented Kalman Filter (UKF) can increase the accuracy of state-of-charge (SoC) estimation provided that ECM parameters are highly accurate. However, in heavy dynamic modes, ECM accuracy decreases due to limited physical expressiveness. Conventional UKF-based approaches for state-of-charge estimation have been widely studied in the context of battery management systems. For example, hybrid methods combining UKF with neural network models have been shown to reduce estimation error and improve robustness across varying temperatures and dynamic operating conditions (Zeng *et al*., 2023). Additional improvements were proposed by H. Bouchareb *et al*. (2024), who showed the effectiveness of joint parameter and SoC estimation using Joint sigma-point Kalman filtering. Their study confirmed that Kalman filters can compensate for noise and parameter drift but remain sensitive to inaccuracies in the physical model.

On the other hand, the field of PINNs (Physics-Informed Neural Network) has shown notable progress. The study by F. Wang *et al*. (2024) has shown that PINN-based models can provide high accuracy in battery degradation forecasting; however, the stability of the solution strongly depends on the correctness of physical constraints, and scalability is limited by the computational cost of training. Hybrid approaches combining PINNs and filtering methods are also actively evolving. L. Lyu *et al*. (2024) also presented a hybrid approach (LSTM (Long Short-Term Memory) + UKF), showing enhanced accuracy considering battery degradation. Both studies emphasised the need to combine data-driven approaches capable of correcting noise with methods that preserve physical consistency. However, none of the existing studies proposed a fully integrated architecture that simultaneously: embeds physical constraints via PINN; performs adaptive state estimation using UKF; ensures robustness to noise; is implementable as a DT dynamically synchronised with SG nodes in real time.

A systematic analysis of existing research showed that current approaches do not simultaneously provide: physical consistency of models (PINN and other physics-based methods); robustness to noise and parametric uncertainty (the UKF/AEKF (Adaptive Extended Kalman Filter)/AUKF (Adaptive Unscented Kalman Filter) family); the necessary dynamic accuracy for real-time tasks under V2G and SG conditions. Despite the progress noted in these studies – including adaptive filtering methods for SoC estimation none of the known models comprehensively addresses the problem of simultaneous physical consistency, adaptive filtering, and stable forecasting in the presence of nonlinear dynamics, measurement noise, and a wide range of operating conditions. The synthesis of identified limitations allows formulating the unresolved scientific and technical gap underlying this research: the absence of a unified hybrid architecture capable of reliably PINN and robust adaptive state filtering (A-UKF) within an end-to-end DT for the SG. Based on this, the purpose of this study was the algorithmic and experimental substantiation of the functional superiority of the DT over traditional DMs, namely the development and validation of the hybrid A-UKF-PINN architecture and the examination of its effectiveness in the task of SoC prediction under V2G operating conditions. The central task of this study was the quantitative and architectural assessment of the effectiveness of DT-based solutions relative to traditional DMs under dynamic SG operating conditions, using hybrid algorithms and real-time architectures.

Materials and Methods

The research methodology was based on the application of contemporary information technology approaches for modelling and assessing the state of complex energy systems. Within the framework of the study, methods for estimating SoC of lithium-ion energy storage systems operating in a smart grid environment were analysed, considering nonlinear electro-thermal dynamics and stochastic

measurement noise generated by PMU and IoT devices. A hybrid DT architecture integrating PINN and A-UKF was developed to ensure physically consistent prediction and adaptive real-time state correction. The architecture forms a closed synchronisation loop between the physical battery system and its digital representation (Fig. 1).

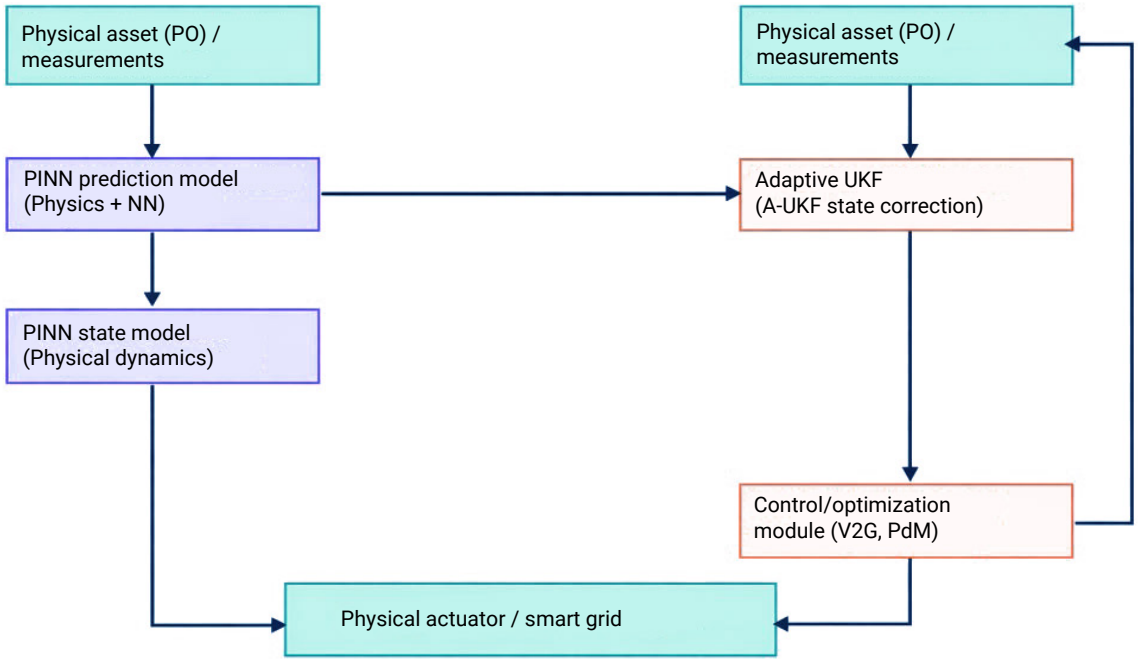

Figure 1. Hybrid A-UKF-PINN architecture

Source: created by the authors

The computational cycle consists of two sequential stages executed at each time step t: (1) physics-informed a priori state prediction and (2) adaptive a posteriori correction using innovation statistics. The PINN was implemented as a fully connected feedforward neural network embedding electro-thermal governing equations into the loss function. The network architecture was defined as follows: input dimension – 4 (terminal voltage V_t, current I_t, temperature T_t, previous state estimate SoC_{t-1}); 4 hidden layers with 64 neurons each; activation function – hyperbolic tangent (tanh); output layer – one neuron (predicted SoC) with linear activation. Xavier (Glorot) uniform initialisation was applied to ensure stable gradient propagation. Training parameters: Adam optimiser; initial learning rate 10^{-3}; batch size 64; maximum 500 epochs; early stopping (25 validation epochs without improvement); L2-regularisation coefficient 10^{-2}.

The composite loss function combines data fidelity and physical consistency:

$$L = L_{data} + \lambda_1 \cdot L_{PDE(V)} + \lambda_2 \cdot L_{PDE(T)}, \quad (1)$$

where L_{data} – mean squared error between predicted and measured SoC; $L_{PDE(V)}$ – residual of the voltage conservation equation; $L_{PDE(T)}$ – residual of the thermal balance equation; $\lambda_1 = 1.0, \lambda_2 = 0.5$ –empirically selected weighting coefficients.

The overall loss function $L(\theta)$ is formulated as a weighted sum of the data loss L_{data} and the physical-law residual $L_{physics}$ (Gao *et al.*, 2025):

$$L(\theta) = L_{data} + \beta \cdot L_{physics}, \quad (2)$$

where θ denotes network parameters and β balances data and physical-law contributions.

The data loss term is defined as:

$$L_{data} = \frac{1}{N_d} \sum_{i=1}^{N_d} (N(x_i; \theta) - y_i)^2, \quad (3)$$

where N_d – total number of data points (measurements); $N(x_i; \theta)$ – value predicted by the neural network for the i-th input vector; y_i – true, actual value measured by sensors (e.g., SoC); x_i – input vector representing the system state.

The physics-based residual term is:

$$L_{physics} = \frac{1}{N_f} \sum_{j=1}^{N_f} (F(N(x_j; \theta)))^2, \quad (4)$$

where N_f – total number of test points selected for verifying the physical laws; F – operator describing the physical dynamics of the system; $N(x_j; \theta)$ – value predicted by the neural network for the j-th test point; x_j – vector of coordinates or parameters of the j-th test point (residual points).

Training was performed offline in a cloud environment using GPU acceleration. The trained model was deployed for online inference within the DT framework. To ensure continuous synchronisation between the digital twin and the physical battery system, dynamic state estimation is performed using the UKF. The predicted state x_t^- and the predicted covariance matrix P_t^- are computed based on a set of sigma points $\hat{X}_t^i$ (Julier & Uhlmann, 1997):

$$x_t^- = \sum_{i=0}^{2n} W_i^m \hat{X}_t^i,$$
$$P_t^- = \sum_{i=0}^{2n} W_i^c [(\hat{X}_t^i - x_t^-)(\hat{X}_t^i - x_t^-)^T] + \theta_{t-1}, \quad (5)$$

where x_t^- – predicted system state vector at time t; P_t^- – predicted state error covariance matrix; $\hat{X}_t^i$ – i-th sigma point projected forward using the nonlinear state transition function; W_i^m – weighting coefficient used to average the sigma points when computing the state prediction; W_i^c – weighting coefficient used to compute the predicted covariance matrix; n – dimensionality of the state vector; θ_{t-1}, – process noise covariance matrix (errors caused by the system's own dynamics).

The innovation (measurement residual) is defined as:

$$v_t = y_t - \hat{y}_t, \quad (6)$$

where y_t – measurement vector (voltage, current, temperature); $\hat{y}_t$ – predicted measurement obtained from sigma-point projection.

The innovation covariance matrix is:

$$S_t = P_{yy,t} + R_t, \quad (7)$$

where $P_{yy,t}$ is the predicted measurement covariance; R_t is the measurement noise covariance.

The Kalman gain is computed as:

$$K_t = C_t \cdot S_t^{-1},$$
$$x_t = x_t^- + K_t[y_t - \mu_t], \quad (8)$$

where K_t – Kalman gain, which determines the degree of trust in the new measurement; C_t – cross-covariance matrix between the predicted state and the predicted measurements; S_t^{-1} – inverse covariance matrix of the measurement error; x_t^- – predicted state vector of the complex technical system (CTS); y_t – actual measurement vector obtained from PMU sensors; μ_t – predicted measurement vector (the expected value of y_t).

To ensure robustness under non-stationary PMU noise, covariance adaptation is performed using innovation-based updating:

process noise covariance update:

$$Q_{t+1} = (1-\alpha) \cdot Q_t + \alpha \cdot K_t \cdot v_t \cdot v_t^T \cdot K_t^T; \quad (9)$$

measurement noise covariance update:

$$R_{t+1} = (1-\beta) \cdot R_t + \beta \cdot v_t \cdot v_t^T, \quad (10)$$

where K_t – Kalman gain; $\alpha = 0.01$ – process adaptation rate; $\beta = 0.02$ – measurement adaptation rate.

At each time step t, the hybrid digital twin executes:

1. PINN-based a priori state prediction:

$$\hat{v}_{t|t-1} = f_{PINN}(x_{t-1}, u_t). \quad (11)$$

2. Sigma-point generation and propagation (UKF prediction phase).

3. Innovation computation:

$$v_t = y_t - \hat{y}_t. \quad (12)$$

4. A-UKF correction step:

$$x_{t|t} = \hat{x}_{t|t-1} + K_{tv_t}. \quad (13)$$

The algorithmic logic of the hybrid solution was represented in the form of the following pseudocode, reflecting the computation cycle and its interdependencies. This sequential prediction-correction mechanism ensured real-time knowledge equivalence between the physical battery system and its digital representation:

```
Initialise θ, x0
for each timestep t:
 zt ← measurements(PMU)
 x_pred ← PINN(θ, xt−1)
 x_upd ← AUKF(x_pred, zt)
 Update PINN loss with physical constraints
 xt ← x_upd
end
return xt
```

Validation was performed in a virtual Hardware-in-the-Loop environment using a high-fidelity 50 Ah lithium-ion battery model. The simulation included 50 dynamic V2G charge-discharge cycles (90-110 min each) with random load profiles, temperature variations, and PMU noise (SNR 25-35 dB). The dataset was split into training/validation/test subsets (70/15/15). All experiments were repeated three times with fixed random seed (42). Performance metrics (RMSE, MAE) were averaged, standard deviations computed, and statistical significance evaluated using the Wilcoxon signed-rank test ($p < 0.05$). The computational implementation followed a Cloud-Edge paradigm: offline PINN training in the cloud and real-time A-UKF correction at the edge level (sampling frequency 1-10 Hz). This architecture reduced latency and ensures scalability under distributed SG operation.

Results and Discussion

Formalisation and differentiation of the digital model and the DT. To demonstrate the systemic complexity and the role of digital representation, Figure 2 presents a conceptual SG architecture. It highlights four interconnected levels: distributed generation (wind and solar), high-voltage transmission (substations), distribution networks, and hybrid consumers/storage systems (V2G and industrial loads). The solid line represents physical power flows, while the dashed line denotes bidirectional information exchange. The presence of high-frequency telemetry streams (PMU/IoT), SCADA control actions, and bidirectional energy interaction with V2G forms a closed cyber-physical loop that requires continuous state estimation and predictive analytics across the entire network.

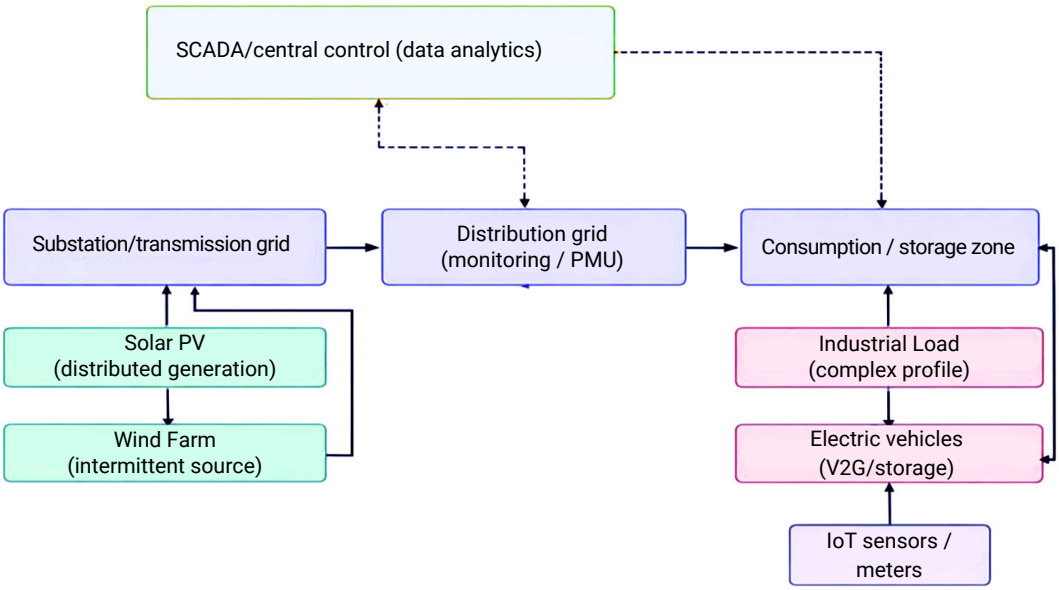

Figure 2. SG conceptual architecture

Source: created by the authors

An analysis of the structure shown in Figure 2 demonstrates that conventional modelling approaches focused on static or quasi-static analysis cannot be adapted to operating conditions characterised by high penetration of distributed generation, bidirectional V2G energy exchange, high-frequency PMU/IoT telemetry (1-10 Hz), stochastic load variability, and nonlinear electro-thermal battery dynamics. Under these conditions, static models capture system behaviour only in isolated operating modes and are unable to provide consistent state estimation during rapid transitions, load fluctuations, and temperature-induced parameter drift. This is conditioned by the growing variability of generation, the increasing spatial heterogeneity of operating conditions, and the exponential growth in data volumes coming from distributed measurement devices. Conventional DMs, used primarily at the design stage, operate with historical or static data and rely on a one-way "data → model" relationship. Such an approach does not provide dynamic synchronisation with the physical object and cannot perform real-time state estimation in the presence of noise, nonlinear effects, and uncertainties characteristic of SG and V2G storage systems. This circumstance makes them insufficient for the operation of modern distributed energy systems. In contrast to DMs, a DT is a dynamic cyber-physical system in which the digital representation is continuously updated in accordance with the current state of the physical object and provides bidirectional integration of analysis and control. A DT combines state estimation, forecasting, adaptive behaviour, and real-time data integration, making it a functionally more suitable tool for SG.

The transition from using a DM to the DT paradigm is a necessary condition for effective management of cyber-physical systems such as the SG. The limitations of DMs, which operate on static data and are intended for simulation at the design stage, make them unsuitable for tasks of dynamic optimisation and predictive maintenance in real time. The fundamental difference between a DM and a DT lies in the mechanism of interaction with the physical object (PO) and in their functional purpose. A DM is a static or quasi-static representation of a system, whereas a DT is a living, dynamically synchronised cyber-physical system. A comparative analysis of the key characteristics of DMs and DTs is presented in Table 1.

Table 1. Comparative characteristics of the DM and the DT

Characteristic	Digital model	Digital twin
Connection with PO	One-way or absent. Updated manually	Continuous bidirectional connection (feedback loop)
Synchronisation	Static, quasi-static	Dynamic, real time
Purpose	Design, planning, pre-operation simulation	Operation management, optimisation, RUL and state prediction during operation
Data sources	Historical, synthetic, experimental	Real-time streams (PMU/IoT), historical, heterogeneous
Critical element	Modelling quality	Maintaining knowledge equivalence between the model and the PO

Source: created by the authors

At the architectural level, the DT is defined as a cyber-physical system (CPS) requiring a standardised approach. The series of international standards ISO 23247-1:2021 (2021) proposes a reference four-domain structure that formalises the interaction between the physical and virtual worlds, ensuring interoperability and scalability, which in current case formalises the interaction between the physical SG and its digital representation. This structure, shown in Figure 3, constitutes a technical imperative for implementing DTs in cyber-physical systems.

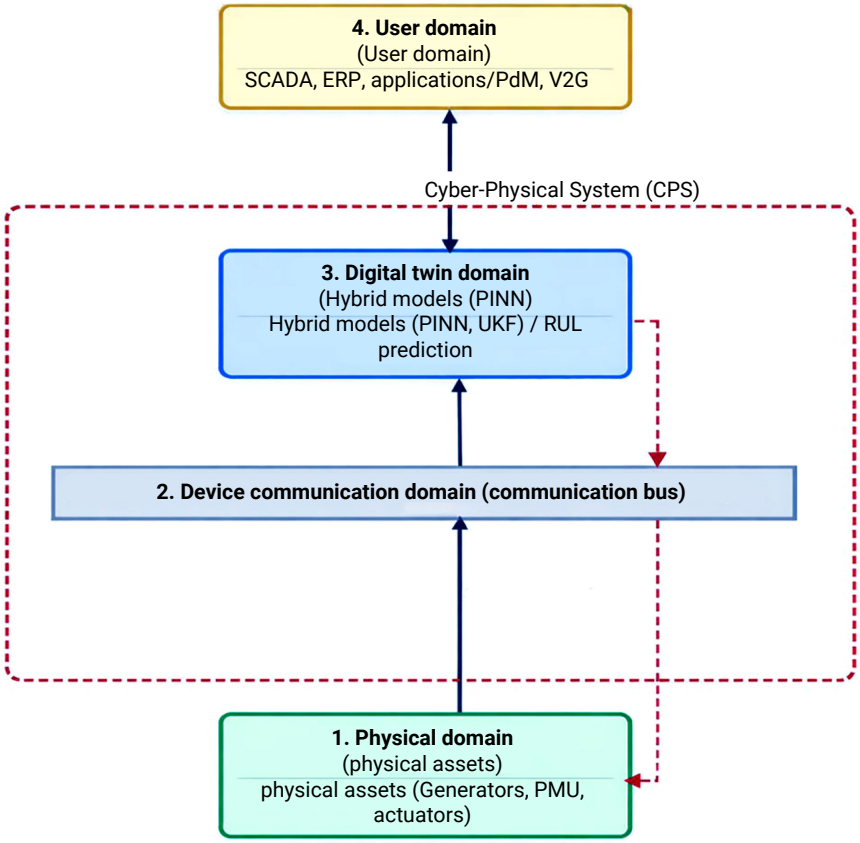

Figure 3. Reference four-domain DT architecture
Source: created by the authors based on ISO 23247-2:2021 (2021)

The conceptual domain structure (ISO 23247-2:2021 (2021)) includes:

1) observable manufacturing domain (physical SG domain): contains physical objects (POs) such as generators, substations, PMUs, and electric vehicles;

2) device communication domain: the interface domain containing sensors and actuators. It provides the critically important bidirectional connection for synchronisation and control actions;

3) DT domain: contains the digital representations of POs, behavioural modelling and forecasting algorithms (including hybrid models such as A-UKF-PINN);

4) user domain: the top level, including SCADA, ERP systems, and applications for DT services (PdM, V2G optimisation).

This four-domain structure shows that the core of the DT lies in the logical integration of the DT domain (3) and the device communication domain (2) into a single CPS. The architectural rigour presented in ISO 23247-1:2021 (2021) provides the necessary context for scaling and integrating this advanced algorithms, confirming that the successful operation of the DT in the SG depends on adherence to this standardised logic. Thus, the application of this four-domain structure provides the necessary foundation for implementing scalable and efficient digital twins in smart grid environments, ensuring their interoperability and adaptation to rapidly changing technological requirements.

Algorithmic foundation and hybrid modelling for dynamic synchronisation in smart grids. The architectural implementation of the DT requires the use of a mathematical framework capable of ensuring predictive accuracy and dynamic synchronisation under SG uncertainty. Hybrid models with physical constraints (PINN). To ensure high reliability of forecasting, DT models of the SG must comply with fundamental physical laws while adapting to real-time data. This is achieved through the use of PINNs. PINNs embed known physical laws into the neural network loss function, ensuring compliance with physical constraints during training. Figure 4 illustrates the operating principle of a PINN, showing how the residual from the physical model is fed back into the overall loss function to correct the neural network weights.

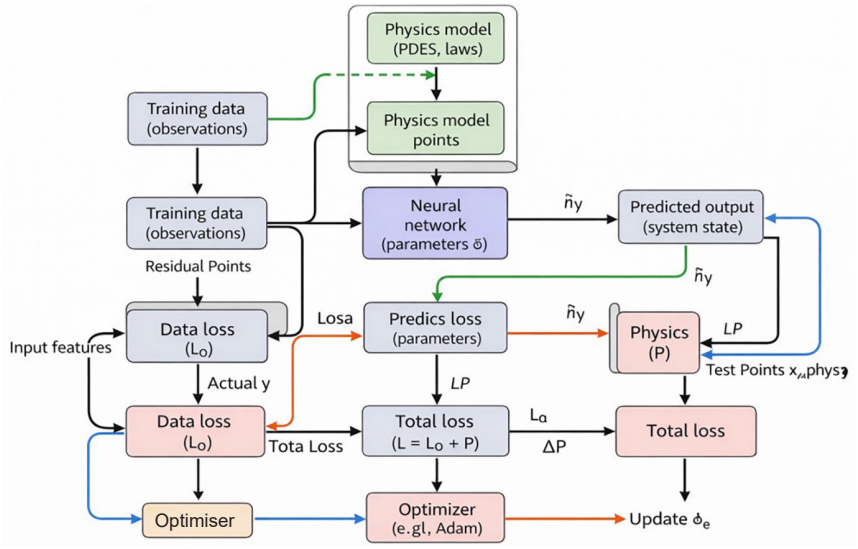

Figure 4. Physics-informed neural network

Source: M. Raissi *et al.* (2019)

To evaluate the practical effectiveness of the proposed hybrid digital twin architecture, a structured experimental comparison was conducted against representative baseline approaches. The objective of this evaluation was to quantify differences in SoC estimation accuracy, robustness, and computational efficiency under dynamic V2G operating conditions. The comparison included (1) a conventional digital model based on an equivalent circuit model with Kalman filtering (ECM + KF), (2) a baseline DT implementation using a pure PINN without adaptive filtering, and (3) the proposed A-UKF-PINN hybrid architecture. The quantitative results of this comparative assessment are summarised in Table 2.

Table 2. Comparative analysis of SoC prediction accuracy

Model	Concept	(RMSE) %	Maximum error (MAE), %	Execution time (per step), ms
ECM + Kalman filter	DM / traditional	1.98	3.55	2.1
Pure PINN	Baseline DT approach	2.54	4.11	15.2
A-UKF-PINN (proposed HM)	Advanced DT	0.87	1.55	16.5

Source: created by the authors

The proposed (Table 2**)** adaptive hybrid A-UKF-PINN model demonstrated an RMSE reduction of more than 56% compared to the pure PINN model and 44% compared to the traditional ECM + KF model. This is a critically important improvement, since a SoC prediction accuracy below 1% is a standard requirement for reliable V2G operations. A detailed comparison of the dynamic response of all three models under charge/discharge cycle conditions, confirming the minimal deviation of the hybrid model from the actual state (true SoC), is presented in Figure 5 (equivalent circuit model).

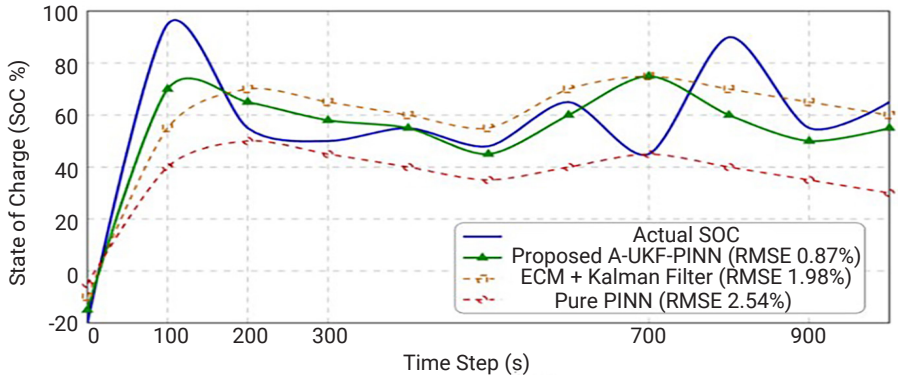

Figure 5. Comparison of SoC prediction accuracy

Source: created by the authors

As follows from Figure 5, the behaviour of the conventional models (ECM + Kalman Filter and Pure PINN, representing the DM and the baseline DT approach) exhibits a significant lag and excessive smoothing, which prevents tracking fast nonlinear changes during the V2G cycle. In contrast, the response of the proposed A-UKF-PINN almost coincides with the actual state (Actual SoC), achieving a minimal error of 0.87% RMSE, which constitutes evidence of the superiority of the hybrid DT approach in the dynamic synchronisation of a critical cyber-technical system asset. The obtained result (RMSE 1.98% for the DM/ECM) experimentally proves the functional limitations of digital models in solving dynamic state estimation tasks for cyber-technical systems.

The obtained results demonstrate improved performance of the proposed A-UKF-PINN hybrid architecture compared to the ECM+KF baseline and the pure PINN implementation in managing critical assets of cyber-technical systems:

1. Comparative performance improvement: The hybrid A-UKF-PINN architecture achieved a minimum RMSE of 0.87%, corresponding to a 44-56% error reduction relative to the ECM+KF and pure PINN baselines under the considered operating conditions. This improvement is associated with the combined effect of dynamic state correction and physics-informed regularisation.

2. Architectural contribution to SG integration: The integration of PINNs enabled physically consistent modelling of nonlinear electro-thermal dynamics, while the adaptive filtering mechanism enhanced robustness to stochastic noise in PMU/IoT measurement streams.

3. Real-time state consistency: The achieved estimation accuracy (RMSE 0.87%) indicates the feasibility of maintaining a consistent digital representation of a critical asset in real time within the examined Smart Grid scenario.

These experimentally validated results indicate that hybrid DT architectures incorporating adaptive filtering and physics-informed learning can provide measurable performance benefits compared to the selected baseline implementations in dynamic SG environments. These findings highlight the potential of the A-UKF-PINN hybrid architecture to revolutionise the management of critical cyber-technical assets in smart grids, offering significant improvements in both accuracy and robustness for real-time state estimation under dynamic conditions.

While Table 2 summarises the nominal accuracy and computational cost of the considered SoC estimation models, practical SG and vehicle-to-grid applications require digital twins to operate reliably under previously unseen operating conditions. The generalisation capability of the proposed A-UKF-PINN hybrid digital twin is evaluated under out-of-distribution scenarios, including variations in battery cell chemistry and temperature regimes. To assess model transferability across different electrochemical characteristics, the baseline lithium-ion NMC cell used in the nominal experiments was replaced with a lithium iron phosphate (LFP) cell. The LFP chemistry exhibits distinct open-circuit voltage behaviour, internal resistance dynamics, and diffusion properties, which typically degrade the performance of equivalent circuit models calibrated for a specific cell type. In this experiment, the physics-informed neural network was not retrained for the new cell configuration. Instead, only the adaptive noise covariance matrices of the unscented Kalman filter were updated online, preserving the original digital twin structure. This setup reflects a realistic deployment scenario in which frequent model re-identification is undesirable or economically infeasible. In addition to cell variation, a structured temperature-based cross-validation was conducted to evaluate extrapolation performance under thermal conditions not represented during training. Unlike random data splits, the training and testing datasets were separated by temperature intervals, enforcing physically meaningful distribution shifts. Cold-start and elevated-temperature scenarios were considered, as these conditions are known to introduce strong nonlinearities and parameter drift in battery models. This evaluation framework directly tests the robustness of the digital twin under realistic environmental variability encountered in large-scale SG cyber-technical systems. ΔRMSE values are computed relative to the nominal RMSE reported in Table 3.

Table 3. Relative degradation of SoC estimation accuracy under cell type variation and temperature cross-validation

Model	ΔRMSE (New cell), %	ΔRMSE (Cold CV), %	ΔRMSE (Hot CV), %
ECM + Kalman filter	+57.6	+94.4	+72.2
Pure PINN	+13.8	unstable	+18.9
A-UKF-PINN (proposed HM)	+20.7	+39.1	+35.6

Source: created by the authors

As shown in Table 3, all baseline approaches experience a pronounced degradation in estimation accuracy under out-of-distribution conditions. The ECM-based method is particularly sensitive to both temperature shifts and cell chemistry variation, with RMSE nearly doubling in cold-start scenarios. The pure PINN model exhibits limited robustness and becomes unstable under low-temperature conditions due to the absence of online state correction. In contrast, the proposed A-UKF-PINN digital twin demonstrates substantially lower sensitivity to changes in operating conditions. Although a moderate increase in RMSE is observed, the degradation remains controlled across all evaluated scenarios, confirming the robustness and transferability of the hybrid physics-informed approach.

The obtained results highlight the advantage of combining physics-informed learning with adaptive state

estimation in closed-loop digital twin architecture. The physics-based constraints embedded in the PINN promote physically consistent extrapolation, while the adaptive UKF compensates for unmodelled dynamics and measurement uncertainty in real time. This synergistic interaction enables the digital twin to maintain synchronisation with the physical asset without requiring repeated offline recalibration. Unlike static digital models and purely data-driven estimators, the proposed approach maintains reliable performance across heterogeneous battery configurations and environmental conditions, which is a critical requirement for scalable SG and V2G deployments.

Economic efficiency of DT implementation in SG cyber-technical systems. The economic impact of DT implementation in SG systems is influenced not only by the achieved algorithmic accuracy but also by the potential to reduce operational risks, improve asset reliability, and lower total life-cycle costs. Unlike conventional DMs, which are primarily oriented toward design-stage analysis and typically do not provide dynamic state estimation, a DT supports a closed-loop "observation – prediction – control" cycle. It can therefore create additional economic value in operational contexts.

The high accuracy of dynamic state estimation achieved by the developed A-UKF-PINN model (RMSE of 0.87% in SoC prediction) creates the prerequisites for a transition from scheduled periodic maintenance toward more proactive asset management strategies. In practical deployments, this capability may: contribute to a reduction in unplanned downtime of V2G storage systems, as improved SoC and RUL estimation can decrease the likelihood of unexpected failures; support lower operation and maintenance (O&M) costs through earlier detection of adverse operating conditions and reduced reliance on reactive maintenance; enhance the economic efficiency of V2G operation by enabling more accurate forecasting of SoC and available power, which may help to reduce imbalance penalties and improve participation in primary and secondary regulation services. In conventional DMs, these effects are unattainable due to the lack of dynamic feedback, the limitations of equivalent electrical circuit models, and the inability to correct predictions under noisy PMU/IoT data streams. The economic feasibility of DT implementation was determined by the combination of two factors: system complexity and the cost of failure consequences. These parameters are systematised in the feasibility matrix.

As follows from Figure 6, the SG is located in the high feasibility ("High Feasibility") zone. This is conditioned by: the high structural and operational complexity of the SG; the critical consequences of failures, including grid overloads, power balance violations, and downtime of V2G assets; the high sensitivity of the SG to state forecasting errors. The accuracy achieved by the hybrid A-UKF-PINN model (RMSE 0.87%) provides the required level of confidence in the forecasts and thereby makes implementation economically justified. Economic feasibility is also confirmed by a comparison of the total life-cycle cost of assets when using: conventional digital models (low initial costs, high operational risks); DTs (higher initial investments, but a significant reduction in subsequent costs).

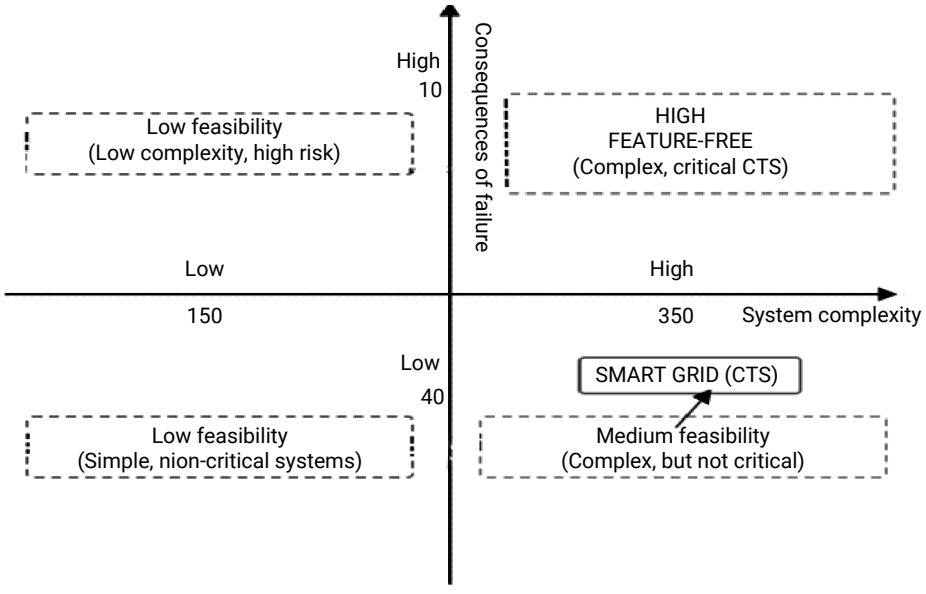

Figure 6. DT Implementation feasibility matrix

Source: created by the authors

The numerical values presented in this section (reduction of unplanned downtime by up to 20%, reduction of O&M costs by up to 18%, and a relative decrease in total cost of ownership) are based on a scenario-based techno-economic analysis typical for SG assets. The assessment was performed using a comparative life-cycle analysis methodology that considered the failure rates of storage systems under V2G operation, the cost

of unplanned downtime, average reactive maintenance costs, and the effects of transitioning to proactive maintenance based on a high-accuracy DSE model (A-UKF-PINN). The resulting value ranges are consistent with results reported in the literature on Smart Grids and energy storage management systems (Kabir *et al*, 2024) and serve to illustrate the expected class of improvements resulting from the implementation of digital twins. Detailed intermediate calculations are not provided, since the main focus of the study is on the algorithmic and architectural components of the DT, while the economic estimates serve as auxiliary confirmation of their practical significance.

Figure 7 demonstrates that the main financial effect of implementing the DT is formed at the operational stage, where: failure risk is reduced due to accurate state prediction; O&M costs are stabilised; the number of emergency situations and the associated expenses decreases; the efficiency of participation in market mechanisms increases. As a result, despite the higher initial deployment cost (PINN training, deployment of A-UKF on edge nodes, integration with PMU/IoT), the overall TCO is lower compared to the DM.

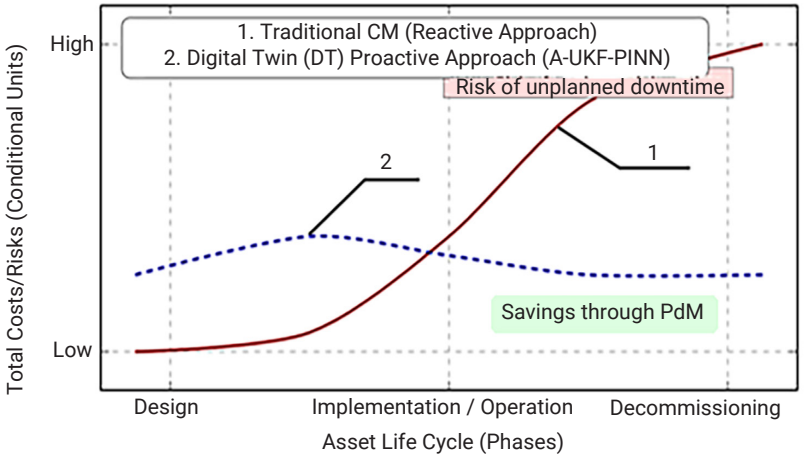

Figure 7. Total cost of ownership (TCO) and risk comparison

Source: created by the authors

The obtained results showed that achieving the economic effect requires compliance with specific architectural conditions: a dual-loop Edge-Cloud structure, where the Cloud performs resource-intensive PINN training on large datasets, while the Edge ensures minimal latency for real-time operation of A-UKF; standardisation of interfaces and protocols, which is especially important in a heterogeneous Smart Grid environment. Implementation in accordance with ISO 23247-1:2021 (2021) ensures interoperability between PMUs, IoT sensors, and control nodes, which is a necessary condition for DT scalability. Architectural validation confirmed that it is precisely the hybrid A-UKF-PINN model that satisfies the latency and stability requirements necessary to achieve the high economic efficiency presented in Figures 6 and 7. The economic analysis confirmed that the transition from static digital models to a dynamic digital twin is: financially justified (TCO reduction, risk mitigation, increased V2G efficiency); operationally necessary (support for PdM, improved reliability); architecturally feasible (subject to compliance with the Edge-Cloud paradigm and ISO 23247-2:2021 (2021) standards). Thus, the algorithmic superiority of A-UKF-PINN is translated into a direct economic effect, making DT implementation strategically justified for critical Smart Grid systems.

For an objective assessment of the achieved results, it is useful to compare the proposed A-UKF-PINN hybrid with a number of other contemporary approaches typical of the field of SoC estimation and dynamic state assessment. L. Hu *et al*. (2022) demonstrated that AUKF combined with a classical equivalent circuit model ensures stable convergence and acceptable accuracy over a wide range of operating conditions; however, dependence on the ECM limits the adequate description of nonlinear thermal behaviour and fast V2G modes. Proposed A-UKF-PINN combines a physics-informed model with adaptive correction, resulting in noticeably lower RMSE and better robustness in noisy, highly dynamic scenarios. S. Hosseininasab *et al*. (2023) proposed a reduced-order model combined with an Adaptive Dual UKF, achieving a balance between accuracy and computational efficiency; the reduced model simplifies computation but loses part of the physical detail required under extreme operating regimes. In contrast, the PINN in current architecture preserves the physical consistency of the model, while A-UKF provides prompt correction–together improving transferability to real SG/V2G conditions.

Current approach reduces this dependence, since PINN directly models the physics, while A-UKF performs real-time correction. Y. Wei (2024) clearly showed the benefit of combining an advanced physical model (fractional-order) with two filters (FOSRCKF + AMIUKF) with parameter exchange: this improves SoC accuracy and terminal prediction compared to integer-order models, but requires complex offline identification and mutual filter tuning. A-UKF-PINN achieves comparable or better robustness with more direct

physics integration (via PINN) and simplifies online tuning through covariance adaptation in A-UKF. Z. Wang *et al*. (2024) proposed an adaptive extended sliding innovation filter and demonstrated improved SoC estimation stability in the presence of disturbances; however, like most enhanced filters, the method is still based on a circuit model and is sensitive to its adequacy. In the HIL tests, the PINN + A-UKF combination showed more uniform accuracy and higher robustness to PMU/IoT noise compared to the AESIF approach.

J. Guo *et al*. (2023) developed UKF-based approaches for SoC (2-RC and others) and demonstrated improvements over classical KF/EKF; nevertheless, estimation quality is largely determined by the accuracy of the circuit model parameters. A-UKF-PINN reduces this dependence: PINN provides a rich physical approximation, while the adaptive UKF performs online correction of residual model errors. Prior research has investigated adaptive UKF-based approaches for state-of-charge estimation that update noise statistics online to enhance robustness under real driving conditions (Xing & Wu, 2021). While such adaptive filtering methods can improve estimation stability, they may still require careful handling of covariance updates and parameter tuning. In contrast, the proposed hybrid architecture reduces the need for manual configuration by combining a trainable PINN component with automatic covariance adaptation in the A-UKF framework.

B. Yao *et al*. (2024) proposed a modified UKF with an improved parameter identification procedure, showing improved SoC estimation in a number of scenarios; nevertheless, the method remains within the "model + filter" paradigm and is therefore limited when battery physics exhibits strong nonlinearity. In contrast, A-UKF-PINN embeds physical laws within the PINN, increasing overall accuracy and transferability under dynamic V2G loads. S. Wang *et al*. (2024) used optimisation schemes (PSO and others) for fine filter tuning and error reduction; such methods improve performance but require frequent offline optimisation and remain sensitive to the initial model. A-UKF-PINN reduces the need for regular offline optimisation, since PINN learns the physics of the modelled process and A-UKF adapts online, facilitating operation within a scalable Edge-Cloud DT architecture.

Overall, analysis of these ten studies showed that contemporary research achieved significant improvements in one or two dimensions (filter adaptation, advanced circuit models, data-driven hybrids, or optimisation-based approximations), but rarely simultaneously addressed physical fidelity, robust online correction under noisy real-world conditions, and architectural suitability for Digital Twin deployment in Smart Grid systems. The proposed A-UKF-PINN integrates these components: PINN provides physical interpretability and the ability to model nonlinear thermal effects, A-UKF facilitates adaptive dynamic synchronisation, and the Edge-Cloud partitioning supports practical implementability. Experimental results indicate an RMSE reduction to approximately 0.87%, suggesting improved performance of the DT compared to the DM.

Conclusions

This study was devoted to the algorithmic analysis of the potential performance advantages of the DT over the DM for controlling critically important CTSs, using the SG as an example. Based on the conducted theoretical analysis and experimental validation, the central thesis of the study was achieved. The functional superiority of the DT was demonstrated through the development and validation of an adaptive hybrid A-UKF-PINN model. This model, which constitutes the key original contribution, for the first time combines physical fidelity (PINN principles) with adaptive dynamic synchronisation (A-UKF), making it possible to overcome the nonlinearity and noise of real PMU/IoT data. As a result of experimental comparison at a critical CTS node (SoC prediction in V2G energy storage systems), a minimal prediction deviation with an RMSE error of 0.87% was achieved. This indicator confirms the possibility of reliable real-time maintenance of knowledge equivalence and corresponds to a 44% error reduction compared to the conventional digital model (ECM + Kalman Filter) and a 56% reduction compared to the baseline DT approach (pure PINN). Thus, the achieved numerical results serve as direct evidence of the superiority of the DT integrated with advanced hybrid algorithms. Based on the obtained results, key engineering and economic conclusions were formulated. It is proven that the transition to DT is an economic imperative for SGs. Architectural validation showed that achieving high accuracy (RMSE 0.87%) and scalability of the DT solution at the scale of the entire CTS is possible only through the use of a hierarchical Edge-Cloud computing architecture and compliance with reference architectural standards. This confirms that an effective DT is formed by the logical integration of the DT domain and the communication domain into a unified cyber-physical system. Further research should focus on the development of decentralised learning mechanisms to protect data privacy when scaling distributed DTs in SGs, and on the creation of self-managing DTs capable of autonomously adapting their hybrid models to long-term asset ageing.

Acknowledgements

None.

Funding

None.

Conflict of Interest

None.

References

[1] Alharbey, R., Shafiq, A., Daud, A., Dawood, H., Bukhari, A., & Alshemaimri, B. (2024). Digital twin technology for enhanced smart grid performance: Integrating sustainability, security, and efficiency. *Frontiers in Energy Research*, 12, article number 1397748. doi: 10.3389/fenrg.2024.1397748.

[2] Bouchareb, H., Saqli, K., M'sirdi, N.K., & Oudghiri Bentaie, M. (2024). Adaptive joint sigma-point Kalman filtering for lithium-ion battery parameters and state-of-charge estimation. *World Electric Vehicle Journal*, 15(11), article number 532. doi: 10.3390/wevj15110532.

[3] Das, O., Zafar, M.H., Sanfilippo, F., Rudra, S., & Kolhe, M.L. (2024). Advancements in digital twin technology and machine learning for energy systems: A comprehensive review of applications in smart grids, renewable energy, and electric vehicle optimization. *Energy Conversion and Management: X*, 24, article number 100715. doi: 10.1016/j.ecmx.2024.100715.

[4] Gao, B., Yao, R., & Li, Y. (2025). Physics-informed neural networks with adaptive loss weighting algorithm for solving partial differential equations. *Computers & Mathematics with Applications*, 181, 216-227. doi: 10.1016/j.camwa.2025.01.007.

[5] Guo, J., Liu, S., & Zhu, R. (2023). An unscented Kalman filtering method for estimation of state-of-charge of lithium-ion battery. *Frontiers in Energy Research*, 10, article number 998002. doi: 10.3389/fenrg.2022.998002.

[6] Hosseininasab, S., Momtaheni, N., Pischinger, S., & Günther, M. (2023). State-of-charge estimation of lithium-ion batteries using an adaptive dual unscented Kalman filter based on a reduced-order model. *Journal of Energy Storage*, 73(D), article number 109011. doi: 10.1016/j.est.2023.109011.

[7] Hu, L., Hu, R., Ma, Z., & Jiang, W. (2022). State of charge estimation and evaluation of lithium battery using Kalman filter algorithms. *Materials*, 15(24), article number 8744. doi: 10.3390/ma15248744.

[8] ISO/DIS 23247-1:2021. (2021). *Automation systems and integration – Digital Twin framework for manufacturing. Part 1: Overview and general principles*. Retrieved from https://www.iso.org/standard/77615.html.

[9] ISO/DIS 23247-2:2021. (2021). *Automation systems and integration – Digital Twin framework for manufacturing. Part 2: Reference architecture and application development*. Retrieved from https://www.iso.org/obp/ui/es/#iso:std:iso:23247:-2:ed-1:v1:en.

[10] Julier, S.J., & Uhlmann, J.K. (1997). A new extension of the Kalman filter to nonlinear systems. In *Proceedings of AeroSense: Signal processing, sensor fusion, and target recognition VI* (Vol. 3068). Bellingham: SPIE. doi: 10.1117/12.280797.

[11] Kabir, M.R., Halder, D., & Ray, S. (2024). Digital twins for IoT-driven energy systems: A survey. *IEEE Access*, 12, 177123-177143. doi: 10.1109/access.2024.3506660.

[12] Lin, X., Tang, Y., Ren, J., & Wei, Y. (2021). State of charge estimation with the adaptive unscented Kalman filter based on an accurate equivalent circuit model. *Journal of Energy Storage*, 41, article number 102840. doi: 10.1016/j.est.2021.102840.

[13] Lyu, L., Jiang, B., Zhu, J., Wei, X., & Dai, H. (2024). An adaptive combined method for lithium-ion battery state of charge estimation using long short-term memory network and unscented Kalman filter considering battery aging. *Batteries & Supercaps*, 7, article number e202400441. doi: 10.1002/batt.202400441.

[14] Mchirgui, N., Quadar, N., Kraiem, H., & Lakhssassi, A. (2024). The applications and challenges of digital twin technology in smart grids: A comprehensive review. *Applied Sciences*, 14, article number 10933. doi: 10.3390/app142310933.

[15] Raissi, M., Perdikaris, P., & Karniadakis, G. (2019). Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational Physics,* 378(1), 686-707. doi: 10.1016/j.jcp.2018.10.045.

[16] Vychuzhanin, V., & Vychuzhanin, A. (2025). *Digital methods and models for control and survivability of complex technical systems*. Lviv-Torun: Liha-Pres.

[17] Wang, F., Zhai, Z., Zhao, Z., Di, Y., & Chen, X. (2024). Physics-informed neural network for lithium-ion battery degradation stable modeling and prognosis. *Nature Communications*, 15, article number 4332. doi: 10.1038/s41467-024-48779-z.

[18] Wang, S., Zhang, S., Wen, S., & Fernandez, C. (2024). An accurate state-of-charge estimation of lithium-ion batteries based on improved particle swarm optimization-adaptive square root cubature Kalman filter. *Journal of Power Sources*, 624, article number 235594. doi: 10.1016/j.jpowsour.2024.235594.

[19] Wang, Z., Shen, J., & Xu, Y. (2024). State-of-charge estimation for lithium-ion battery based on adaptive extended sliding innovation filter. *Energies*, 17(14), article number 3495. doi: 10.3390/en17143495.

[20] Wei, Y. (2024). State of charge estimation for lithium battery based on fractional order square root cubature Kalman filter and adaptive multi-innovation unscented Kalman filter. *Proceedings of the Bulgarian Academy of Sciences*, 77(4), 485-495. doi: 10.7546/CRABS.2024.04.02.

[21] Xing, J., & Wu, P. (2021). State of charge (SOC) estimation of lithium-ion batteries based on an improved adaptive unscented Kalman filter. *Sustainability*, 13(9), article number 5046. doi: 10.3390/su13095046.

[22] Yao, B., Cai, Y., Liu, W., Wang, Y., Chen, X., Liao, Q., Fu, Z., & Cheng, Z. (2024). State-of-charge estimation for lithium-ion batteries based on modified unscented Kalman filter using improved parameter identification. *International Journal of Electrochemical Science*, 19, article number 100574. doi: 10.1016/j.ijoes.2024.100574.

[23] Zeng, Y., Li, Y., & Yang, T. (2023). State of charge estimation for lithium-ion battery based on unscented Kalman filter and long short-term memory neural network. *Batteries*, 9(7), article number 358. doi: 10.3390/batteries9070358.

Гібридна цифрова двійникова архітектура A-UKF–PINN для оцінювання стану в реальному часі в інтелектуальних електромережах (Smart Grid)

Володимир Вичужанін

Доктор технічних наук, професор
Національний університет «Одеська політехніка»
65044, просп. Шевченка, 1, м. Одеса, Україна
https://orcid.org/0000-0002-6302-1832

Олексій Вичужанін

Доктор філософії, асистент
Національний університет «Одеська політехніка»
65044, просп. Шевченка, 1, м. Одеса, Україна
https://orcid.org/0000-0001-8779-2503

Анотація. Зростаюча мінливість, нелінійність та вимоги до роботи в режимі реального часу розумних енергомереж роблять статичні цифрові моделі недостатніми для надійної оцінки стану та контролю розподілених активів, таких як системи зберігання Vehicle-to-Grid (V2G). Метою дослідження було формальне та імітаційне обґрунтування переваг динамічних цифрових двійників (DTs) порівняно зі статичними data model (DM) у задачах оцінювання стану літій-іонних накопичувачів у реальному часі. Для цього запропоновано гібридну архітектуру A-UKF–PINN, що поєднує адаптивний несцентований фільтр Калмана (A-UKF), який забезпечує стійке оцінювання стану за наявності шумів і невизначеностей, із фізично інформованою моделлю PINN (Physics-Informed Neural Network), яка враховує динаміку та нелінійні процеси акумуляторного елемента. Новизна роботи полягає в інтеграції цих компонентів в єдину модель із двобічною синхронізацією, що підвищує стабільність прогнозування та істотно зменшує десинхронізацію між моделлю й фізичним об'єктом в умовах Smart Grid. Імітаційну валідацію проведено на робочих циклах V2G з урахуванням змодельованих шумів датчиків PMU/IoT (Phasor Measurement Unit / Internet of Things). Отримане значення Root Mean Square Error (RMSE) 0,87 % продемонструвало підвищення точності на 44 % порівняно з традиційною DM (ECM (equivalent circuit models) + UKF, RMSE 1,98 %) та на 56 % відносно базового цифрового двійника (чистий PINN). Архітектурна оцінка підтвердила необхідність використання ієрархічної платформи Edge-Cloud, що забезпечує оптимальний розподіл обчислювальних навантажень: навчання PINN у хмарному середовищі та високочастотне оцінювання стану на периферії. Запропонована архітектура формує основу для масштабованих динамічних DT у Smart Grid, сприяє зниженню операційних ризиків, підтримує впровадження стратегій проактивного технічного обслуговування та підвищує ефективність життєвого циклу енергетичної інфраструктури

Ключові слова: гібридне моделювання; фізично інформовані нейронні мережі; несцентований фільтр Калмана; функціональна перевага; Edge-Cloud

Journal homepage: https://itce.vn.ua/en

Information Technologies and Computer Engineering **Vol. 23, No. 1. 2026**

Article's History: Received: 17.11.2025 Revised: 19.02.2026
Accepted: 26.03.2026 Published: 20.04.2026

ISSN 1999-9941; e-ISSN 2078-6387
UDC 621.396.6:355.40
DOI: 10.31649/vitce/1.2026.153

Analysis of the construction of a communication network of the tactical control link based on software-defined radio communication means

Hryhorii Radzivilov
PhD in Technical Sciences, Associate Professor
Kruty Heroes Military Institute of Telecommunications and Information Technology
01011, 45/1 Knyaziv Ostrozkykh Str., Kyiv, Ukraine
https://orcid.org/0000-0002-6047-1897

Dmytro Pavliuk*
Adjunct
Kruty Heroes Military Institute of Telecommunications and Information Technology
01011, 45/1 Knyaziv Ostrozkykh Str., Kyiv, Ukraine
https://orcid.org/0000-0001-8461-3899

Abstract. The purpose of the study was to develop an architectural solution for the construction and management of a tactical communication circuit of the company-battalion-brigade levels based on flexible radio platforms to ensure continuity and reliability of communication in conditions of active enemy counteraction. Methods of structural and functional modelling, scenario and comparative analysis were used. It was established that in conditions of electronic warfare (EW), the communication circuit of the company-battalion-brigade levels should be built as a hybrid multi-level architecture, which at the company level combines separate voice communication and data transmission channels, uses a self-organised Mobile Ad Hoc Network based on software-defined radio systems for tactical exchange, and at the battalion level – a gateway node for traffic aggregation, routing, and integration with higher-level communication channels via Low Earth Orbit satellite backhaul. It was shown that Digital Mobile Radio should be used for the voice loop of the command-and-control minimum, Software-Defined Radio Mobile Ad Hoc Network with Multiple-Input Multiple-Output – for the tactical data transmission layer, and LEO satellite backhaul – as a main or backup communication channel between the battalion-brigade levels. An iterative algorithm for planning and configuring the tactical command link communication network in the presence of electromagnetic interference and limited resources was proposed. The advantages of the model were increased availability of the command-and-control minimum, preservation of controllability, prioritisation of traffic by quality of service and controlled degradation of services. Its effectiveness was determined by the continuity of voice communication, data exchange stability, speed of connection restoration and communication redundancy between the company-battalion-brigade levels. The practical significance lies in the possibility of applying the results by specialists of communication units during planning, deployment and adjustment of the tactical communication circuit of the company – battalion – brigade in field conditions in a complex electromagnetic environment and under active countermeasures

Keywords: communication network; self-organising network; multichannel architecture; gateway node; satellite communication channel; electronic warfare

Introduction

Stable communication of the tactical control link (TCL) (company – battalion – brigade) is a basic condition for effective command and control (C2) in an environment with limited radio resources, high mobility, uneven coverage and the effects of electronic warfare (EW). In this context, Software-Defined Radio (SDR) tools and self-organising ad

Suggested Citation:
Radzivilov, H., & Pavliuk, D. (2026). Analysis of the construction of a communication network of the tactical control link based on software-defined radio communication means. *Information Technologies and Computer Engineering*, 23(1), 153-169. doi: 10.31649/vitce/1.2026.153

*Corresponding author

hoc networks are the basis of flexible TCL networks, where the quality of service and survivability are determined not by the "passport" of radio stations, but by the architecture, control algorithms and information security (IS) measures. However, at the tactical level, communication is built from a combination of different carriers (narrowband voice, packet channels based on SDR, satellite backbone channel), but without a formalised methodology for the integration and control under the influence of EW. As a result, the "passport" characteristics of individual devices do not guarantee stable Key Performance Indicators (KPIs) of the network (C2 availability, delay/variation (oscillation) of the delay, delivery reliability, recovery time) and do not ensure systematic consideration of IS requirements.

Modern scientific discourse has proposed a number of approaches to counteracting interference in Mobile Ad hoc Network (MANET). In the study, J. Kim *et al*. (2021) showed that inherent interference effects can critically degrade MANET connectivity if media access and routing protocols do not take into account interference interaction. This emphasises the need for specialised protocols/policies that reduce network degradation under interference effects and preserve the operability of management services. In a broader review, Z. Patel *et al*. (2023) systematised the technological trends of tactical self-organising (ad hoc) networks and showed that the key benefits are not provided by isolated "radio solutions", but by the coordinated design of the radio signal form – network layer – service policies. This formalises the typical trade-offs "delay/bandwidth/survivability/controllability", which leads to the formulation of the problem of multi-criteria optimisation of the TCL network construction. In the work of N. Chen *et al*. (2025), the network challenges of satellite-ground integrated networks were generalised, in particular the problems of routing, mobility, resource management and Quality of Service (QoS) in heterogeneous topology. Such an approach means that effective integration requires communication carrier selection policies and traffic management mechanisms at the network level, and not only at the "satellite connection" level. For the physical layer and transmission mode selection, A. Mureșan & P. Bechet (2024) found that different radio signal forms provide different transmission modes and balance robustness, bandwidth, and delay differently. The authors' findings are useful for formalising the rules for selecting a communication medium in a TCL depending on the type of environment (Line of Sight (LOS)/ Non-Line of Sight (NLOS)) and traffic priorities.

A separate scientific direction is related to the use of Multiple-Input Multiple-Output (MIMO) as a tool to increase resilience and throughput in interference conditions. K.-P. Hui *et al.* (2024) demonstrated at the prototype level the potential of MIMO with interference suppression for tactical communication, where interference can dominate the "clean" radio channel. This reinforces the thesis that MIMO should not be considered as a "speed option", but as an element of link survivability and QoS stabilisation. Additionally, in the work of C.E. Thornton *et al*. (2023) considered the role of sidelink in 5G/5G-Beyond for multi-hop tactical networks and showed that direct inter-device communication via the PC5 interface can support multi-hop connectivity and reduce dependence on centralised infrastructure. This justifies direct inter-device communication via the PC5 interface as an additional communication medium for the self-organising TCL network and reinforces the concept of multichannel architecture.

For Ukraine, taking into account the experience of the Anti-Terrorist Operation (ATO)/Joint Forces Operation (JFO) and the full-scale war since 2022, the construction of communication networks that are rapidly deployed, adaptable to interference, and support multichannel data exchange is of paramount importance. In the study, M. Masesov *et al*. (2021) found that active traffic queue management using fuzzy logic can stabilise latency and reduce the effects of congestion in tactical radio networks. According to the authors, QoS in TCL should be ensured by traffic management, and not only by choosing a "more powerful" channel. In the work of R. Shtonda *et al*. (2023), the authors demonstrated that small-sized digital tropospheric stations can be used as an alternative communication channel in operational conditions to maintain connectivity with limited availability of other means. This justifies tropospheric communication as an additional carrier/backbone channel in the multichannel TCL architecture, which enhances network survivability in the event of degradation of the ground-based self-organising network or in the absence of infrastructure. Analysing the use of software-defined radio communication systems in mobile radio networks, S.V. Salnyk & P.H. Sydorkin (2024) emphasised the role of SDR as a technological basis for flexible configuration of operating modes and integration of various protocols. This forms the basis for considering SDR not as a separate "device", but as a platform for building network functions and adapting to application conditions.

Despite the existing body of works, there remains a gap between individual technological solutions (radio signal form, MIMO, Active Queue Management (AQM), sidelink, satellite channel integration) and a holistic methodology for building a TCL network, which simultaneously takes into account the company-battalion-brigade hierarchy, multi-bearer architecture, QoS/survivability and basic IS requirements. Therefore, the purpose of the study was to substantiate an engineering approach to building and managing a company-battalion-brigade tactical communication circuit based on flexible radio platforms to ensure the reliable functioning of tactical communication under active countermeasures. To achieve the goal, the following tasks were set: to describe the structure of the TCL network (company – battalion – brigade) to form a reproducible planning/configuration methodology (algorithm) taking into account QoS, managed degradation, Satellite Communication (SAT) integration and KPI verification, to compare the characteristics of the main communication technologies, in particular Digital Mobile Radio (DMR), SDR-MANET networks and Low Earth Orbit Satellite (LEO-SAT), to

determine the functional roles in the TCL and to substantiate the KPI-oriented hybrid multichannel TCL model for the conditions of Ukraine "DMR + SDR(MIMO)-MANET + LEO-SAT backhaul".

Materials and Methods

The study was carried out as an engineering and architectural justification for building a communication network for the tactical command link in conditions of active counteraction. Within the framework of structural and functional modelling, the TCL network was presented as a hierarchical-cluster multi-level system of company – battalion – brigade levels, where at the company level subscriber nodes (infantry units, crews, unmanned aerial vehicle (UAV) operators, tablets/terminals, sensors), data access nodes and, if necessary, relay nodes were considered. At the battalion level, aggregation nodes and gateways were identified that provided unification of company segments, traffic prioritisation, routing, relaying, network control and monitoring. At the brigade level, control nodes were identified, within which C2 services, routing, logging and key management were implemented. The types of communication channels considered were DMR channels for the C2-minimum voice circuit, SDR-MANET channels for tactical packet data exchange, and a satellite backhaul channel for main or backup communication between the battalion and brigade levels. Topologically, the network was described as a combination of local company clusters, united through battalion gateway nodes into a hierarchical structure, where mesh or ad hoc interaction prevailed within the clusters, and gateway and main connections were implemented between the control levels (Fig. 1).

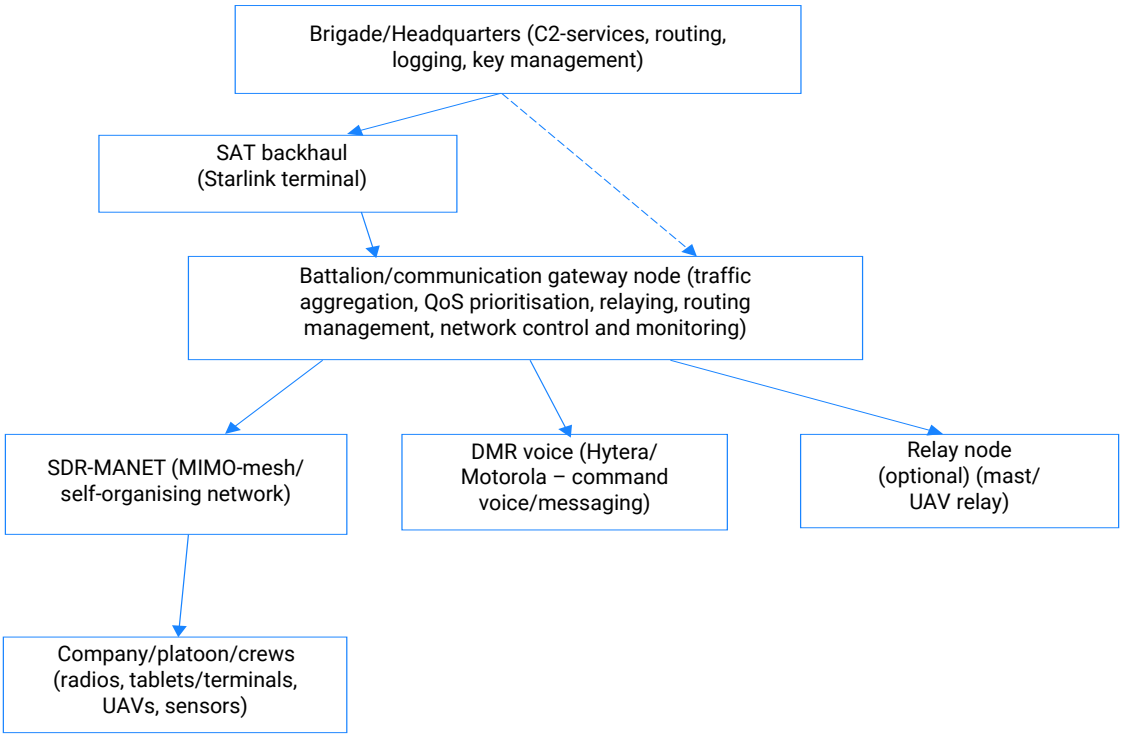

Figure 1. Architecture of the TCL communication network

Source: compiled by the authors based on European Telecommunications Standards Institute (2016), J. Suess (2022), V.H. Sholudko *et al.* (2023), A.A. Hrozdov *et al.* (2024), L. Bojor *et al.* (2024), P.V. Khomenko *et al.* (2025)

The European Telecommunications Standards Institute (2016) standard was used as a regulatory basis for the analysis of the DMR voice circuit, which provided C2 radio exchange, and the architectural model of the company-battalion-brigade circuit was used as an input basis for further analysis of technologies, KPIs and network operation scenarios, in particular for determining the battalion communication node – gateway as a key element of traffic aggregation, routing, QoS prioritisation and route reservation. The described architectural model was used as an input basis for further analysis of communication technologies, determination of KPIs, operation scenarios and formation of the TCL network construction algorithm. Using the method of comparative and system analysis, typical communication means were compared: Hytera (2018) Professional Digital (PD)7i (China) and Motorola Digital Mobile (DM)4000e (USA) (Motorola Solutions, 2026) as typical representatives of DMR for basic voice C2 (Push-To-Talk (PTT)/C2-minimum); Silvus StreamCaster (SC)4400 Enhanced (E) (USA) (Silvus Technologies, 2025) – as a representative of SDR-MANET (MIMO) for tactical broadband data exchange; Starlink (2026) (USA) – as LEO-SAT backhaul/reserve at the battalion-brigade level. These models of communication means as representatives of different technological classes were chosen because such systems were used in real combat missions in the conditions of war in Ukraine during

2022-2025. The comparison was performed according to the engineering-role principle and in conformity with the basic technical parameters: technological class, band/channelling, frequency ranges, power, bandwidth, delay, scalability, MIMO/beamforming, encryption and IS, since it was these parameters that comprehensively characterised deployment, QoS, survivability/scalability and IS in countermeasure conditions.

The method of structural-logical classification systematised the distribution of communication services and technologies in the TCL according to the following criteria: priority services, main technology, reserve/hybrid, sequence of controlled degradation and minimum IS requirements. This approach unified the policy of prioritisation and degradation between levels and the preservation of basic IS mechanisms even in the minimum mode (encryption, key management/zeroize, segmentation of C2/data domains, protection of gateways and control nodes). Using the method of system analytical synthesis, an applied framework for assessing the stability of TCL communication was formed, a set of KPIs focused on the minimum required command and control services was determined, and a field verification protocol was proposed (control pairs of nodes, traffic profiles, measurements in three modes: normal/overload/interference-EW) with fixation of the sequence of service degradation. For this, the National Institute of Standards and Technology standard (2001; 2019) was used as a normative basis, and the AES algorithm was considered as a basic guideline for describing the requirements for traffic encryption in TCL.

Using the method of systematic analysis of the experience of using tactical communication of the Armed Forces of Ukraine in real combat conditions, typical application scenarios for Ukraine (city/NLOS, partial NLOS, LOS with vulnerability to EW, variable LOS) were systematised and translated into engineering requirements for communication media and topology (node density, relay, multichannel mode, readiness for SAT) and modern requirements for TCL were formed. The source base was the analytical report by J. Watling & N. Reynolds (2023), as the context of combat conditions 2022-2023, the annual report of L3Harris Technologies, Inc. (2024), as an indicator of practical relevance on the practical significance of SDR/network solutions. The formed requirements were used as a basis for substantiating the hybrid architecture of the TCL network "DMR + SDR(MIMO)-MANET + LEO-SAT backhaul", determining the functional distribution of roles between its components and further developing an algorithm for its planning and configuration. At the final stage, a logical-algorithmic generalisation of the engineering approach to building a TCL network was applied, within which a step-by-step algorithm was formed, covering the collection of input data, determination of optimisation criteria, design of the C2-minimum and data layer, selection of gateway nodes and relay facilities, QoS settings, integration of SAT, implementation of IS measures and iterative verification of KPIs with correction of network parameters. This was done in order to standardise engineering solutions during planning, deployment and operational management of tactical communication and to ensure guaranteed operability of priority services with limited resources and the influence of EW.

Results and Discussion

Hierarchical architecture and choice of communication technologies for tactical control link under EW conditions

To ensure the effectiveness of tactical communication at different levels of control, it is necessary to take into account not only the technical characteristics of communication facilities, but also the ability of the network to adapt to changing conditions, such as EW and limited resources. In this context, it is important to use such a TCL network that allows for communication stability with minimal loss of connectivity, while optimising the use of technologies to ensure proper controllability and reliability. In the current study, the TCL communication network is considered as a multi-level system of nodes and channels at the company – battalion – brigade levels, where each level has its own functions, services, and requirements for stability under EW conditions. This approach corresponds to the practice of organising military communication, within which communication is considered as an element of the control system, and not as a set of individual radio stations (Sholudko *et al*., 2023).

Within the framework of the analysed model at the company level, the priority is to ensure C2-minimum (command voice, short messages/coordinates) with simple deployment and resistance to local node losses. It is engineeringly expedient to divide the voice control channel and the data channel so that loaded services do not degrade control (Hrozdov *et al*., 2024). To implement the voice control channel, DMR, standardised by the European Telecommunications Standards Institute (2016), is used, which supports basic group services and typical radio exchange organisation modes. At the same time, a local SDR-MANET segment is formed for data transmission and situational awareness (SA), capable of self-organisation and self-recovery of routes during movement and loss of some nodes. Increasing the efficiency of data exchange – increasing the throughput and noise immunity of the MANET segment – is associated with the use of MIMO approaches, in particular in the context of countering jamming, which is relevant for an environment with intensive EW.

The battalion level performs traffic aggregation and routing functions, for which gateway nodes are formed that unite company segments, perform docking of MANET ↔ Internet Protocol (IP) services and implement the QoS traffic prioritisation mechanism, provide redundancy and controlled degradation of services. In addition, these nodes monitor and log the network status taking into account the requirements of electromagnetic discipline (Sholudko *et al*., 2023; Hrozdov *et al*., 2024). Gateway nodes not only perform traffic aggregation and routing, but also have the task of

maintaining network stability, guaranteeing its ability to self-organise and restore routes when the topology changes. This allows the network to function effectively even in the event of malfunctions or losses of some nodes, which is important for ensuring continuous communication in EW conditions or in combat.

At the brigade level, a control backbone is formed that integrates C2 services and provides communication with higher levels via a satellite channel. However, the impact of cyber impacts and jamming on the space and ground communication segments necessitates considering the satellite channel as a managed resource with mechanisms for prioritisation, degradation, and redundancy (Bojor *et al.*, 2024). This allows for the stability of communication between the battalion and brigade levels, even in the event of serious interference. At the brigade level, the satellite channel is the main backbone channel for communication with higher levels, but to ensure reliability and fault tolerance, this channel is supplemented with backup paths, such as SDR-MANET for alternative traffic, which allows for quick switching to other channels in the event of malfunctions or high interference.

Thus, the engineering model of the TCL network should be built as a hierarchical multilayer system: DMR provides reliable command voice communication and short messages at the tactical level; SDR-MANET with MIMO forms a resilient tactical layer for data transmission with self-organisation and self-recovery of routes; and SAT backhaul is used as a backbone channel for communication between the battalion and the brigade, with support for redundancy and traffic prioritisation. Such a hybrid architecture allows maintaining a high level of network adaptability to changes in topology and electromagnetic interference conditions, which makes it resistant to interference and ensures continuous communication at all levels of control. At the same time, the use of different levels of redundancy and degradation of services ensures that even under conditions of high load or failures, key services, such as C2, will remain available.

For the effective practical implementation of the considered TCL network model, it was important to analyse available communication technologies that can be applied in different segments of the tactical architecture. Comparison of technologies is necessary to select the most optimal solutions, taking into account the requirements for resilience, scalability, and QoS in conditions where the network is exposed to EW and changes during combat operations. Since the technologies used belong to different classes and perform specific functions in the network architecture (from portable radios to satellite channels), the comparison was not a direct "one-to-one" comparison. Instead, key technical parameters that directly affect the performance of the TCL network, such as bandwidth, latency, scalability and security capabilities, were considered. This allowed comparing technology classes that represent specific models of real-world communication facilities that can be used in the corresponding TCL circuits (Table 1).

Table 1. Technical characteristics of communication facilities for TCL circuits

Parameter	Hytera PD7i (portable DMR)	Motorola MOTOTRBO DM4000e (automotive DMR)	Silvus SC4400E (SDR-MANET node)	Starlink (LEO satellite channel, backhaul)
Technology/class	DMR (Tier II/III)	DMR (Tier II/III)	SDR-MANET (MN-MIMO)	LEO satellite broadband channel (SATCOM)
Channel bandwidth/ channel spacing	12.5/20/25 kHz (channel step)	12.5/20/25 kHz	20/10/5 MHz (opt. 2.5/1.25)	depends on the service/ terms
Frequency ranges	VHF 136-174; UHF 400-470/450-520/350-400; 210-270; 806-941 (for trunking)	136-174; 300-360; 350-400; 403-470; 450-527 MHz	Available ranges from 300 MHz to 6 GHz	The terminal's RF band is not fixed
Transmitter power	VHF: High 5W/Low 1W	High 25-45 W (depending on range)	"Native transmit power" up to 20 W (depending on configuration/model), efficiency increased by beamforming	Not used as "radio station power" within the scope of TCL role comparison
Bandwidth (class/typical)	Limited to DMR class (voice + basic data services)	Limited to DMR class (voice + data services)	Up to 100 Mbps (adaptive)	25-220 Mbps (downlink), 5-20 Mbps (uplink)
Latency (class/typical)	Suitable for PTT/voice (class)	Suitable for PTT/voice (class)	Average ~7 ms (at 20 MHz)	Typically 25-60 ms on land
Network scalability	Depends on network/relay organisation	Depends on configuration (MOTOTRBO network modes)	550+ nodes	Depends on coverage/load
MIMO/ beamforming	No (as DMR class)	No (as DMR class)	4×4 MIMO, beamforming; spatial multiplexing	Not a "tactical mesh network", the terminal uses an antenna with electronic beam steering within the system

Table 1. Continued

Parameter	Hytera PD7i (portable DMR)	Motorola MOTOTRBO DM4000e (automotive DMR)	Silvus SC4400E (SDR-MANET node)	Starlink (LEO satellite channel, backhaul)
Encryption and IS	AES/ARC4, end-to-end + over-the-air, analogue scrambling	AES-256 (declared in features)	DES56 (standard), AES256 (optional), Zeroize Crypto	Jamming/cyber-impact risks confirmed, service encryption parameters not detailed

Note: AES – Advanced Encryption Standard; ARC4 – Alleged RC4; RC4 – Rivest Cipher 4; DES – Data Encryption Standard
Source: compiled by the authors based on Hytera (2018), National Institute of Standards and Technology (2019), European Telecommunications Standards Institute (2023), D. McCrory (2023), L. Bojor *et al*. (2024), Silvus Technologies (2025), Motorola Solutions (2026), Starlink (2026)

Analysis of the table data showed that DMR technology is appropriate for use as a basic TCL layer, intended primarily for organising PTT communication and transmitting short service messages. Its advantages are relative ease of implementation, energy efficiency, proven solutions and suitability for operation in conditions of limited infrastructure. At the same time, the data transmission capabilities of this class of systems remain limited, which does not allow considering DMR as the main technology for supporting modern situational awareness services, telemetry and other streaming or broadband applications. The bandwidth of DMR is limited by the class of technology, which allows efficiently processing only voice traffic and basic data services, but it cannot support high-speed applications such as video or big data.

SDR-MANET technology fundamentally differs from DMR in that it is focused on broadband tactical data exchange in dynamic network topology. Its key advantages are self-organisation, multi-hop routing, self-recovery of routes, as well as support for MIMO and beamforming mechanisms. Together, this provides higher throughput, greater resilience to changes in the radio environment and better suitability for supporting situational awareness services in a complex electromagnetic environment. The bandwidth of the presented SDR-MANET model (Silvus SC4400E) – up to 100 Mbps (adaptive) – enables the support of broadband services that are critical for modern military applications, such as video surveillance or big data transmission. This makes this technology an ideal solution for tactical networks with high bandwidth and adaptability requirements.

The technical characteristics of the Starlink SAT model (the model's bandwidth ranges from 25 to 220 Mbps for downlink and 5-20 Mbps for uplink) show that LEO-SAT backhaul allows for effective communication between battalion and brigade levels even under difficult conditions with high loads. However, due to the higher latency (25-60 ms), the satellite channel is not suitable for tactical data exchanges in real time. Therefore, within the proposed architecture, LEO-SAT backhaul technology is considered not as an element of a tactical mesh network, but as a backbone or backup channel for communication between higher levels of control. Its use is advisable due to its high bandwidth and acceptable latency, which allows for traffic aggregation and access to a higher-level network. However, the use of such a channel requires a controlled operation mode taking into account the risks of jamming, cyber impact, as well as the dependence of actual service parameters on the system operating conditions.

According to IS criteria, not only encryption algorithms were of crucial importance, but also key management mechanisms, access control, and the ability to quickly reset or destroy cryptographic parameters (zeroize) in case of a threat of compromise. All technologies presented in the table support encryption (AES, DES, ARC4), which provides a basic level of information protection. However, each technology has its limitations, in particular, encryption in LEO-SAT is not detailed, which may create additional risks in the context of cyber threats. Encryption mechanisms can affect data transfer speed and network latency. For DMR, where encryption is used for voice channels, the impact on data transfer speed is insignificant, since such channels are designed to transmit small amounts of data (voice, short messages). However, for SDR-MANET systems with large amounts of data, such as situational awareness, additional encryption mechanisms can lead to increased latency due to processing of large data and high requirements for computing power, which is important to consider when planning a TCL network. Therefore, a hybrid model is technically justified for the TCL, in which DMR is used to provide the minimum necessary control loop, SDR-MANET as the main tactical data layer, and LEO-SAT as a backhaul layer with QoS policies and redundancy. The comparison showed that the DMR channel should be used as a voice C2 loop, which provides minimal system controllability even in the event of degradation or loss of the data network. SDR-MANET should serve as the main tactical data transmission channel, as it provides network self-organisation, multi-hop routing and support for situational awareness services. LEO-SAT backhaul should be used as a backbone or backup battalion-brigade communication channel, designed for traffic aggregation and access to a higher-level network.

The results of the engineering analysis of the roles of communication channels in the tactical control link indicate that at the company level, it is reasonable to divide the voice control channel and the data channel so that loaded services do not cause control degradation. This statement is consistent with Y.W. Lo *et al*. (2024), who analysed the use of DMR in an applied monitoring system and pointed out the need to evaluate the reliability and performance of services under real load conditions. The researchers treat DMR not as a "default channel", but as a technology layer with measurable characteristics. In the context of TCL, this supports the interpretation of DMR given in the current study as a basic C2 loop (voice and short messages) with predictable behaviour, while data exchange and SA should be moved

to a separate SDR-MANET segment to avoid competition for resources within the critical control loop. Accordingly, separating the circuits has not only architectural but also methodological meaning: it allows setting KPIs separately for C2 and for SA/data and checking the performance under conditions of changing load and interference.

In the study by J. Suomalainen *et al.* (2022), tactical mobile networks were considered as isolated tactical segments in which cyber defence and response mechanisms must operate autonomously and cannot rely on the constant availability of a remote Security Operations Centre (SOC). The key conclusion of the authors is that traffic prioritisation must be security-oriented and dynamic: priority decisions are based on traffic analysis and security posture assessment, and as an example of an "intelligent" response to availability threats, dynamic adjustment of live video stream quality parameters is demonstrated. In the current study, these provisions are related to the fact that the criterion for the success of building a TCL network is the implementation of KPIs of priority services with the dominance of the C2-minimum, and at the battalion level, the functions of QoS prioritisation, managed degradation, and monitoring/logging were concentrated in the gateway node, which serves as a practical point of implementation of such dynamic policies under resource shortages under EW conditions.

The functional role of the gateway node at the battalion level is justified by the concentration of aggregation and traffic management functions, which provides MANET ↔ IP docking, QoS prioritisation, redundancy, and managed degradation of services. This approach is conceptually consistent with R. Mahmud *et al.* (2021), who considered SDN-oriented tactical networks as multi-domain systems, where the key condition for supporting heterogeneous services is policy-driven orchestration. The authors proposed a multi-layered Software-Defined Networking (SDN) architecture that provides monitoring and aggregation of network state for orchestrating services in terms of resilience, compatibility, and policy enforcement. In such an architecture, the gateway performs the role of implementing service management policies at the interface of domains, providing monitoring and logging of network state. Therefore, optimality is determined not by the total throughput, but by preserving the operability of priority services by limiting secondary traffic.

According to the analysis of the role of the satellite channel, at the battalion-brigade level it should be considered as a managed backhaul resource that requires prioritisation, degradation, and reservation policies. This statement is consistent with the work of M. Kang *et al*. (2024), which systematises threats to satellite systems in terms of confidentiality, integrity and availability, emphasising the presence of real incidents and the need for not only technological, but also political and organisational countermeasures, which actually leads to the need for "controllability" of the satellite domain at the level of resource access policies. In the study by V.S. Kantheti *et al*. (2023), an anti-jamming approach for cooperative LEO satellite constellations based on distributed combining of received signals (distributed Maximal Ratio Combining (d-MRC), distributed Linear Minimum Mean Square Error (d-LMMSE)) is considered, the effectiveness of which was tested on a hardware and software stand based on SDR. These results are consistent with the current study that satellite backhaul is considered a resource sensitive to EW: the presence of specialised anti-jamming solutions confirms that the interference immunity of the satellite segment is a separate engineering problem. In the proposed TCL architecture of this study, the satellite backhaul channel (LEO/Starlink) must be planned as a managed resource with redundancy and managed degradation/switching modes.

A comparison of communication technologies for the tactical control link showed that for each level of the TCL network it is advisable to use specific technologies – DMR, SDR-MANET and LEO-SAT backhaul, based on the technical characteristics and capabilities. These technologies interact with each other, creating a hybrid architecture in which each of these technologies performs its role, ensuring network stability even in difficult EW conditions. The use of separate circuits for voice control and data transmission allows optimising the network, ensuring continuity and efficiency of operation at all levels of control.

Functional distribution of services and requirements for the TCL network under EW conditions

For the effective operation of the tactical command link network, it is important not only to determine the technical characteristics of the communication means used, but also to configure the functional distribution of services between the TCL levels, taking into account the roles of nodes at each level, which were described in the previous section. Proper organisation of this distribution allows for the optimal combination of different communication technologies, ensuring the priority of important services and creating degradation mechanisms in conditions of limited resources or EW influence. Determining these aspects is critical for ensuring the uninterrupted operation of the TCL network in difficult combat conditions. The functional distribution of technologies and services between different TCL levels, as well as the specified degradation modes and minimum IS requirements for each level are given in Table 2.

Table 2. Distribution of communication services and technologies in the tactical command chain link (company – battalion – brigade) and degradation modes

TCL level/node role	Priority services	Basic technology	Reserve/hybrid	Degradation mode	Minimum IS requirements
Company/ subscriber mode (infantry, crews)	C2-minimum voice, short messages/ statuses	DMR	MANET (via a gateway, if available)	Cutting off "heavy" data → only short messages → C2-minimum (voice)	Encryption; key management/zeroize, minimising open services

Table 2. Continued

TCL level/node role	Priority services	Basic technology	Reserve/hybrid	Degradation mode	Minimum IS requirements
Port/gateway (voice ↔ data)	Voice-data gateway, local routing/ aggregation	DMR + MANET	DMR-only (in degradation)	Video/high bitrates → COP with less frequent updates → short messages + voice	Voice /data key domain separation, gateway authentication, access policies
Battalion/HQ aggregator	C2 data, COP, dispatching, telemetry	SDR MANET (MIMO/mesh)	DMR (voice) + SAT (backhaul if needed)	ISR/video → COP update rate reduction → "control only" (CU/C2 + critical messages)	C2/data segmentation, network management protection, access prioritisation
Battalion/SAT gateway node (if needed)	BLOS/backhaul boot to brigade/senior level	SATCOM (Starlink)	MANET (last mile) + DMR (voice)	Background traffic → C2/service data only → emergency minimum profile	Crypto overlay over the internet channel, access control, endpoint protection
Brigade/ inter-battalion integration node	Stream integration, COP, resource management	MANET + SAT backhaul	DMR (voice C2)	ISR/video → COP with less frequent updates → C2-minimum	Domain-role access model (br/battalion/ company), key distribution, event auditing
Brigade/ strategic backhaul node	Channel to senior level/ interdepartmental interaction	SATCOM	Alternative transport channels (where available)	Non-critical traffic → "management only" → "store and forward" mode for messages	Crypto protection + endpoint control, anti-eavesdropping/ spoofing protection, cyber/EW risk management

Note: ISR – Intelligence, Surveillance, and Reconnaissance
Source: compiled by the authors based on National Institute of Standards and Technology (2001; 2019), European Telecommunications Standards Institute (2016), O. Lavrut *et al*. (2019; 2021), J. Suess (2022), V.H. Sholudko *et al*. (2023), A.A. Hrozdov *et al.* (2024), P.V. Khomenko *et al*. (2025), Silvus Technologies (2025), S. Halwa & L. Harriss (2025), Starlink (2026)

The above matrix confirms the feasibility of a multilayer communication architecture of the TCL according to the primary/fallback bearer principle, where each layer uses the optimal technology for its tasks, taking into account the possibility of degradation when resources are limited. At the company level, the priority is C2-minimum (voice and short messages): DMR provides basic availability, and MANET is used for data transmission in the presence of a gateway. This allows maintaining communication even under limited resource conditions. At the battalion level, where data aggregation and distribution functions (COP, telemetry) are concentrated, SDR-MANET (MIMO/mesh) acts as the main transport layer, and SATCOM acts as a managed backhaul for BLOS and redundancy in case of degradation of the ground topology. At the brigade level, integration between battalions and access to higher levels require a combination of MANET+SAT backhaul with a fixed minimum service profile as a condition for maintaining manageability.

The degradation sequence (limiting high-bitrate ISR/video services, reducing the COP update rate, switching to "control only" mode) supports the prioritisation of critical functions and reduces the risk of network overload. At the same time, degradation should not be accompanied by the abandonment of basic IS mechanisms: encryption, key management/zeroize, C2 segmentation, and protection of control nodes and gateways remain necessary in the minimal mode. Thus, the table formalises the engineering logic of building a TCL network as a hybrid system with many bearers, where MIMO-MANET provides performance and adaptability, DMR provides basic C2 availability at the lower level, and SATCOM provides a backup/transport circuit for BLOS integration between the battalion and the brigade.

The experience of the war in Ukraine has shown that the main problem of tactical communication is not just voice radio exchange, but stable network interaction in conditions of EW, manoeuvre, node losses and the need for operational data exchange between control levels (Watling & Reynolds, 2023). That is why traditional DMR devices, despite the suitability for command voice and short messages, cannot fully meet the needs of TCL, since these devices belong to narrowband systems and are not designed for broadband tactical data exchange and flexible multi-hop networking (European Telecommunications Standards Institute, 2016). In contrast, SDR-MANET platforms with MIMO, such as the Silvus SC4400E, provide higher throughput, spatial immunity to interference, as well as self-organisation and self-healing of routes, which allows more effectively maintaining network connectivity in a complex electromagnetic environment (Silvus Technologies, 2025).

For TCL, performance indicators should characterise not only "channel quality", but also the ability of the network to maintain controllability under conditions of EW, node losses and resource shortages. In applied methodologies,

resilience is considered an integral category related to immunity to interference and cybersecurity, and should be suitable for operational application (Hrozdov *et al.*, 2024). Given the nature of the threats in the war in Ukraine, KPIs should be oriented towards the C2-minimum and managed degradation of services. The recommended set of KPIs for TCL includes: C2-minimum availability between key nodes, end-to-end (end-to-end) delay of delivery of commands and critical messages, latency variation for voice and streaming services, throughput by traffic class (SA/telemetry as priority data, high-bitrate streams, only if resource is available), PTT/group call setup time, recovery time after node/link loss, resilience to EW/cyber impacts, including taking into account combined threats for the satellite segment.

For practical verification of KPI, it is necessary to apply a simple protocol during exercises/deployments: define control pairs of nodes (company ↔ battalion, battalion ↔ brigade, adjacent companies), set traffic profiles (C2-minimum, SA/telemetry, flows "if resource is available") and take KPI in three modes: regular, with resource shortage (simulation of overload) and with interference/EW (simulation of degradation), fixing the sequence of disconnection of services according to the degradation policy. In the TCL, "range" should be interpreted as coverage determined by LOS/NLOS, terrain and buildings, antenna solutions, node density and EW level, respectively, coverage is the result of network organisation (planning, backup routes/frequencies, counteraction to interference), and not a constant passport value (Sholudko *et al.*, 2023). In order to effectively address these challenges in real-world combat environments, it is necessary to properly organise the network topology and technology selection based on signal propagation scenarios in different environments. Table 3 below shows typical signal propagation scenarios and corresponding network topologies for TCL, including changes under EW conditions, manoeuvres, infrastructure changes, and the need to adapt to various geographical conditions.

Table 3. Signal propagation scenarios and corresponding network topologies for TCL

Scenario	Signal propagation conditions	Technologies and network solutions	Degradation mode
City (NLOS, multipath)	"Radio shadows" and unstable links, multipath propagation	SDR-MANET priority for data, DMR for voice, self-healing mesh network, relay.	Cut off "heavy" data → only short messages → C2-minimum (voice)
Forest belts/plantations (partial NLOS)	Coverage fragmentation, reduced signal density over long distances	A denser network of nodes, repeaters, gateways, multichannel network with multiple transport carriers.	Shorter hops, higher node density, thoughtful placement of repeaters/gateways
Open terrain (LOS, EW vulnerability)	Better connectivity, but possible sharp drops under interference under EW conditions	SDR-MANET for data, SAT backhaul for backup, mode adaptation, route backup, readiness for transition to SAT.	Adaptation of modes, reservation of routes
Rough terrain (variable LOS)	Geometric "shadows" and gaps, changes in the line of communication	SDR-MANET for network self-organisation, fast rerouting, alternative routes, local autonomy of subnets.	Fast rerouting, alternative routes/gateways

Source: developed by the authors

All scenarios emphasise the importance of network adaptation to different signal propagation conditions, which makes it possible to flexibly and effectively manage tactical TCL networks in real combat conditions. The experience of combat operations in Ukraine has shown that the TCL communication network degrades under the combined influence of: high density of EW/interference, absence or destruction of infrastructure, rapid topology changes (manoeuvre, loss of nodes), difficult propagation conditions (NLOS), spectrum congestion and cyber risks (compromise of terminals, interception, attacks on gateways/endpoints). Under such conditions, static planning that depends on a single transport medium leads to a decrease in availability and an increase in recovery time, which directly affects the control cycle (company – battalion – brigade) and maintaining COP/data exchange (Halwa & Harriss, 2025). The engineering conclusion is the feasibility of a multilayer architecture in which the tactical data layer is implemented on the basis of a self-organising and self-healing MANET, and maintaining manageability is ensured by QoS prioritisation, managed degradation, and KPI-oriented stability verification.

In the conditions of war in Ukraine, threats to TCL communication are complex and include both electromagnetic and informational and cybernetic influences. The basic risks include interception and eavesdropping due to the use of unprotected or incorrectly configured channels, as well as jamming and electromagnetic incompatibility/mutual noise phenomena at high density of means in the common air. A separate group is made up of influences on navigation components, including spoofing, which complicate the use of modes and services dependent on navigation support. For the backbone component, cyber impacts on the satellite

segment and jamming of satellite terminals are critical, which increases the vulnerability of SAT backhaul and justifies the need for architectural redundancy. Given the above threats, the implementation of command and control (C2) exchange in the TCL should provide for cryptographic protection of traffic and correct organisation of key material at the radio network and terminal levels. As an example of an engineering implementation for tactical SDR-MANET nodes, the Silvus SC4400E specification provides support for DES56 (standard) and AES256 (optional), the presence of the zeroize function and declared compliance with the National Institute of Standards and Technology (2019) as a characteristic of the cryptomodule level. In combat conditions, taking into account the probability of loss/capture of nodes, the availability of procedures for the operational destruction of cryptomaterial and minimising the consequences of potential compromise of a node for the entire network are critical.

Based on the real combat experience of using tactical communication in Ukraine and technological capabilities, the key requirements for the TCL network in conditions of mobility and limited infrastructure were summarised:

1. Mobility and limited infrastructure. The TCL network must support the mobility of units and function in conditions of limited infrastructure capabilities. This includes adaptation to rapid topology changes and the availability of backup channels for communication.

2. Digitalisation and networking. The need for digital means of communication has become urgent, providing not only voice transmission, but also support for various data and integration with other systems, including automation of unit management processes.

3. C2 availability and recovery time. This shifts emphasis away from "passport range" to the availability of minimum C2, in particular voice communication and short messages, during resource shortages or in conditions of electromagnetic effects. Rapid restoration of communication after the loss of nodes or channels.

4. Comprehensive IS. Ensuring network IS, including traffic encryption, key management, domain segmentation, access control, as well as implementing service minimisation policies and operational destruction of cryptographic materials (zeroize) in case of compromise.

5. Adaptation to electromagnetic interference (EW). The network must be resistant to EW and other interference that can disrupt or degrade data transmission. This includes the use of self-organising and self-healing technologies, such as SDR-MANET.

6. Protection against interception and cyber risks. Ensuring protection against interception and compromise of data, especially in tactical networks, where there may be attacks on endpoints and gateways.

7. Flexibility of network topology. Considering various signal propagation scenarios in different conditions (city, forest, open areas), the network must be flexible, with the ability to adapt to topology changes. The important things are the thoughtful placement of repeaters and gateways, support for multichannel networks, and adaptation to EW conditions.

8. Scalability and manageability. The network must have the ability to scale under conditions of increased load and ensure communication stability even with partial loss of nodes or changing conditions.

These requirements determined the basis for the development of an algorithm and engineering model of a communication network for the TCL, which must provide high adaptability, stability, and security in difficult conditions. In general, building a TCL network and planning its deployment requires not only adaptation to changing operating conditions, but also the integration of modern technologies capable of ensuring stability, mobility, and communication security in combat conditions. The proposed criteria and principles, in particular the use of hybrid technologies and controlled degradation of services, are key to ensuring the effective operation of the TCL network in difficult EW conditions, limited resources, and high cyber threats.

Algorithm for designing and configuring a TCL network under conditions of EW and electromagnetic effects

The "DMR + SDR(MIMO)-MANET + LEO-SAT backhaul" model presented in the current study takes into account all the above requirements and is focused on ensuring C2-minimum and maintaining controllability under conditions of EW, node losses and degradation of individual channels. Such a system can help ensure uninterrupted control in complex combat situations. The practical implementation of the proposed model requires a formalised sequence of engineering solutions that connects the choice of network architecture with the procedures for its planning, configuration and verification. For this purpose, a step-by-step algorithm for designing and configuring a TCL communication network under conditions of limited resources and electromagnetic effects is proposed. The algorithm covers input data collection, optimisation criteria definition, C2-minimum and data layer design, gateway and relay node selection, QoS and managed degradation mode configuration, SAT integration, IS implementation, and iterative KPI verification with subsequent network parameter adjustment. A generalised algorithm diagram is shown in Figure 2.

Figure 2 should be interpreted as a detailed algorithm for the phased design, configuration, verification and adjustment of the TCL communication network in the "company – battalion – brigade" loop under EW conditions. The algorithm is focused on ensuring the functional suitability of the network for performing management tasks. The ultimate criterion for its effectiveness is not the maximisation of total throughput, but the guaranteed provision of C2-minimum, communication stability, controlled degradation of services and network recoverability after the loss of nodes or channels.

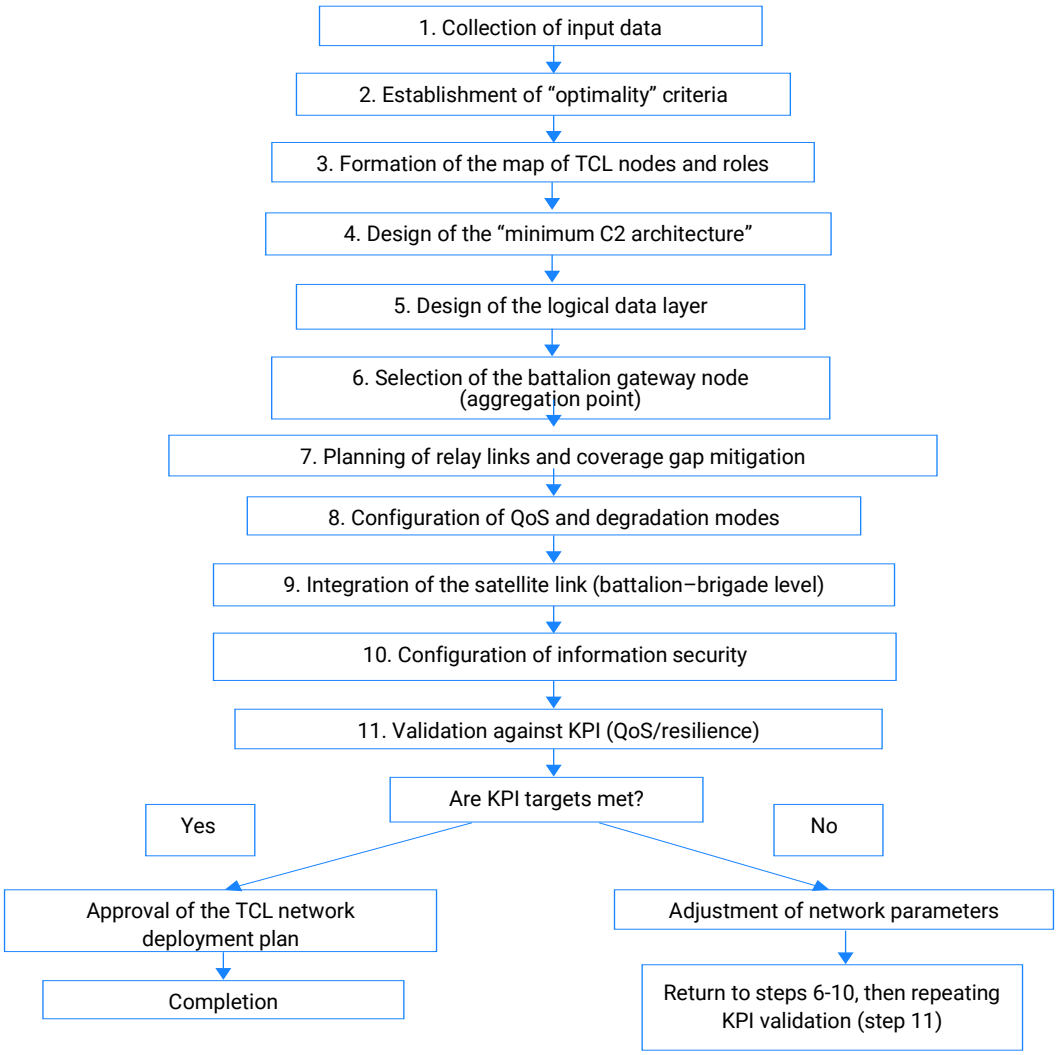

Figure 2. Generalised scheme of the algorithm for designing and configuring a TCL communication network under EW conditions

Source: compiled by the authors based on National Institute of Standards and Technology (2001; 2019), European Telecommunications Standards Institute (2016), O. Lavrut *et al*. (2019; 2021), J. Suess (2022), V.H. Sholudko *et al*. (2023), A.A. Hrozdov *et al*. (2024), P.V. Khomenko *et al*. (2025), Silvus Technologies (2025), S. Halwa & L. Harriss (2025), Starlink (2026)

In step 1 – input data is collected, which determine the initial conditions for building the network. At this stage, the features of the area of operations are analysed, in particular the terrain, the nature of the development, the presence of radio shadow zones, expected radio conditions and probable spatial restrictions for the placement of nodes. Additionally, the composition and number of network elements at the company, battalion, and brigade levels are determined, the presence or need for relaying is assessed, and a list of services that must be supported by the network is formed: command voice exchange, transmission of commands and coordinates, situational awareness data, telemetry and, if resources are available, streaming data. At the same time, EW threats, cyber threats, risks of equipment loss, as well as the probability of degradation or loss of the satellite channel are taken into account. It is at this step that the available technological set of the network is fixed: DMR as the basis for stable voice C2 exchange, SDR-MANET as the basis for a self-organised data layer, and the satellite channel as a main or backup means of communication between the battalion and brigade levels.

In step 2 – the criteria are established according to which the network is considered suitable and meets the requirements for performing the task. At this stage, the network requirements are formalised through coverage, connectivity, survivability, QoS, electromagnetic compatibility, and IS indicators. The priority criterion is the guaranteed provision of the C2-minimum even with partial degradation of services, and lower priority information flows may be limited. Also, requirements are set here for the minimum permissible availability of the command channel, permissible delays, route stability, speed of recovery after node loss, as well as for radio discipline, radiation control, and cryptographic protection modes.

Step 3 – a map of TCL nodes is formed and the roles in the network are determined. At the company level, subscriber nodes and data access nodes are set; at the battalion level, aggregation and gating nodes are allocated; at the brigade level, a control node and an exit point to the main or backup channels are determined. This allows, even before configuring individual technologies, to distribute functions between network levels: local C2 exchange, data aggregation, routing between units, integration with external channels, and redundancy of critical functions. The criticality of this stage is that an error in determining the roles of nodes subsequently leads to inefficient routing, overloading of network reference points, and loss of controllability during degradation.

In step 4 – the company-level C2-minimum circuit is designed. At this stage, DMR technology is introduced, which is used as the basic and most stable channel for voice command communication and transmission of short critical messages. For the company level, channel resources, frequency plan, backup frequencies, operating modes, and the order of transition between these modes are determined in accordance with communication management procedures. This step directly implements the network requirement for continuous management, even under deteriorating propagation conditions or under the influence of EW. If risks of unstable voice coverage are identified at the initial planning stage, this is recorded as a basis for further adjustment of node placement or introduction of relaying.

In step 5 – the company data layer is designed, within which SDR-MANET is introduced as a self-organised multi-hop network. At this stage, the logic of building routes between nodes, the permissible depth of multi-hop connections, the procedure for self-recovery of routes after the loss of individual nodes, and the rules for servicing situational awareness, telemetry, and other service packages are determined. Using SDR-MANET allows meeting the requirements for topology flexibility, distributed control, and adaptability to changes in the tactical situation. The criticality of this step is that it is here that the network's ability to support local data exchange in the event of losses, manoeuvres, and destruction of individual communication sections is formed.

Step 6 – the battalion gateway node is selected, which acts as a point of traffic aggregation and inter-level exchange management. At this stage, one or two reference nodes are determined, for which the position, coverage area, security, power supply, antenna deployment, and redundancy are assessed. This is where the network transitions from local company segments to an integrated battalion structure. The gateway node concentrates the functions of traffic aggregation, message prioritisation, route management, switching between available channels, and preparing data for transmission to a higher level. This is one of the most critical stages of the algorithm, since an incorrectly selected gateway can become a point of overload or the only vulnerability of the entire network.

In step 7 – relay planning and filling of areas of insufficient coverage are carried out. If the previous stages have established the presence of radio shadows, interruption of connectivity between companies, instability of access to the battalion gateway, or insufficient depth of coverage, additional relay nodes are introduced. These can be stationary mast solutions, mobile platforms, or UAV repeaters, depending on the terrain conditions and available resources. In this step, the algorithm directly responds to the network requirement to ensure connectivity and survivability in difficult terrain or with partial loss of infrastructure. If connectivity remains insufficient after the introduction of relaying, this stage is the primary point of correction before re-checking the KPI.

In step 8 – QoS and service degradation modes are configured. This is the stage at which the algorithm establishes a correspondence between traffic types and network functional priorities. The highest priority is given to the C2-minimum: command voice, commands, coordinates, and other critical management messages. The second level of priority is given to situational awareness and telemetry data. Streaming services, additional information exchanges or volumetric data are of lower priority and can be limited, compressed or completely disconnected in the event of a shortage of radio resources or during EW. It is at this step that the mechanism of controlled network degradation is formed, thanks to which the controllability of the unit is maintained even under deteriorating operating conditions.

In step 9 – the satellite channel is integrated into the "battalion-brigade" loop. Here, the satellite segment is not considered as a permanently guaranteed resource, but as a managed backbone or backup channel with predefined usage rules. At this stage, the conditions under which the satellite link is used as the main one for inter-level exchange are established, as well as scenarios for transition to autonomous battalion-level operation in the event of its loss, jamming, or cyber impact. Therefore, this step directly implements the requirement for the network to provide redundancy for inter-level exchange and ensure operability even in the absence of an external backbone channel.

Step 10 – IS setup. At this stage, encryption mechanisms, key material management, delimitation of exchange domains, as well as procedures for rapid resetting of keys and critical settings in the event of a threat of equipment capture are implemented for all involved technological segments. For cryptographic protection, the algorithm uses AES and other provided protection means compatible with the network configuration. Measures to counter radio interception, technical analysis of emissions and compromise of network parameters are separately determined. This means that IS requirements are not considered as an additional component, but are included in the network setup algorithm itself as its mandatory phase.

Step 11 is a network check using KPIs. The availability of C2-minimum, latency and stability of critical message transmission, data throughput, self-healing speed after loss of nodes or channels, as well as the network behaviour under the influence of EW are assessed. Verification at this stage completes the full design cycle and puts the

algorithm into decision-making mode. If the KPIs are met, the network deployment plan is approved as meeting the functional requirements. If the KPIs are not met, the algorithm proceeds to the adjustment of the network parameters, returning first to steps 6-10, i.e., reviewing the gateway configuration, relaying, QoS policies, degradation modes, satellite channel reservation, and protection parameters. This cycle is repeated until an acceptable level of network stability, QoS, and manageability is achieved.

The presented algorithm for phased design, configuration, and verification of the TCL network in EW conditions provides a structured approach to creating and optimising tactical networks. Its main advantage is the integration of many technologies into a single hybrid architecture that includes DMR, SDR-MANET, and satellite channels, which allows not only to ensure communication stability in conditions of high interference and dynamic topology, but also to quickly adapt the network to changes in combat conditions. This algorithm significantly improves network planning and deployment, as it is focused on real-world operating conditions. In particular, it allows clearly defining priorities for each network layer and technology, ensuring optimal resource allocation and minimising the likelihood of failures or degradation in critical situations. The algorithm also implements backup, degradation, and recovery mechanisms that allow the network to remain operational even in the event of loss of nodes or channels. In addition, it includes KPI verification stages that allow directly assessing the effectiveness and reliability of the network under combat and EW conditions, which is important for increasing the overall stability of the system. Thus, the algorithm not only structurally organises the process of network design and configuration, but also allows actively responding to changes in operating conditions, ensuring flexibility, stability, and communication security at all levels of command.

The correctness and efficiency of the proposed model largely depend on the algorithm for its design and configuration. Therefore, it is important to assess how the architectural principles embedded in the algorithm meet the requirements for the TCL network in EW conditions. Although it is not possible to make a direct comparison with other similar algorithms, it is possible to compare the architectural approaches and technologies used in the proposed model. This allows analysing how the choice of technologies, such as DMR, SDR-MANET and satellite channels, affects the stability, adaptability, and efficiency of the network in real conditions.

The results of the engineering analysis of the TCL network architecture showed that it is correctly described not through the "best channel", but through a matrix of services, roles of nodes and carriers and a formalised degradation sequence (ISR/video → COP/SA intensity reduction → "control only" with the dominance of the C2-minimum). This statement is consistent with D. Darsena & F. Verde (2022), who emphasised that the standardised Mission-Critical Push-To-Talk (MCPTT)/Mission-Critical Video (MCVideo)/MCData classes are designed under strict requirements for availability, reliability, latency, security and QoS, i.e., the "criticality" of services sets the priorities of the network resource. At the same time, the authors consider this hierarchy mainly in the context of the 4G/5G ecosystem and standardisation, while in the current study it is translated into operational degradation rules for a heterogeneous stack of communication carriers (DMR/MANET/SAT) and tied to the roles of TCL nodes (subscriber/gateway/aggregator/SAT gateway). S. Yuan *et al*. (2023) show that integrated satellite-ground networks are characterised by limitations in flexibility and adaptability and problems with efficient resource use, in connection with which the authors substantiated the need for SDN and intelligent control approaches. In the current study, these provisions are interpreted for the battalion-brigade level as an engineering requirement to treat SAT backhaul not as an unconditional support of the network, but as a managed resource with traffic prioritisation and degradation modes to the minimum profile ("C2/service data only") and with reservation/switching procedures, which is reflected in the service and carrier distribution matrix.

The results of the engineering analysis of the criteria for selecting the base data layer in the TCL network showed that the use of SDR-MANET at the battalion aggregator level and the inclusion of KPIs of recovery time after the loss of a node or link are methodologically justified, since in TCL the manageability is determined not by the peak speed, but by the network's ability to quickly restore connectivity and delivery of critical messages in conditions of dynamic topology and losses. These conclusions correlate with M. Baumgartner *et al*. (2024), who considered approaches to increasing the resilience of routing in MANETs and showed that the proposed improvements to routing protocols increase the network's resilience in conditions of dynamic topology, which is manifested in improving deliverability and latency indicators. The discrepancy lies in the level of detail: the authors worked at the level of specific protocols/mechanisms, while the current study forms an architectural policy and KPI framework, without fixing specific implementations of routing protocols. The proposed approach to service distribution and degradation policy in this study requires further experimental verification on specific MANET implementations regarding the reachability of the KPI of connectivity recovery time under typical COP/telemetry loads.

D. Falcão *et al*. (2021) substantiate the feasibility of disruption-tolerant networking (DTN) approaches for tactical messaging under conditions of discontinuous connectivity: DTN is considered as an evolution of ad hoc networks for environments with low node density and intermittent connections, where continuous end-to-end (end-to-end) connectivity is absent, and delivery is ensured by buffering and deferred forwarding. In the current study, these results are interpreted for degraded scenarios of the backbone transport (strategic backhaul) in the TCL as an engineering requirement to provide a separate "store and forward" degradation profile for critical messages, which is reflected

in the service and carrier distribution matrix. At the same time, despite the difference in scenarios (marine context in the authors vs. terrestrial TCL in this study), the basic principle is transferred: in the case of backhaul instability, the guaranteed delivery of management messages is ensured by the exchange mode (buffering/prioritisation/TTL), and not by the assumption of constant channel availability. J. Suomalainen *et al*. (2024) analysed the cybersecurity of autonomous rapidly deployable tactical networks and showed that orchestration and autonomous control (in particular, with the use of machine learning methods) simultaneously increase the complexity of the system and expand the threat landscape, as a result of which risk-based approaches, including threat analysis and prioritisation, are necessary for mission-critical applications. This conclusion correlates with the results of the current study that even in the minimum degradation mode, basic IS mechanisms must be maintained, the reduction of services should not be accompanied by the rejection of segmentation, access control and protection of control nodes/gateways and endpoints, which is reflected in the service and carrier distribution matrix. Therefore, IS is a component of system resilience and should be specified through the role of the node and the mode of operation, including degradation modes. Thus, the TCL communication network should be designed as a hybrid multilayer system with primary and backup communication media and managed degradation of services with a C2-minimum priority. Under any degradation profile, it is necessary to preserve the basic IS mechanisms (encryption, key management/zeroize, domain segmentation) and ensure architectural stability through channel diversification.

Conclusions

The paper analyses the model of the tactical command link communication network for the "company – battalion – brigade" loop. The model architecture has three levels: company, battalion and brigade, and includes access nodes, aggregation and gateway nodes, as well as integration and access nodes to main or backup channels. The model was considered under the conditions of unit mobility, limited infrastructure, EW, and possible losses of nodes and communication channels. The study analyses tactical communication technologies, determines the key characteristics for building TCL networks, justifies the distribution of services between network levels and the roles of nodes, taking into account TCL application scenarios. The feasibility of using DMR to ensure C2-minimum, SDR-MANET to build an adaptive data layer, and a satellite channel for inter-level main or backup exchange is shown. Based on real combat experience, the requirements for the TCL network were formulated: resistance to EW, survivability, adaptability, rapid recovery, support for priority services and IS.

The main result of the work was the development of an algorithm for planning the TCL network. It is a step-by-step procedure for deploying, configuring, testing and adjusting the network in changing operating conditions. The algorithm covers the collection of input data, determining performance criteria – coverage, connectivity, survivability and security, distributing node roles, configuring the C2-minimum and data layer, selecting gateway nodes and relay facilities, configuring QoS and redundancy, prioritising traffic and managed service degradation modes. The effectiveness is assessed through KPIs, which allows checking the stability and adaptability of the network in real combat conditions. The results of the study showed that the choice of technologies and architectural approaches, such as DMR, SDR-MANET and satellite channels, significantly improve the efficiency and stability of the TCL network. Ensuring manageability, stability, and high throughput even under EW conditions and channel degradation was one of the key aspects when building such a system. The proposed hybrid architecture allows the network to quickly adapt to changes in topology and conditions, which increases its reliability and security at all levels of command.

The limitation of the study is its theoretical and conceptual nature and the lack of full-scale experimental/field verification of the proposed solutions under real EW and mobility conditions. Future research should focus on developing and improving methods for protecting the TCL network from active interference and attacks. In addition, additional attention is required to study dynamic routing algorithms in conditions of limited resources and variable topology, in particular with the use of artificial intelligence and machine learning methods.

Acknowledgements

None.

Funding

The study was not funded.

Conflict of Interest

None.

References

[1] Baumgartner, M., Papaj, J., Kurkina, N., Dobos, L., & Cizmar, A. (2024). Resilient enhancements of routing protocols in MANET. *Peer-to-Peer Networking and Applications*, 17, 3200-3221. doi: 10.1007/s12083-024-01746-3.

[2] Bojor, L., Petrache, T., & Cristescu, C. (2024). Emerging technologies in conflict: The impact of Starlink in the Russia-Ukraine war. *Land Forces Academy Review*, 29(2), 185-194. doi: 10.2478/raft-2024-0020.

[3] Chen, N., Song, Y., Cao, Y., Sun, Z., Zhao, B., Wang, M., He, D., & Peng, G. (2025). Network-layer perspectives on satellite-terrestrial integrated networks in 6G: A comprehensive review. *Engineering*, 54, 69-92. doi: 10.1016/j.eng.2025.05.012.

[4] Darsena, D., & Verde, F. (2022). Anti-jamming beam alignment in millimeter-wave MIMO systems. *arXiv*. doi: 10.48550/arXiv.2110.08134.

[5] European Telecommunications Standards Institute. (2016). *Electromagnetic compatibility and Radio spectrum Matters (ERM); Digital Mobile Radio (DMR) Systems; Part 2: DMR voice and generic services and facilities*. Retrieved from https://www.etsi.org/deliver/etsi_ts/102300_102399/10236102/02.03.01_60/ts_10236102v020301p.pdf.

[6] European Telecommunications Standards Institute. (2023). *Digital Mobile Radio (DMR) Systems; Part 1: DMR Air Interface (AI) protocol (ETSI TS 102 361-1 V2.6.1)*. Retrieved from https://www.dmrassociation.org/public-downloads/standards/ts_10236101v020601p.pdf.

[7] Falcão, D., Salles, R., & Maranhão, P. (2021). Performance evaluation of disruption tolerant networks on warships' tactical messages for secure transmissions. *Journal of Communications and Networks*, 23(6), 473-487. doi: 10.23919/JCN.2021.000043.

[8] Halwa, S., & Harriss, L. (2025). *Electromagnetic (electronic) warfare*. Retrieved from https://researchbriefings.files.parliament.uk/documents/POST-PN-0749/POST-PN-0749.pdf.

[9] Hrozdov, A.A., Zinchenko, I.A., Hromliuk, M.M., Bilyi, O.A., Ivchenko, M.M., & Tsymbal, I.V. (2024). Method for assessing the sustainability of a military communication system based on troops' combat capabilities. *Systems and Technologies of Communication, Informatization and Cybersecurity*, 5, 64-70. doi: 10.58254/viti.5.2024.05.64.

[10] Hui, K.-P., Phillips, D., Kekirigoda, A., Allwright, A., Zhang, J.A., Zhang, H., Le, A.T., & Jayawickrama, B.A. (2024). Unveiling MIMO potential: A prototype for enhanced tactical communications with interference suppression. In *Proceedings of the IEEE military communications conference* (pp. 1-6). Washington: IEEE. doi: 10.1109/MILCOM61039.2024.10773940.

[11] Hytera. (2018). *PD7i series*. Retrieved from https://www.hytera.us/wp-content/uploads/2023/01/PD7i-Series_20190524-Web.pdf.

[12] Kang, M., Park, S., & Lee, Y. (2024). A survey on satellite communication system security. *Sensors*, 24(9), article number 2897. doi: 10.3390/s24092897.

[13] Kantheti, V.S., Lin, C.-H., Lin, S.-C., & Chu, L.C. (2023). Anti-jamming resilient LEO satellite swarms. In *Proceedings of the military communications conference (MILCOM): Workshop on 5G military communications* (pp. 77-82). Boston: IEEE. doi: 10.1109/MILCOM58377.2023.10356296.

[14] Khomenko, P.V., Radzivilov, H.D., & Ilinov, M.D. (2025). Analysis of the functionality of MANET tactical radio systems. *Systems and Technologies of Communication, Informatization and Cybersecurity*, 7, 222-231. doi: 10.58254/viti.7.2025.20.222.

[15] Kim, J., Biswas, P.K., Bohacek, S., Mackey, S.J., Samoohi, S., & Patel, M.P. (2021). Advanced protocols for the mitigation of friendly jamming in mobile ad-hoc networks. *Journal of Network and Computer Applications*, 181, article number 103037. doi: 10.1016/j.jnca.2021.103037.

[16] L3Harris Technologies, Inc. (2024). *2023 annual report*. Retrieved from https://www.l3harris.com/sites/default/files/2024-02/L3Harris_2023-Annual-Report_web_.pdf.

[17] Lavrut, O., Davidenko, S., Opalynskyi, V., Boichuk, B., & Oliinyk, S. (2021). *Harris: Digital communication means of the tactical command and control of the Armed Forces of Ukraine: Study guide*. Lviv: National Army Academy named after Hetman Petro Sahaidachnyi.

[18] Lavrut, O.O., Lavrut, T.V., Klimovych, O.K., & Zdorenko, Y.M. (2019). New technologies and means of communication in the Armed Forces of Ukraine: The path of transformation and development prospects. *Science and Technology of the Air Force of Ukraine*, 34(1), 91-101. doi: 10.30748/nitps.2019.34.13.

[19] Lo, Y.W., Tsoi, M.H., Chow, C.-F., & Mung, S.W. (2024). An NB-IoT monitoring system for digital mobile radio with industrial IoT performance and reliability evaluation. *IEEE Sensors Journal*, 25(3), 5337-5348. doi: 10.1109/JSEN.2024.3512859.

[20] Mahmud, R., Toosi, A.N., Rodriguez, M.A., Madanapalli, S.C., Sivaraman, V., Sciacca, L., Sioutis, C., & Buyya, R. (2021). Software-defined multi-domain tactical networks: Foundations and future directions. In A. Mukherjee, D. De, S.K. Ghosh & R. Buyya (Eds.), *Mobile edge computing* (pp. 183-227). Cham: Springer. doi: 10.1007/978-3-030-69893-5_9.

[21] Masesov, M., Krotov, V., & Openko, P. (2021). Active queue management in tactical radio networks using fuzzy logic. *Modern Information Technologies in the Field of Security and Defense*, 40(1), 37-46. doi: 10.33099/2311-7249/2021-40-1-37-46.

[22] McCrory, D. (2023). *Electronic warfare in Ukraine: Preliminary lessons for NATO air power capability development*. Retrieved from https://www.japcc.org/articles/electronic-warfare-in-ukraine/.

[23] Motorola Solutions. (2026). *MOTOTRBO™ DM4000e series: Mobile two-way radios*. Retrieved from https://www.motorolasolutions.com/content/dam/msi/docs/EA_Collaterals/ENGLISH/MOTOTRBO/Mobiles/dm4000e_datasheet_eng.pdf.

[24] Mureșan, A., & Bechet, P. (2024). Waveform analysis in integrated tactical radio systems. *Land Forces Academy Review*, 29(4), 584-595. doi: 10.2478/raft-2024-0060.

[25] National Institute of Standards and Technology. (2001). *Advanced Encryption Standard (AES) (FIPS PUB 197)*. Retrieved from https://csrc.nist.gov/files/pubs/fips/197/final/docs/fips-197.pdf.

[26] National Institute of Standards and Technology. (2019). *Security requirements for cryptographic modules (FIPS PUB 140-3)*. Retrieved from https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.140-3.pdf.

[27] Patel, Z., Khanpara, P., Valiveti, S., & Raval, G. (2023). The evolution of ad hoc networks for tactical military communications: Trends, technologies, and case studies. In S. Shakya, V. Balas & W. Haoxiang (Eds.), *Proceedings of the 3rd international conference on sustainable expert systems* (pp. 331-346). Singapore: Springer. doi: 10.1007/978-981-19-7874-6_24.

[28] Salnyk, S.V., & Sydorkin, P.H. (2024). Analysis of the use of programmable radio communication means in mobile radio networks. *Systems of Arms and Military Equipment*, 77(1), 110-116. doi: 10.30748/soivt.2024.77.16.

[29] Sholudko, V.H., Yesaulov, M.Yu., Vakulenko, O.V., Hurskyi, T.H., & Fomin, M.M. (2023). *Organization of military communications: Study guide*. Kyiv: Skif Publishing House.

[30] Shtonda, R., Zinchenko, M., & Chayka, Y. (2023). Application of small-sized digital tropospheric communication stations during combat operations. *Modern Information Technologies in the Field of Security and Defense*, 47(2), 25-30. doi: 10.33099/2311-7249/2023-47-2-25-30.

[31] Silvus Technologies. (2025). *StreamCaster® 4400 Enhanced: SC4400E (4x4 MIMO radio)*. Retrieved from https://silvustechnologies.com/wp-content/uploads/2025/12/StreamCaster-4400-SC4400E-Enhanced-Datasheet.pdf.

[32] Starlink. (2026). *Starlink specifications*. Retrieved from https://starlink.com/legal/documents/DOC-1723-29826-76.

[33] Suess, J. (2022). *Jamming and cyber attacks: How space is being targeted in Ukraine*. Retrieved from https://www.rusi.org/explore-our-research/publications/commentary/jamming-and-cyber-attacks-how-space-being-targeted-ukraine.

[34] Suomalainen, J., Ahmad, I., Shajan, A., & Savunen, T. (2024). Cybersecurity for tactical 6G networks: Threats, architecture, and intelligence. *Future Generation Computer Systems*, 162, article number 107500. doi: 10.1016/j.future.2024.107500.

[35] Suomalainen, J., Julku, J., Heikkinen, A., Rantala, S.J., & Yastrebova, A. (2022). Security-driven prioritization for tactical mobile networks. *Journal of Information Security and Applications*, 67, article number 103198. doi: 10.1016/j.jisa.2022.103198.

[36] Thornton, C.E., Allen, E., Jones, E., Jakubisin, D., Templin, F., & Liu, L. (2023). On the role of 5G and beyond sidelink communication in multi-hop tactical networks. *arXiv*. doi: 10.48550/arXiv.2309.16628.

[37] Watling, J., & Reynolds, N. (2023). *Meatgrinder: Russian tactics in the second year of its invasion of Ukraine*. Retrieved from https://static.rusi.org/403-SR-Russian-Tactics-web-final.pdf.

[38] Yuan, S., Peng, M., Sun, Y., & Liu, X. (2023). Software defined intelligent satellite-terrestrial integrated networks: Insights and challenges. *Digital Communications and Networks*, 9(6), 1331-1339. doi: 10.1016/j.dcan.2022.06.009.

Аналіз побудови мережі зв'язку тактичної ланки управління на основі програмно керованих засобів радіозв'язку

Григорій Радзівілов
Кандидат технічних наук, доцент
Військовий інститут телекомунікацій та інформатизації імені Героїв Крут
01011, вул. Князів Острозьких, 45/1, м. Київ, Україна
https://orcid.org/0000-0002-6047-1897

Дмитро Павлюк
Ад'юнкт
Військовий інститут телекомунікацій та інформатизації імені Героїв Крут
01011, вул. Князів Острозьких, 45/1, м. Київ, Україна
https://orcid.org/0000-0001-8461-3899

Анотація. Метою дослідження було розроблення архітектурного рішення щодо побудови та керування тактичним контуром зв'язку рівнів рота – батальйон – бригада на базі гнучких радіоплатформ для забезпечення безперервності та надійності комунікацій в умовах активної протидії противника. Були використані методи структурно-функціонального моделювання, сценарного та порівняльного аналізу. Встановлено, що в умовах радіоелектронної боротьби (РЕБ) контур зв'язку рівнів рота – батальйон – бригада слід будувати як гібридну багаторівневу архітектуру, яка на рівні роти поєднує окремі канали голосового зв'язку та передавання даних, використовує для тактичного обміну самоорганізовану мережу Mobile Ad Hoc Network на основі програмно-керованих радіосистем, а на батальйонному – вузол-шлюз для агрегації трафіку, маршрутизації та інтеграції з каналами зв'язку вищого рівня через Low Earth Orbit satellite backhaul. Показано, що Digital Mobile Radio необхідно використовувати для голосового контуру command and control-мінімуму, Software-Defined Radio Mobile Ad Hoc Network із Multiple-Input Multiple-Output – для тактичного шару передавання даних, а LEO satellite backhaul – як магістральний або резервний канал зв'язку між рівнями батальйон – бригада. Запропоновано ітераційний алгоритм планування та налаштування мережі зв'язку тактичної ланки управління (ТЛУ) при наявності електромагнітних перешкод та обмежених ресурсів. Перевагами моделі є підвищення доступності мінімуму командування та управління, збереження керованості, пріоритезація трафіку за якістю обслуговування та керована деградація сервісів. Її ефективність визначалася безперервністю голосового зв'язку, стійкістю обміну даними, швидкістю відновлення зв'язності та резервуванням зв'язку між рівнями рота – батальйон – бригада. Практична значимість полягає у можливості застосування результатів фахівцями підрозділів зв'язку під час планування, розгортання та налаштування тактичного контуру зв'язку рота – батальйон – бригада у польових умовах за складної електромагнітної обстановки та активної протидії

Ключові слова: мережа зв'язку; самоорганізована мережа; багатоканальна архітектура; вузол-шлюз; супутниковий канал зв'язку; радіоелектронна боротьба

Journal homepage: https://itce.vn.ua/en

Information Technologies and Computer Engineering **Vol. 23, No. 1. 2026**

Article's History: Received: 06.11.2025 Revised: 12.02.2026
Accepted: 26.03.2026 Published: 20.04.2026

ISSN 1999-9941; e-ISSN 2078-6387
UDC 004.4:004.5:622.3
DOI: 10.31649/vitce/1.2026.170

Application of deep learning methods to image processing and enhancement: A case study on seismic data

Ruslan Malikov*
Postgraduate
Institute of Geology and Geophysics of Azerbaijan National Academy of Sciences
AZ1143, 119 H. Cavid Ave., Baku, Azerbaijan
https://orcid.org/0009-0005-2126-1642

Abstract. The aim of this study was to evaluate the effectiveness of a modified encoder-decoder neural network architecture for denoising and image enhancement using synthetic and real data. The research methodology was based on a computational experiment and included training the model on synthetic images, quantitatively comparing the obtained results with the f-x deconvolution method and an alternative convolutional denoising model, and testing the robustness on real data with the presence of various noise characteristics. It was found that the applied denoising technique was characterised not only by reducing the noise component but also by preserving spatially significant image characteristics, including sharpness of edges, local transitions, morphology, and the relative positions of structural elements without signs of excessive smearing. A final comparison of the methods on synthetic test images showed that the average signal-to-noise ratio, peak signal-to-noise ratio, and multiscale structural similarity index for the proposed approach were 45.9 dB, 29.7 dB, and 0.99, respectively. For the f-x deconvolution method, the corresponding values were 31.5 dB, 23.9 dB, and 0.94, while for the alternative convolutional noise reduction model, the values were 20.9 dB, 18.4 dB, and 0.86. When applied to real data, the same enhancement behaviour was preserved, including the removal of pronounced noise contamination and derivation of a relatively clean signal without signal distortion. Depending on the input features, the method was accompanied by a decrease in intense noise masking, a reduction in residual noise, while maintaining a distinguishable signal structure, and reconstruction under conditions of a more complex spatial organisation of interference. Spectral analysis revealed a reduction in noise energy without disrupting the spectral configuration in the informative frequency range. The practical significance lies in the potential application of the proposed approach as a computational method for processing noisy images in systems designed for noise reduction and restoration of various data structures

Keywords: image restoration; image denoising; structural preservation; seismic image reconstruction; supervised learning

Introduction

In image restoration research, improving the quality of visual data was considered not only as a noise reduction task, but also as a broader problem of reconstructing image structure under various types of degradation. This approach stemmed from the fact that in real-world visual information processing scenarios, distortions were rarely limited to an isolated noise pattern, but were accompanied by loss of contours, weakening of textures, blurring of local intensity transitions, and deformation of subtle details. For this reason, image restoration had emerged as an independent field of computer vision, where the quality of the result was determined not only by the degree of noise suppression, but also by the ability of the model to preserve the spatial organisation of the image. B. Goyal *et al.* (2020) demonstrated that the transition from classical noise reduction methods to trainable models altered the approach to reproducing structural image elements. The authors emphasised that modern methods of visual data restoration were based not on local smoothing, but on modelling the spatial relationships between the degraded and target images. Further development of this area was driven by deep learning, specifically, convolutional neural networks. C. Tian *et al.* (2020) showed that such models formed a new stage in the development of noise reduction methods, providing a transition from manual feature construction to automatic learning of multi-level image representations. Within the framework of this approach, noise reduction was considered to be not only as a reduction of the noise component, but also as a restoration of visual structure, in which the result depended on the ability of the model to combine local analysis of pixel connections with a broader context to preserve edges, textures, and low-contrast

Suggested Citation:
Malikov, R. (2026). Application of deep learning methods to image processing and enhancement: A case study on seismic data. *Information Technologies and Computer Engineering*, 23(1), 170-182. doi: 10.31649/vitce/1.2026.170

*Corresponding author

details. A similar direction of generalisation was presented in the work of J. Mao *et al.* (2025), where digital noise reduction methods were considered as a consistent transition from filtering schemes to architectures oriented towards image reconstruction under conditions of complex degradation. Scientists showed that modern models were most effective in cases, where noise reduction was combined with the preservation of structural similarity between the reconstructed and original images.

In image restoration architectures, encoder-decoder models have become widespread. These models combined multi-level feature extraction with spatial restoration and were used in problems where, along with noise suppression, the reconstruction of hidden or weakened structural elements was required. In the study by Y. Cui *et al.* (2024), it was shown that updated convolutional networks for image restoration worked with spatially detailed representations and reproduced informative elements without pronounced smoothing. Researchers noted that the preservation of structure in such models was considered an independent condition for the quality of reconstruction. A similar aspect was considered in the work by N. Nazir *et al*. (2024), where it was shown that deep learning systems were evaluated not only by the degree of signal purification, but also by the ability to preserve diagnostically significant edges, density transitions, and local morphological details. This indicated that for various types of images, the methodological task lain in combining noise suppression with the preservation and restoration of image structure. Further development in this direction was associated with residual learning, diffusion and generative models. In the work of B. Xia *et al.* (2023), the diffusion model was considered as a tool for image restoration with various types of distortions. The authors showed that such an architecture ensured a consistent refinement of the visual structure during the reconstruction process. In the study of Z. Luo *et al.* (2025), diffusion reconstruction was presented as an independent modern direction of image restoration, in which the reconstruction process was associated with a step-by-step cleaning and stabilisation of the structural features of the image. In the work of Y.N. Imamverdiyev & F.I. Musayeva (2022), the potential of adversarial approaches for the formation of realistic visual representations was analysed. Scientists noted that such models were focused not only on eliminating defects, but also on reproducing a visually consistent image structure. The expansion of this research framework can be seen in the applied areas of processing specialised visual data.

In the work of S. Azizova *et al.* (2026), it was shown that one of the quality criteria for image restoration remained the correct reproduction of its structural and colour consistency. In this logic, seismic image processing was a special case of applying restoration models to data with a complex internal structure, where not only noise reduction was essential, but also the preservation of the continuity of reflections, weak signals, and the geometric consistency of useful structures. A study by M. Ding *et al.* (2024) demonstrated that the Swin Transformer, Convolutional Neural Network, U-Net (Swin-Conv-UNet) architecture was applicable to seismic denoising by combining deep feature extraction with the reconstruction of spatial details. The analysis established that seismic data can be considered an example for testing general approaches to modern image restoration on structurally complex material. A summary of the cited studies reveals that modern approaches to image restoration were increasingly focused not only on noise reduction, but also on restoring the structural integrity of the image. However, the extent to which the U-Net architecture was capable of combining noise reduction with the reconstruction of structurally significant elements in the presence of complex background noise and weakened useful signals remained insufficiently explored. The aim of the study was to evaluate the application of a modified U-Net encoder-decoder neural network architecture for image denoising and quality enhancement on synthetic and real data. To achieve this goal, the following objectives were formulated: analyse modern deep learning approaches to processing and restoring noisy images and determine the place of the U-Net architecture among reconstruction methods; perform a quantitative assessment of reconstruction quality on synthetic data; evaluate the model's performance on real data with various noise characteristics, using seismic images as an example; and perform a spectral analysis of the denoising results to determine the degree of image structure preservation.

Materials and Methods

The study was conducted from February 2025 to March 2026 as a computational experiment. The U-Net architecture proposed by O. Ronneberger *et al.* (2015) served as the methodological basis, applied as a baseline encoder-decoder model for restoring noisy images, while preserving local structures. Seismic data were considered as an applied example of noisy images to test the model's performance under conditions of complex signal structures and varying degrees of noise distortion. The methodological framework included forming a training set, constructing and training a neural network model on synthetic data, quantitatively evaluating the results, and then applying the trained network to real images with various noise characteristics. This allowed evaluation of the model both under controlled conditions and when transferred to real data. The study material consisted of synthetic and real images. The synthetic set was generated based on three-dimensional data constructed from one-dimensional reflectivity traces. These one-dimensional signals were transformed into volumetric structures by introducing geometric deformations, such as inclined areas, folded shapes, and faults. Once the clean volumes had been formed, noise of varying intensities and textures was added to the clean volumes, creating "noisy image/clean image" pairs used for network training. This approach prevented the network from adapting to a single fixed level of degradation and facilitated the creation of a generalised model stable to a wide range of noise scenarios. All images were normalised to a range from -1 to 1, ensuring comparability of amplitude characteristics and computational stability. To expand the variability of the training data, amplitude scaling coefficients were

additionally applied to change the overall intensity level of the synthetic images; the values were sampled within the range from 0 to 1. This allowed for the modelling of varying signal intensities and reduced the risk of overfitting at a fixed level of noise degradation.

The network architecture had a symmetrical "encoder-decoder" structure. The encoding section used successive convolutional blocks, including convolution, batch normalisation, and a nonlinear activation function, which ensured feature extraction with a gradual decrease in spatial resolution. In the decoding section, its sequential reconstruction was performed using transposed convolutions. The transfer of local spatial information between symmetric layers of the network was carried out via skip connections, which helped to preserve fine image details. In the output section, a residual block was additionally used, based on the concept of deep residual learning by K. He *et al.* (2016), to refine the reconstruction result. The output layer was a 1×1 convolution with linear activation to form the reconstructed image. The estimated noise was defined as the difference between the input noisy image and the reconstructed result; this was the form in which it was further used in the visual analysis of the results. The model was trained using a supervised learning scheme: a noisy image was fed as input, and its clean version was used as the target output. Parameter optimisation was performed using the Adam algorithm, which provided adaptive updating of network weights based on gradient information (Kingma & Ba, 2015). The mean absolute error (MAE) was used as the primary loss function, based on the results of H. Zhao *et al.* (2017), who demonstrated its suitability for preserving image boundaries and local features. Thus, the computational scheme was aimed not only at reducing the MAE, but also at preserving the structural integrity of the data.

To evaluate the quality of the model on synthetic data, a test set of 1,600 samples formed on the basis of 100 synthetic cubes was used. Since reference clean images were available for this data, the reconstruction efficiency was determined using the metrics of signal-to-noise ratio, peak signal-to-noise ratio, and multiscale structural similarity index. The use of the latter metric was based on the approach of Z. Wang *et al.* (2003), according to which structural similarity was an informative characteristic of image quality, when comparing reconstructed and reference data. The signal-to-noise ratio (SNR) metric was used to characterise the degree of noise suppression, the peak signal-to-noise ratio (PSNR) was used to assess the relationship between signal power and reconstruction error, and the multiscale structural similarity index measure (MS-SSIM) was used to analyse the preservation of the spatial organisation of the reconstructed image. Together, these metrics provided a quantitative assessment of noise suppression, reconstruction accuracy, and structure preservation. For comparative analysis, the proposed model was compared with two alternative approaches: the FXDECONV frequency-domain deconvolution method described in the works of L.L. Canales (1984) and N. Gülünay (1986), and the DnCNN architecture developed by K. Zhang *et al.* (2017). This comparison was consistent with further quantitative and visual analysis of the results on synthetic test images.

To test the model on real data, three seismic datasets were used: Kerry, Volve, and F3. The choice was determined by differences in noise intensity, its complexity, and readability of the useful signal, which allowed these datasets to be considered as three model testing scenarios: processing data with pronounced noise masking, reconstruction with a relatively low level of noise contamination, and processing images with a mixed nature of random and more coherent interference. When characterising the F3 dataset, the data presented by R.M. Silva *et al.* (2019) were taken into account, for Volve – the data on the deposit presented by T.J. Szydlik *et al.* (2006), and for Kerry – the results of the structural interpretation considered by W.D.M. Alotaby (2015). The choice of these sets was determined by the differences in the structural organisation of the main signal and noise characteristics, which made it possible to evaluate the stability of the model in conditions of heterogeneity of the input data. The estimated noise was calculated as the difference between the input noisy image and the reconstructed result. Since reference noise-free data were not available for real images, the quality of the reconstruction was estimated indirectly – by the nature of the suppression of the noise component, the visual evaluation of spatially significant structures and changes in the averaged Fourier amplitude spectra before and after processing. This analysis scheme corresponded to the subsequent interpretation of the results for Kerry, Volve and F3 as three modes of noise degradation of real data. The computational implementation of the model was carried out in the Python environment using the PyTorch library (PyTorch Foundation, USA). Computational experiments were conducted on an NVIDIA Quadro P5000 GPU (NVIDIA Corporation, USA). A complete training cycle took approximately 8 hours, with the average duration of one epoch being about 140 seconds. Applying the trained model to the full test dataset took approximately 1 hour. The methodology used was aimed at testing reconstruction accuracy, model robustness to varying noise intensities and structures, and the preservation of spatial and spectral image characteristics after denoising.

Results

Quantitative evaluation of synthetic image reconstruction quality

After training the model using the Adam algorithm, a quantitative evaluation of reconstruction quality was performed on blind synthetic test images (Kingma & Ba, 2015). A comparison of the proposed method with the classical FXDECONV method and the DnCNN convolutional architecture revealed differences in SNR, PSNR, and MS-SSIM metrics (Canales, 1984; Gülünay, 1986; Zhang *et al.*, 2017). These differences affected both the degree of noise suppression and the preservation of image morphology, including image sharpness, local texture modification, and the relative positions of structural elements. The corresponding average SNR, PSNR, and MS-SSIM values were presented in Table 1.

Table 1. Average SNR, PSNR, and MS-SSIM values for synthetic test images using different reconstruction methods

Metrics	Noisy	FXDECONV	DnCNN	The proposed method
SNR, dB	31.0	31.5	20.9	45.9
PSNR, dB	23.8	23.9	18.4	29.7
MS-SSIM	0.94	0.94	0.86	0.99

Source: developed by the author

As shown in Table 1, the highest average values for all three metrics were recorded for the proposed method. The most pronounced differences between the compared approaches were observed for SNR and MS-SSIM. For the proposed method, the SNR was 45.9 dB, while for the noisy input, FXDECONV, and DnCNN, it was 31.0, 31.5, and 20.9 dB, respectively. These results indicated that after reconstruction, the contribution of the useful signal to the final image increased not only compared to the original noisy data but also compared to the results of alternative methods. Comparison with the FXDECONV revealed a significant difference both in the absolute SNR value and in the nature of the reconstruction. With the proposed method, reconstruction was accompanied by a change in SNR toward better separation of noise from signal, while for FXDECONV, the final value remained close to the level of the original image. When compared with DnCNN, the SNR was lower not only compared to the result of the proposed method but also for the noisy input image. This indicated a different trade-off between noise suppression and preservation of the image's content. The same direction of differences persisted for MS-SSIM. For the proposed method, the value of this metric reached 0.99, while for the noisy input and FXDECONV it was 0.94, and for DnCNN – 0.86. This comparison showed that reconstruction in the proposed approach was accompanied by the preservation of the spatial organisation of the image at a level close to the reference one, which was consistent with the interpretation of MS-SSIM as an indicator of structural similarity at several scale levels (Wang *et al.*, 2003). In the compared methods, the structural correspondence either remained unchanged or decreased. The reduction in the residual noise level occurred simultaneously with the preservation of the relative sharpness of boundaries, local intensity transitions, and the overall morphology of the image. Thus, the differences between the methods were recorded not at a single particular level, but in two interrelated dimensions of reconstruction – the degree of noise reduction and the degree of structure preservation. Taken together, these results demonstrated that the proposed approach resulted in image reconstruction with a different level of residual noise compared to FXDECONV and DnCNN. Quantitative comparison was supplemented by a visual comparison of the reconstructed images. Figure 1 showed reference samples, noisy input data, and reconstruction results obtained using FXDECONV, DnCNN, and the proposed method. Preservation of boundaries and fine details in the reconstructed images was consistent with the use of residual refinement at the model output (He *et al.*, 2016).

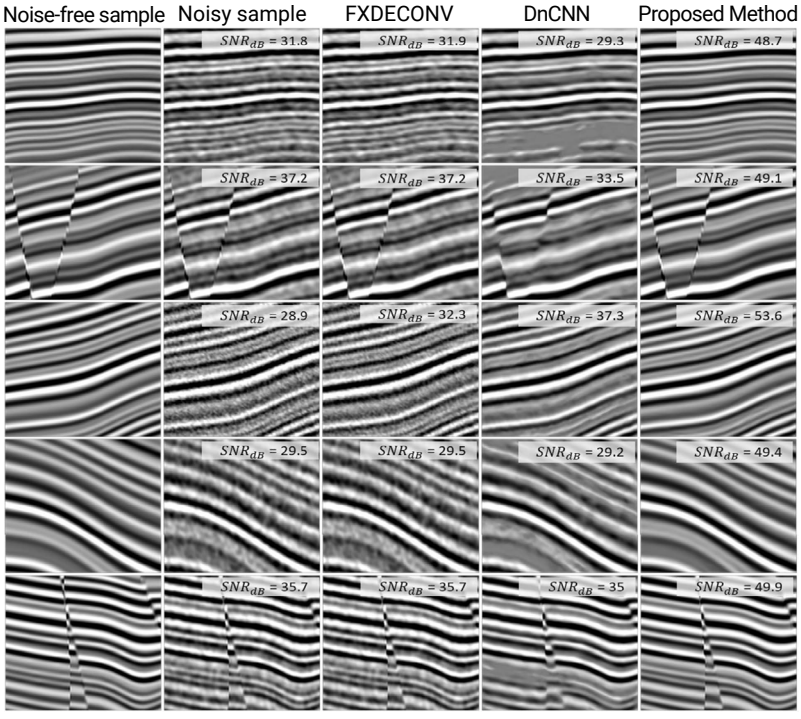

Figure 1. Visual comparison of synthetic test image reconstruction: reference image, noisy input, FXDECONV, DnCNN, and the proposed method

Source: based on L.L. Canales (1984), N. Gülünay (1986), K. Zhang *et al.* (2017)

As shown in Figure 1, the differences between the compared reconstruction options were evident not only in quantitative metrics but also in the visual reconstruction of the synthetic image structure. With the noisy input, SNR values ranged from 28.9 to 37.2 dB, while after applying the proposed method, the values increased to 48.7 to 53.6 dB. Moreover, the noise component in the original noisy images reduced the clarity of boundaries, blurred local transitions, and disrupted the continuity of reflective events. The FXDECONV method only partially reduced noise: in some samples, residual distortion, background inhomogeneity, and incomplete reconstruction of reflective boundary geometry remained, consistent with limited changes in SNR, PSNR, and MS-SSIM compared to the noisy input. The DnCNN results exhibited a different pattern of deviation, such as a reduction in the noise component with more pronounced smearing of local features, attenuation of fine details, and changes in the textural organisation of the image. Against this background, the reconstruction obtained by the proposed method was closest to the reference image, resulting in effective noise reduction with accurate preservation of edges, local contrasts, the continuity of reflective events, and the configuration of fine details without introducing excessive smoothing. Thus, the results on synthetic test images demonstrated a consistent advantage of the proposed method over FXDECONV and DnCNN. This advantage was demonstrated through both qualitative and quantitative evaluations of the structure of the reconstructed images.

Application of the modified U-Net architecture to real-world data with different noise characteristics

The results of applying the modified U-Net architecture to real-world data showed that the previously established reconstruction properties were preserved not only under controlled synthetic testing conditions but also when processing images with different noise intensity, structure, and frequency content (Ronneberger *et al.*, 2015). Analysis of three datasets revealed that the algorithm's performance was not limited to a single reconstruction type. Depending on the characteristics of the input image, differences were evident in three practical testing modes: with pronounced noise contamination, with a relatively clean signal, where reconstruction required more careful noise suppression, and with a mixed-mode distortion, combining random and locally coherent noise. In this regard, the Kerry, Volve, and F3 datasets were considered not as isolated application examples, but as complementary scenarios for testing the model on heterogeneous input data (Szydlik *et al.*, 2006; Alotaby, 2015; Silva *et al.*, 2019). The first of these scenarios was related to the Kerry dataset, for which the original images were characterised by pronounced noise contamination, which hindered the perception of the useful signal. In this case, the analysis focused on whether the reduction of the random noise component was accompanied by the restoration of the structural readability of the image under conditions of partial masking of useful elements by interference. The obtained results showed that when processing the Kerry data, reconstruction was accompanied not only by a reduction in the visually perceived noise level but also by a clearer representation of the internal organisation of the image. Consequently, individual structural elements were more clearly traced, and the spatial relationships between these elements were recorded more consistently. The result of applying the model to the Kerry data is presented in Figure 2, which compares the original image, the reconstructed result, and the estimated noise component.

As can be seen in Figure 2, the original image was characterised by a high level of noise contamination, as a result of which part of the useful signal was partially masked, and the spatial structure of the image was perceived less clearly. After applying the model, the background noise component was reduced, extended structural lines were more clearly visible, and local variations were reproduced with less interference. The reconstructed image showed no signs of excessive smoothing. Its internal heterogeneity was preserved, while becoming more consistent with the underlying signal rather than to random noise fluctuations. Analysis of the panel with estimated noise revealed that the extracted component contained predominantly chaotic high-frequency and weakly structured elements, while the main extended features of the useful image were largely absent. These results indicated a separation of signal and noise without noticeable disruption of the spatially organised components. In other words, the model reduced not the overall image variability as such, but primarily that portion of it related to noise contamination, while preserving the fundamental signal geometry in the reconstructed result. In other words, for the intense noise scenario, this was reflected in a reduction in the noise level while maintaining the structural coherence of the image. The result obtained on the Kerry dataset characterised the first mode of model validation on real data, namely processing an image with pronounced noise masking. In this case, the reconstruction effect was primarily evident in the restoration of seismic reflective events clarity and the reconstruction of spatial structure. Thus, the Kerry dataset illustrates the model's behaviour in conditions where processing involves mitigating substantial noise contamination while preserving the signal's internal geometry rather than performing minor residual corrections.

The next testing mode was performed using the Volve dataset, for which the original data had a lower level of noise contamination. In this situation, the analysis shifted from the severity of noise reduction to the degree of preservation of weak and already distinguishable structural details after reconstruction. The corresponding visual result was presented in Figure 3.

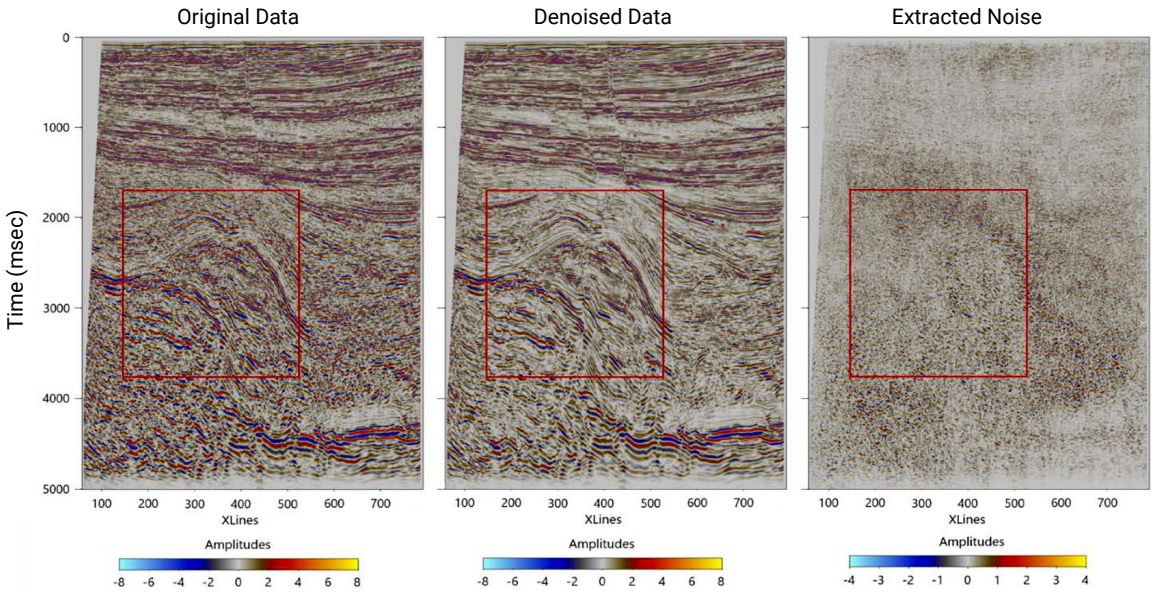

Figure 2. The result of applying the model to real Kerry data: original image, reconstructed image, and estimated noise
Note: the colour range used to estimate the noise level is compressed by half. Red rectangles highlight fragments for visual comparison of changes in the signal structure after processing. The colour scale reflects the amplitude distribution in the original image, reconstructed data, and the extracted noise component
Source: based on T.J. Szydlik *et al.* (2006), W.D.M. Alotaby (2015), R.M. Silva *et al.* (2019)

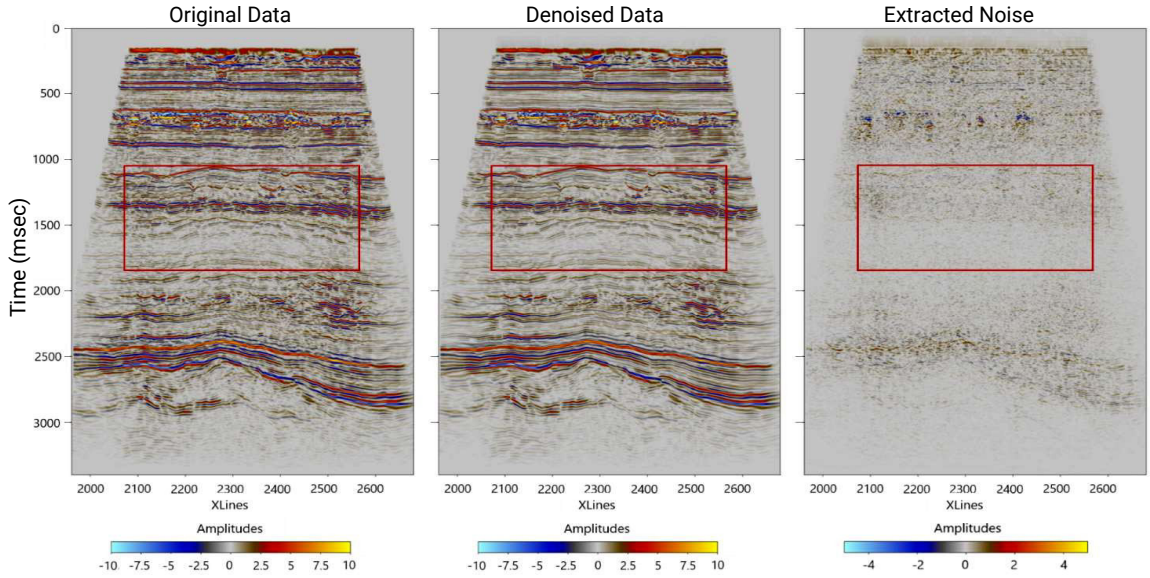

Figure 3. The result of applying the model to real Volve data: original image, reconstructed image, and estimated noise
Source: based on T.J. Szydlik *et al.* (2006), W.D.M. Alotaby (2015), R.M. Silva *et al.* (2019)

As can be seen in Figure 3, in the case of the Volve dataset, the original image was characterised by a lesser degree of noise masking of reflective events compared to the intense noise contamination scenario. In this setting, the reconstruction result was assessed not so much by the degree of noise suppression as by the preservation of already distinguishable structural details. After applying the model, the main seismic events retained the continuity and geometry, while the small-scale chaotic component within the selected area was reduced. At the same time, local transitions, low-contrast elements, and the configuration of seismic reflection events remained distinguishable. The estimated noise component contained predominantly scattered, weakly structured fluctuations and did not reproduce the extended elements of the primary signal. This indicated that the processing in this case was focused on reducing residual noise without disrupting the internal structure of the image. Overall, the results on the Volve dataset characterised a different operating mode of the model compared to the Kerry case: whereas

in the former case, the main effect was associated with reducing pronounced noise masking, here, reconstruction manifested itself in local correction of the residual noise component while preserving the already discernible signal structure without introducing artefacts. The third testing mode was represented by the F3 dataset, which was characterised by a non-uniform complex spatial organisation of noise distortions. In this case, reconstruction included both reducing the random noise component and separating the useful true signal from more structured distortions while preserving the geometry of the reflective events. The corresponding result was shown in Figure 4.

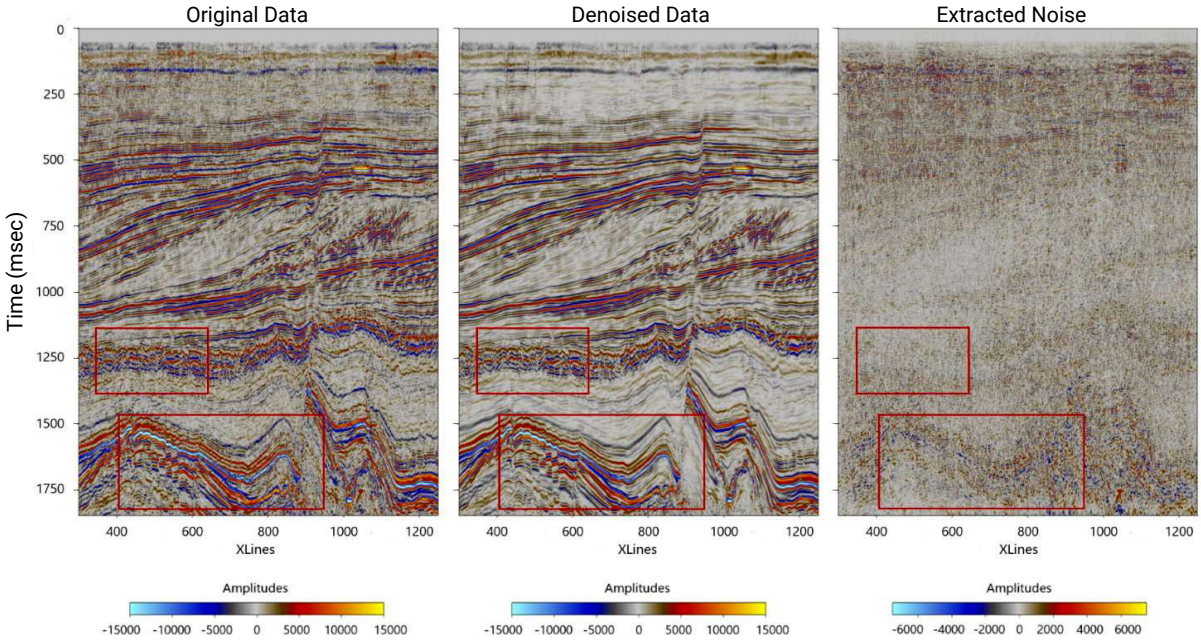

Figure 4. The result of applying the model to real F3 data: original image, reconstructed image and estimated noise
Source: based on T.J. Szydlik *et al.* (2006), W.D.M. Alotaby (2015), R.M. Silva *et al.* (2019)

As can be seen in Figure 4, the F3 dataset was characterised by a non-uniform noise distortion configuration compared to previous cases, as it combined random and more organised noise components. In this formulation, the reconstruction task was associated not only with reducing the overall level of noise variability, but also with separating structurally non-uniform distortions from the main seismic signal. After applying the model, the overall level of the noise component decreased, while the main reflection elements retained visual continuity and spatially consistent. This indicated that the reconstruction addressed not only the chaotic background, but also the more organised noise components without noticeably disrupting the internal structure of the signal. Analysis of the panel with estimated noise showed that both random and locally coherent interfering elements were converted into the noise component, while spatially significant contours, boundaries, and local texture variations were preserved in the reconstructed image. This indicated the model's performance under mixed-distortion conditions, where the boundary between the useful and unwanted components is less obvious. Thus, the result on F3 complemented the two previous testing scenarios: while the main effect on Kerry was restoring readability under significant noise masking, and on Volve, reducing residual noise while preserving the already discernible structure, on F3, reconstruction was associated with separating the useful signal from inhomogeneous noise components under conditions of the mixed spatial organisation. Comparison of the results for Kerry, Volve, and F3 showed that the reconstruction pattern was preserved under three different noise degradation regimes of real data. In the first case, reconstruction involved reducing intense noise masking, in the second, correction with a relatively clean signal, and in the third, processing mixed random and coherent distortions. Thus, the results demonstrated that the reconstruction features were reproduced with varying degrees and spatial organisation of noise.

Spectral analysis of image denoising results

Spectral analysis results showed that after applying the modified U-Net architecture to all three datasets, the amplitude of noise-related components was reduced while maintaining the general shape of the spectral distribution in the informative frequency range. This indicated a reduction in noise energy without significantly altering the spectral components associated with the useful signal. In this case, spectral characteristics were used as an additional tool to verify whether denoising preserved signal structure not only at the visual but also at the frequency level. The averaged amplitude spectra of the original and reconstructed data for the Kerry, Volve, and F3 datasets were presented in Figure 5.

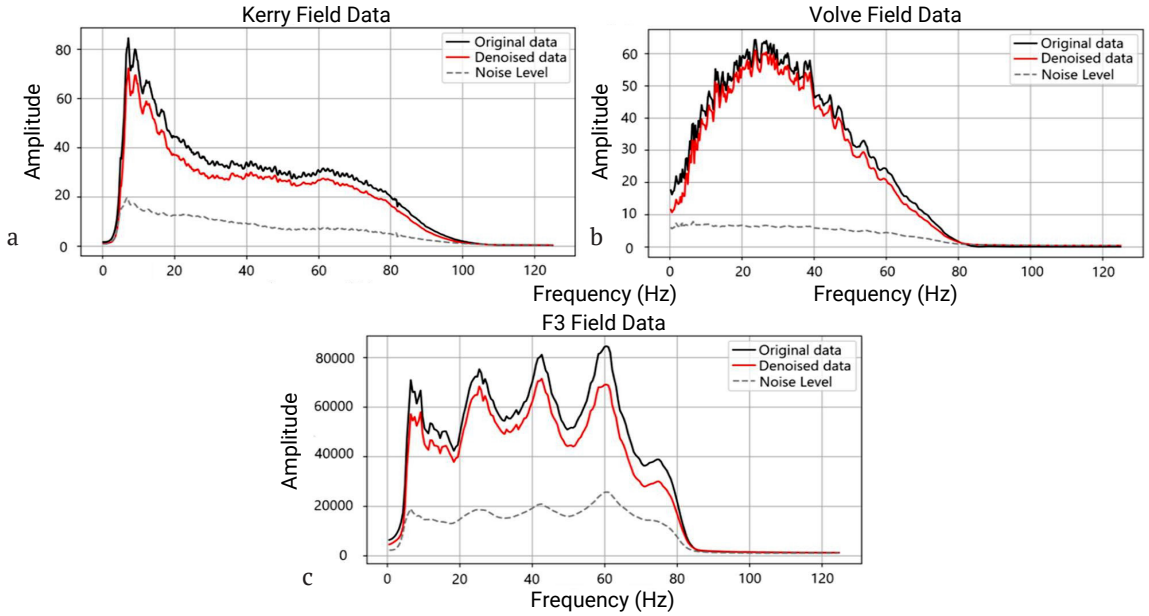

Figure 5. Averaged Fourier amplitude spectra over the entire time interval for the Kerry (a), Volve (b) and F3 (c) sets: original data, noise reduction result and estimated noise component
Source: based on T.J. Szydlik *et al.* (2006), W.D.M. Alotaby (2015), R.M. Silva *et al.* (2019)

As shown in Figure 5, for all three datasets, a reduction in spectral amplitude in the region associated with the noise component was observed after noise reduction, while the spectral configuration in the informative frequency range was preserved. A graphical comparison of the original data, the reconstructed result, and the estimated noise component revealed that the main differences between the curves were concentrated in the part of the spectrum where the noise contribution remained most pronounced, while these differences diminished as the curves approached the upper limit of the working range. For Kerry, discrepancies between the original and reconstructed data were observed up to approximately 100 Hz, where a decrease in amplitude was observed after processing. However, the basic shape of the spectrum in the working range was preserved, and the position of characteristic sections of the curve did not change significantly. For Volve, the changes were less pronounced and the discrepancy between the curves was noted primarily up to 80 Hz, after which the curves converged. These results indicated a reduction in the residual noise component without a noticeable change in the informative portion of the spectrum, and corresponded to a data processing mode with less pronounced noise contamination. For F3, a similar pattern persisted under conditions of non-uniform interference organisation, where a reduction in amplitude after processing was observed across almost the entire operating range up to approximately 85 Hz. Moreover, the dotted curve representing the estimated noise component exhibits a lower amplitude compared to the spectra of the original and reconstructed data, further reflecting the redistribution of energy after processing. This indicated that the reconstruction process attenuates both random and partially structured noise components without changing the overall shape of the spectrum in the region associated with the useful signal. Thus, the spectral data for Kerry, Volve, and F3 showed that reconstruction effectively reduced the noise component while preserving the main characteristics of the useful signal. Taken together, the quantitative, qualitative, and spectral results demonstrated that the modified U-Net architecture and the computational framework implemented on its basis ensured the reconstruction of noisy images across a range of noise intensities and distortion types. On synthetic data, this was reflected in improved SNR, PSNR, and MS-SSIM values relative to FXDECONV and DnCNN, while on real data it was evidenced by the consistent reproduction of reconstruction patterns under varying noise degradation conditions. The obtained results indicated that the proposed approach under consideration was aimed not only at reducing the noise component but also at preserving the spatial organisation of the image, local boundaries, transitions, and spectrally significant signal characteristics. Seismic data were used as a representative application example to evaluate the method on images with complex internal structures and heterogeneous noise patterns. Overall, this formulation allowed considering the proposed computational framework as a generalisable approach, applicable beyond seismic data to a broader range of problems that required both reduction of the noise component and preservation of the structural integrity in reconstructed images.

Discussion

The obtained results showed that the performance of the modified U-Net architecture was characterised not only by effective suppression of the noise component, but also by preservation of spatially organised image structures. These findings were consistent with the results of T. Zhong *et al.* (2022), where a deep residual U-Net architecture was

also accompanied by random noise attenuation while preserving reflective events and geometric consistency. This comparison indicated a relationship between the use of the U-Net architecture with residual connections and the simultaneous attenuation of noise energy and the preservation of the internal structure of the signal. The combination of multi-level feature extraction with skip and residual connections facilitated the separation of the noise component from the useful content without pronounced smoothing. Quantitative evaluation on synthetic data showed higher values of the SNR, PSNR, and MS-SSIM compared to the baseline methods. These observations aligned with the more general results of Z. Wu *et al.* (2025), where improvements in the quality of reconstruction were also considered as a consequence of the simultaneous correction of noise distortions and the preservation of the structural characteristics of the image in the frequency domain. In this study, these differences were particularly evident for data, in which the spatial organisation of the signal was important for interpretation. Unlike general image restoration tasks, where emphasis was often placed on overall reconstruction fidelity, structurally complex images require the preservation of low-contrast yet spatially consistent features. Accordingly, the advantages of the proposed method were reflected not only in quantitative metrics, but also in visual reconstruction quality, where noise reduction does not compromise local transitions, contours, or fine details. Analysis of the data presented by Y.-T. Wu & R.R. Stewart (2023) showed similar conclusions, who associated the use of U-Net with attenuation of coherent noise, while preserving the structural integrity of seismic data. In both cases, neural network reconstruction was accompanied by a reduction in noise while preserving the informative signal. This suggested that the U-Net architecture can be considered as a tool for separating spatially organised useful and parasitic components. Analysis of real data further demonstrated that the nature of the reconstruction was preserved under several noise degradation conditions, such as pronounced noise contamination, low noise presence, and with a mixed nature of distortions. This robustness was consistent with the findings of F.K. Anjom *et al.* (2024), according to which modern machine learning methods in seismic exploration were characterised by stability with respect to variability in geological and noise conditions. Overall, the results suggest that the proposed neural network framework provides not only high accuracy under controlled conditions but also reliable and reproducible performance on heterogeneous real-world data. Controlled variability of training pairs and noise contamination ranges ensured the formation of a more generalised scheme for separating signal and interference.

The obtained results showed that, when processing the Kerry dataset, the main reconstruction effect was manifested in the restoration of the readability of reflective events under conditions of intense noise masking, whereas for Volve, a more local correction of residual noise predominated, and for F3, a stable separation of random and more organised distortions. These findings partially correlated with the results of H. Tang *et al.* (2023), where the RRU-net (residual recurrent U-Net) architecture was used for simultaneous reconstruction and noise reduction of complex data obtained by the distributed acoustic sensing method in vertical seismic profiling. In both cases, the neural network approach demonstrated effectiveness not only within a single distortion regime, but under conditions of a more complex and structurally complex input data. The key difference was determined by the specificity of the source material: in the case of distributed acoustic sensing data, the spatial organisation of the signal and noise had distinct characteristics, whereas this study, the stability of the model was assessed on several types of real seismic images with varying levels presence of random and locally-coherent noise. The obtained results showed the effective reduction of the noise component was accompanied by the preservation of the structural integrity of the image and was not reduced to a simple high-frequency smoothing. These findings were consistent with the results of L. Yang *et al.* (2021), where an improved residual convolutional neural network provided random noise suppression, while preserving main events. The agreement across studies suggested that architectures with residual mechanisms provided a balance between noise suppression and preservation of signal. This can be explained by the role of residual connections, which increased the stability of feature extraction and reduced the risk of losing weak, but meaningfully significant image elements. Comparison with the results of H. Xi *et al.* (2026) further confirmed that the effectiveness of U-Net-based architectures in seismic denoising was maintained in more recent modifications aimed for more accurate signal protection in complex noise environments. This research results demonstrated a similar trend, as reconstruction was accompanied by a reduction in the noise level without significant degradation of the primary signal content. However, the emphasis here extended beyond noise suppression alone, and also on the stability and generalisation of the effect under various degradation conditions in real data. This distinction reflected a broader analytical framework, integrating synthetic, visual, and spectral evaluations rather than focusing solely on the performance of a specific environment.

The results showed that the reconstruction process was learned under supervised training conditions using "noisy image/clean image" pairs, in which the model was trained to establish a direct mapping between degraded and original signals. This formulation was consistent with the adopted methodological framework and supported the interpretation of improvements in SNR, PSNR, and MS-SSIM as outcomes of supervised learning. This approach differed from that of J. Lehtinen *et al.* (2018), where Noise2Noise was an unsupervised approach to restore images without clean reference data. The distinction lain in the nature of the training information: in supervised settings, reconstruction relied on explicit correspondence between noisy and clean signals, whereas Noise2Noise exploited the

statistical consistency across multiple noisy realisations of the same image. A similar contrast can be drawn with the work of J. Batson & L. Royer (2019), where self-supervised denoising strategies were employed without an explicit ground-truth reference. In this study, the use of supervised training pairs was accompanied by a change in quantitative indicators on synthetic data and preservation of the nature of reconstruction when applied to real images. For structurally complex data with a heterogeneous noise background, the presence of a reference target provided a straightforward scheme for separating useful and parasitic components while preserving spatially significant elements.

Differences among noise reduction approaches were evident not only in comparison with classical methods but also relative to early convolutional models. In the work of X. Si *et al.* (2019), convolutional neural networks were applied to attenuate random noise in seismic data, reflecting a similar objective of separating useful signal from noise. However, in this study, reconstruction was addressed more broadly, such as not merely as denoising, but as the preservation of both spatial and spectral organisation. This perspective was consistent with the model design, where skip and residual connections support the retention of internal image structure during reconstruction. An alternative formulation was presented by A. Krull *et al*. (2019), where noise reduction was achieved without clean reference data through self-supervised learning on single noisy images. In contrast, the approach considered here relied on supervised training with explicit correspondence between degraded and reference data, enabling more direct separation of useful and spurious elements. A related emphasis appeared in the work of T. Garber & T. Tirer (2024), where image restoration was evaluated not only by error reduction but also by preservation of meaningful signal content. In this study, the distinction lain primarily in the computational implementation, as the modified U-Net with skip and residual connections provided a specific mechanism for achieving this balance. Similar principles extended to more recent architectures. In the work of S.W. Zamir *et al.* (2022), the Restormer model was designed to restore high-resolution images by capturing long-range dependencies. As in the present case, reconstruction goes beyond local smoothing across multiple representation levels. However, while transformer-based models emphasised global context modelling, the approach here relied on multi-level convolutional feature extraction combined with skip connections and residual refinement. Another direction was illustrated by V. Potlapalli *et al.* (2023) with PromptIR, which targeted multiple degradation types within a single framework. A partial similarity can be observed in the stability of reconstruction across different noise conditions in this study, from strong contamination to mixed distortions. Nevertheless, PromptIR was inherently designed as a general-purpose restoration model that operated without predefined degradation types and incorporated a prompting mechanism. In contrast, the present framework addressed a more specific denoising task, with emphasis on preserving signal structure under defined noise conditions. Overall, the key distinction lain in the scope of applicability: generalised model aimed to handle diverse degradation types within a unified architecture, whereas the proposed approach focused on accurate reconstruction within a specified noise regime while maintaining structural fidelity.

Conclusions

The study presented a method for image quality enhancement though noise removal using the modified U-Net architecture and synthetic images for training purpose. On synthetic blind test images, the proposed methodology achieved a higher quantitative performance compared to FXDECONV and DnCNN. On real data, it consistently maintained the reconstruction quality across a range of noise degradation conditions. This was reflected in the accurate restoration of intensity characteristics, as well as the preservation of edges, variations in local features, morphology, and the relative spatial arrangement of structural elements, without introducing smoothing effect. A quantitative comparison on synthetic test images revealed clear differences between the methods across the main reconstruction metrics. The proposed approach achieved average values of 45.9 dB for SNR, 29.7 dB for the PSNR, and 0.99 for the MS-SSIM. In comparison, FXDECONV yielded 31.5 dB, 23.9 dB, and 0.94, while DnCNN produced 20.9 dB, 18.4 dB, and 0.86, respectively. These differences indicated improvements not only in noise suppression, but also in intensity fidelity and preservation of spatial structure of the image. When applied to real data, the method demonstrated consistent reconstruction behaviour across multiple noise degradation scenarios. For the Kerry dataset, this was reflected in reduced noise masking and improved visibility of structural features. For Volve, it was expressed as a decrease in residual noise while preserving the already distinguishable signal structure. In the case of F3, stable performance was maintained despite more complex spatial interference patterns. Spectral analysis further confirmed that reconstruction, across all three datasets was associated with attenuation of the noise content without altering the overall spectral structure within the main frequency bandwidth. Taken together, obtained results demonstrated that the modified U-Net architecture was applicable for denoising and reconstructing structurally complex images. The combination of encoding and decoding components, skip, and residual corrections enabled noise reduction without significantly disrupting the image. These findings supported the applicability of the proposed approach beyond seismic data to other types of noisy images, for which simultaneous noise suppression and preservation of fine scale detail was crucial. The study was limited by the finite number of datasets analysed, and the model was tested primarily under noise-based distortion conditions, without addressing a wider range of combined image degradations. Further research may include expanding the dataset diversity, evaluating robustness under more complex distortion scenarios, and comparing

performance with other modern reconstruction architectures across varied image degradation conditions.

Acknowledgements
None.

Funding
None.

Conflict of Interest
None.

References

[1] Alotaby, W.D.M. (2015). *Fault interpretation and reservoir characterization of the Farewell formation within Kerry Field, Taranaki Basin, New Zealand*. (Master's thesis, Missouri University of Science and Technology, Rolla, USA).
[2] Anjom, F.K., Vaccarino, F., & Socco, L.V. (2024). Machine learning for seismic exploration: Where are we and how far are we from the holy grail? *Geophysics*, 89(1), 157-178. doi: 10.1190/geo2023-0129.1.
[3] Azizova, S., Mammadova, S., Hasanov, J., & Aliyeva, S. (2026). Qualitative analysis of medical image colorization with the realistic color palette adjustment. *Problems of Information Technology*, 17(1), 23-31. doi: 10.25045/jpit.v17.i1.03.
[4] Batson, J., & Royer, L. (2019). Noise2Self: Blind denoising by self-supervision. In *Proceedings of the 36th international conference on machine learning*. California: PMLR 97.
[5] Canales, L.L. (1984). Random noise reduction. In *Proceedings of the SEG technical program expanded abstracts 1984* (pp. 525-527). New York: American Society of Mechanical Engineers. doi: 10.1190/1.1894168.
[6] Cui, Y., Ren, W., Cao, X., & Knoll, A. (2024). Revitalizing convolutional network for image restoration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(12), 9423-9438. doi: 10.1109/TPAMI.2024.3419007.
[7] Ding, M., Zhou, Y., & Chi, Y. (2024). Seismic signal denoising using Swin-Conv-UNet. *Journal of Applied Geophysics*, 223, article number 105355. doi: 10.1016/j.jappgeo.2024.105355.
[8] Garber, T., & Tirer, T. (2024). Image restoration by denoising diffusion models with iteratively preconditioned guidance. In *IEEE/CVF conference on computer vision and pattern recognition* (pp. 25245-25254). Seattle: IEEE Computer Society. doi: 10.1109/CVPR52733.2024.02385.
[9] Goyal, B., Dogra, A., Agrawal, S., Sohi, B.S., & Sharma, A. (2020). Image denoising review: From classical to state-of-the-art approaches. *Information Fusion*, 55, 220-244. doi: 10.1016/j.inffus.2019.09.003.
[10] Gülünay, N. (1986). FXDECON and complex wiener prediction filter. In *Proceedings of the SEG technical program expanded abstracts 1986* (pp. 279-281). New York: American Society of Mechanical Engineers. doi: 10.1190/1.1893128.
[11] He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *IEEE conference on computer vision and pattern recognition* (pp. 770-778). Las Vegas: IEEE. doi: 10.1109/CVPR.2016.90.
[12] Imamverdiyev, Y.N., & Musayeva, F.I. (2022). Analysis of generative adversarial networks. *Problems of Information Technology*, 13(1), 20-27. doi: 10.25045/jpit.v13.i1.03.
[13] Kingma, D.P., & Ba, J.L. (2015). Adam: A method for stochastic optimization. In *3rd international conference on learning representations*. San Diego, USA. doi: 10.48550/arXiv.1412.6980.
[14] Krull, A., Buchholz, T.-O., & Jug, F. (2019). Noise2Void – learning denoising from single noisy images. In *IEEE/CVF conference on computer vision and pattern recognition* (pp. 2129-2137). Piscataway: IEEE.
[15] Lehtinen, J., Munkberg, J., Hasselgren, J., Laine, S., Karras, T., Aittala, M., & Aila, T. (2018). Noise2Noise: Learning image restoration without clean data. In *Proceedings of the 35th international conference on machine learning*. Stockholm: PMLR.
[16] Luo, Z., Gustafsson, F.K., Zhao, Z., Sjölund, J., & Schön, T.B. (2025). Taming diffusion models for image restoration: A review. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 383(2299), article number 20240358. doi: 10.1098/rsta.2024.0358.
[17] Mao, J., Sun, L., Chen, J., & Yu, S. (2025). Overview of research on digital image denoising methods. *Sensors*, 25(8), article number 2615. doi: 10.3390/s25082615.
[18] Nazir, N., Sarwar, A., & Saini, B.S. (2024). Recent developments in denoising medical images using deep learning: An overview of models, techniques, and challenges. *Micron*, 180, article number 103615. doi: 10.1016/j.micron.2024.103615.
[19] Potlapalli, V., Zamir, S.W., Khan, S., & Khan, F.S. (2023). PromptIR: Prompting for all-in-one blind image restoration. In *Proceedings of the 37th international conference on neural information processing systems* (pp. 71275-71293). Red Hook: Curran Associates, Inc. doi: 10.5555/3666122.3669243.
[20] Ronneberger, O., Fischer, P., & Brox, T. (2015). U-Net: Convolutional networks for biomedical image segmentation. In N. Navab, J. Hornegger, W. Wells & A. Frangi (Eds.), *Medical image computing and computer-assisted intervention – MICCAI 2015* (pp. 234-241). Cham: Springer. doi: 10.1007/978-3-319-24574-4_28.
[21] Si, X., Yuan, Y., Si, T., & Gao, S. (2019). Attenuation of random noise using denoising convolutional neural networks. *Interpretation*, 7(3), 269-280. doi: 10.1190/INT-2018-0220.1.
[22] Silva, R.M., Baroni, L., Ferreira, R.S., Civitarese, D., Szwarcman, D., & Brazil, E.V. (2019). Netherlands dataset: A new public dataset for machine learning in seismic interpretation. *arXiv*. doi: 10.48550/arXiv.1904.00770.

[23] Szydlik, T.J., Way, S., Smith, P., Aamodt, L., & Friedrich, C. (2006). 3D PP/PS prestack depth migration on the Volve field. In *68th EAGE conference and exhibition incorporating SPE EUROPEC 2006* (cp-2-00185). Houten: European Association of Geoscientists & Engineers. doi: 10.3997/2214-4609.201402177.

[24] Tang, H., Cheng, S., Li, W., & Mao, W. (2023). Simultaneous reconstruction and denoising for DAS-VSP seismic data by RRU-net. *Frontiers in Earth Science*, 10, article number 993465. doi: 10.3389/feart.2022.993465.

[25] Tian, C., Fei, L., Zheng, W., Xu, Y., Zuo, W., & Lin, C.-W. (2020). Deep learning on image denoising: An overview. *Neural Networks*, 131, 251-275. doi: 10.1016/j.neunet.2020.07.025.

[26] Wang, Z., Simoncelli, E.P., & Bovik, A.C. (2003). Multiscale structural similarity for image quality assessment. In *The thrity-seventh asilomar conference on signals, systems & computers* (pp. 1398-1402). Pacific Grove: IEEE. doi: 10.1109/ACSSC.2003.1292216.

[27] Wu, Y.-T., & Stewart, R.R. (2023). Attenuating coherent environmental noise in seismic data via the U-net method. *Frontiers in Earth Science*, 11, article number 1082435. doi: 10.3389/feart.2023.1082435.

[28] Wu, Z., Liu, W., Wang, J., Li, J., & Huang, D. (2025). FrePrompter: Frequency self-prompt for all-in-one image restoration. *Pattern Recognition*, 161, article number 111223. doi: 10.1016/j.patcog.2024.111223.

[29] Xi, H., Luo, J., Liu, J., Shi, W., Chen, G., Wang, N., & Huang, X. (2026). MSBE-UNet: A deep learning denoising method for effective seismic noise suppression. *Acta Geophysica*, 74, article number 65. doi: 10.1007/s11600-026-01799-3.

[30] Xia, B., Zhang, Y., Wang, S., Wang, Y., Wu, X., Tian, Y., Yang, W., & Van Gool, L. (2023). DiffIR: Efficient diffusion model for image restoration. In *IEEE/CVF international conference on computer vision* (pp. 13049-13059). Paris: IEEE. doi: 10.1109/ICCV51070.2023.01204.

[31] Yang, L., Chen, W., Wang, H., & Chen, Y. (2021). Deep learning seismic random noise attenuation via improved residual convolutional neural network. *IEEE Transactions on Geoscience and Remote Sensing*, 59(9), 7968-7981. doi: 10.1109/TGRS.2021.3053399.

[32] Zamir, S.W., Arora, A., Khan, S., Hayat, M., Khan, F.S., & Yang, M.-H. (2022). Restormer: Efficient transformer for high-resolution image restoration. In *IEEE/CVF conference on computer vision and pattern recognition* (pp. 5728-5739). New Orleans: IEEE. doi: 10.1109/CVPR52688.2022.00564.

[33] Zhang, K., Zuo, W., Chen, Y., Meng, D., & Zhang, L. (2017). Beyond a gaussian denoiser: Residual learning of deep CNN for image denoising. *IEEE Transactions on Image Processing*, 26(7), 3142-3155. doi: 10.1109/TIP.2017.2662206.

[34] Zhao, H., Gallo, O., Frosio, I., & Kautz, J. (2017). Loss functions for image restoration with neural networks. *IEEE Transactions on Computational Imaging*, 3(1), 47-57. doi: 10.1109/TCI.2016.2644865.

[35] Zhong, T., Cheng, M., Dong, X., Li, Y., & Wu, N. (2022). Seismic random noise suppression by using deep residual U-Net. *Journal of Petroleum Science and Engineering*, 209, article number 109901. doi: 10.1016/j.petrol.2021.109901.

Застосування методів глибокого навчання для обробки та покращення зображень: тематичне дослідження сейсмічних даних

Руслан Маліков

Аспірант
Інститут геології та геофізики Національної академії наук Азербайджану
AZ1143, просп. Х. Джавіда, 119, м. Баку, Азербайджан
https://orcid.org/0009-0005-2126-1642

Анотація. Метою цього дослідження була оцінка ефективності модифікованої архітектури нейронної мережі кодер-декодер для шумозаглушення та покращення зображень з використанням синтетичних та реальних даних. Методологія дослідження базувалася на обчислювальному експерименті та включала навчання моделі на синтетичних зображеннях, кількісне порівняння отриманих результатів за допомогою методу f-x деконволюції та альтернативної моделі згорткового шумозаглушення, а також перевірку стійкості на реальних даних з наявністю різних шумових характеристик. Було виявлено, що застосований метод шумозаглушення характеризувався не тільки зменшенням шумової складової, але й збереженням просторово значущих характеристик зображення, включаючи різкість країв, локальні переходи, морфологію та відносне положення структурних елементів без ознак надмірного розмиття. Остаточне порівняння методів на синтетичних тестових зображеннях показало, що середнє співвідношення сигнал/шум, пікове співвідношення сигнал/шум та багатомасштабний індекс структурної подібності для запропонованого підходу становили 45,9 дБ, 29,7 дБ та 0,99 відповідно. Для методу f-x деконволюції відповідні значення становили 31,5 дБ, 23,9 дБ та 0,94, тоді як для альтернативної моделі згорткового шумозаглушення значення становили 20,9 дБ, 18,4 дБ та 0,86. При застосуванні до реальних даних зберігалася та сама поведінка покращення, включаючи видалення вираженого шумового забруднення та отримання відносно чистого сигналу без спотворень. Залежно від вхідних характеристик, метод супроводжувався зменшенням інтенсивного маскування шуму, зменшенням залишкового шуму зі збереженням чіткої структури сигналу та реконструкцією в умовах більш складної просторової організації перешкод. Спектральний аналіз виявив зменшення енергії шуму без порушення спектральної конфігурації в інформативному діапазоні частот. Практичне значення полягає в потенційному застосуванні запропонованого підходу як обчислювального методу обробки шумних зображень у системах, призначених для шумозаглушення та відновлення різних структур даних

Ключові слова: відновлення зображень; шумозаглушення зображень; структурне збереження; реконструкція сейсмічних зображень; навчання з учителем

ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ ТА КОМП'ЮТЕРНА ІНЖЕНЕРІЯ

Науково-технічний журнал

Том 23, № 1, 2026

Заснований у 2004 р. Виходить 3 рази на рік

Оригінал-макет видання виготовлено
у редакційно-видавничому відділі Вінницького національного технічного університету.

Відповідальний редактор:
В. Белзецька

Підписано до друку 26.03.2026 р. Формат 60*84/8
Умовн. друк. арк. 21,4
Наклад 100 примірників

Адреса видавництва:
Вінницький національний технічний університет
21021, вул. Хмельницьке шосе, 95, м. Вінниця, Україна
Тел: +38 (0432) 560848
Факс: +38 (0432) 465772
E-mail: info@itce.vn.ua
https://itce.vn.ua/uk

INFORMATION TECHNOLOGIES AND COMPUTER ENGINEERING

Scientific and Technical Journal

Vol. 23, No. 1, 2026

Founded in 2004. Published three times per year

The original layout of the publication is made
in the publishing department of Vinnytsia National Technical University

Managing editor:
V. Belzetska

Signed for print 26.03.2026. Format 60*84/8
Conventional printed pages 21.4
Circulation 100 copies

Publishing address:
Vinnytsia National Technical University
21021, 95 Khmelnytske Shose Str., Vinnytsia, Ukraine
Telephone: +38 (0432) 560848
Fax: +38 (0432) 465772
E-mail: info@itce.vn.ua
https://itce.vn.ua/en