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

Information Technologies and Computer Engineering **Vol. 21, No. 3. 2024**

Article's History: Received: 02.09.2024 Revised: 22.11.2024 Accepted: 26.12.2024

UDC 004.896 10.63341/vitce/3.2024.20

Method of multi-purpose term search in the terminology database

Andrii Yarovyi

Doctor of Technical Science, Professor
Vinnytsia National Technical University
21021, 95 Khmelnytske Shose Str., Vinnytsia, Ukraine
https://orcid.org/0000-0002-6668-2425

Dmytro Kudriavtsev*

Postgraduate Student
Vinnytsia National Technical University
21021, 95 Khmelnytske Shose Str., Vinnytsia, Ukraine
https://orcid.org/0000-0001-7116-7869

Abstract. This study investigated the method of multi-purpose term search in a terminological knowledge base, which is based on semantic analysis and the use of modern natural language processing methods. The study considered the key factors affecting the search efficiency, including the structure of data organisation, data format and parameters, and sample size. Particular focus was placed on the semantic similarity between terms, which allows increasing the search accuracy by using vector representations and the Louvain algorithm. The study also described the use of cosine similarity to quantify the similarity between terms. Furthermore, the search process was optimised by filtering relevant databases and dynamically identifying relevant terms using the modularity metric. A comparative analysis of existing methods for searching for terms by the identified factors was conducted. The study noted the advantages and disadvantages of using the Louvain algorithm in comparison with the search algorithms in graph data structures. A series of experiments were conducted on data samples, including dictionary, graph, and network data structures. The study analysed the use of logistic constraints for searching in network data structures and noted the possibility of optimisation due to uniform and dynamic data distribution. Experimental results showed the effectiveness of using a combination of the Louvain algorithm and network data structures in terminological knowledge bases. Examples of the scope of application of this method in information technologies for searching and processing text data were given. A software architecture scheme with the use of a software interface and the possibility of integration for web applications in the form of a package or library was developed. The proposed approach demonstrates effectiveness in the context of intelligent decision support systems and automated chatbots, which makes it particularly useful for industries where access to accurate professional terms is critical. A basic version of the software interface for using this method in information technologies for searching and analysing data for use in search engines was developed

Keywords: terminological knowledge base; semantic similarity; Louvain algorithm; vector representations; natural language processing

Introduction

In the modern world, the amount of information is constantly growing and the need for efficient data search and analysis is becoming increasingly more relevant. This is particularly true for terminology knowledge bases (TKBs), which are key tools for storing and processing specialised terms in various subject areas. Considering this, finding the right terms to process user queries, especially when working with large amounts of data, requires the introduction of new, more efficient methods. One of these approaches is the method of multi-purpose term search, which allows searching by several criteria simultaneously, considering the complex structure of the TKBs and semantic relationships between terms. The principal task of this method is to identify the terms that best match the user's

Suggested Citation:
Yarovyi, A., & Kudriavtsev, D. (2024). Method of multi-purpose term search in the terminology database. *Information Technologies and Computer Engineering*, 21(3), 20-28. doi: 10.63341/vitce/3.2024.20

*Corresponding author

query. For this, several key factors must be considered: the structure of the data in the TKB, the format and parameters of this data, its volume, as well as the number and size of the sample. Moreover, the semantic similarity between terms plays a significant role, as it is based on the analysis of the relationships between terms and their degree of relevance to the subject area. The relevance of the subject under study is conditioned by the rapid development of data processing technologies and the growing popularity of artificial intelligence (AI) in various fields of activity. The format of textual data stays unchanged, but its variability and the number of subject areas are constantly growing exponentially, which creates a demand for efficient data storage, processing, and use. In this context, terminological knowledge bases play a significant role, as they allow structuring and systematising specialised terms in various subject areas.

In scientific sources, term knowledge bases are considered the basis for storing and managing specialised terms in various fields of knowledge. TKBs perform an essential function in information systems, especially in the context of automated processing of user queries. Existing methods of searching in terminological knowledge bases include a series of approaches, such as keyword search, semantic classification, and machine learning approaches. For instance, keyword-based methods often face relevance issues because they ignore semantic relationships between terms (Abdykerimova *et al.,* 2024). In contrast, semantic classification allows considering the meaning of terms, which reduces the risk of misinterpreting queries.

Current research in this area focused on the development of methods that account for semantic similarity, data structure, and a multi-criteria approach to improve the accuracy and relevance of results (Bourgaux *et al.,* 2024). The early development of search methods in TKBs was based on keyword search and keyword-based approaches used in many classical systems. According to D. Simian & M.-E. Șerban (2024), this approach workes well for databases with clear categories and meanings, but often does not consider complex relationships between terms, which limits its use in large and multi-component TKBs. Some researchers point out the problems of keyword searching due to the need to account for ambiguities in the meaning of words and polysemy (Wu *et al.,* 2023). To overcome these limitations, it became necessary to develop methods based on semantic analysis.

Semantic data processing methods have been considerably developed through vector representations of terms, which have become possible with the development of machine learning. S. Rathje *et al*. (2024) noted that modern methods based on semantic similarity enable a more accurate assessment of the relationships between terms using metrics such as cosine similarity, which is often used to compare values in multidimensional spaces. This approach considers not only the surface meaning of terms, but also their location in the semantic space, which allows automating the search process and increasing its accuracy.

However, according to S. Roy *et al.* (2024) and M. Bienvenu *et al*. (2024), most conventional methods cannot simultaneously process a considerable number of links in the TKBs, which limits their effectiveness. C. Kaya *et al.* (2024) and E. Mohabir & Y. Yoshi (2024) noted that the increasing complexity and volume of knowledge bases requires innovative approaches to their processing, with the researchers focusing on the search for a conceptually new method of multi-objective search.

The purpose of the present study was to develop a multi-purpose search method in terminological knowledge bases, considering the dynamic data structure, data format, and sample size, for further use in information technologies for text data processing. The key features of this study included the use of algorithmisation of intermediate search results processing and the use of machine learning algorithms to identify semantic chains. The key objectives of this study were to identify search characteristics, describe them, and determine the functional features of search using artificial intelligence and iterative search technologies in dynamic data structures.

Materials and Methods

The research methodology involved the development of a software solution for searching for analogous terms in terminological knowledge bases (TKB). The key element of this solution was the introduction of a similarity coefficient between terms, which allows assessing their semantic proximity. The formula for calculating the coefficient considers the number of terms in the semantic chain and the weights of the links between them, factors in the function of determining the semantic value, and the similarity coefficient is defined as the ratio of the number of possible links and the value of the semantic significance to the number of terms connecting a pair of terms:

$$minlength(T_1, T_2) = \min\left(\frac{\sum_{i=1}^{N-1} W_i}{N}\right), \quad (1)$$

where $minlength(T_1, T_2)$ is the function that determines the semantic value of terms T_1 to T_2, T_1 and T_2 are the terms belonging to the same TKBs, between which it is necessary to establish the similarity, $W_i \in \{0,1]$ is the weight of the semantic relationship between neighbouring terms that are part of a chain between terms T_1, T_2, N is the number of terms in the chain between the terms T_1, T_2.

$$S_{T_1T_2} = \frac{N_w * minlength(T_1,T_2)}{N_T}, \quad (2)$$

where N_w is the number of possible semantic chains between terms T_1 *and* T_2, N_T is the number of terms in the chain between the terms T_1, T_2.

To determine the set of terms that best reflect the context of the user's input query, the study analysed vector representations of the terms, which allows comparing them based on semantic relatedness. The vector representation of the data was evaluated using a matrix that reflects the relationships between the terms. This

approach helped to find the most relevant terms and assess their significance in the overall context. To improve the search efficiency, the coefficient of terms belonging to the TKB, namely to the group of root terms that form the kernel of the TKB, was used, which factors in the number and weight of links between terms, as well as their distance to the root term in the chain.

$$F(T_{kernel}, T_1) = \frac{\sum_{j=1}^{J} max\left(\sum_{i=1}^{N-1} s_{T_j T_i}\right)}{J}, i \neq main, \quad (3)$$

where $F(T_{kernel}, T_1)$ is the function that determines the semantic value of a term T_1 to a group of terms T_{kernel}; T_j *are* terms that have the greatest similarity in the TKB and form the TKB kernel; N is the number of terms in the chain between the terms T_i, T_j; J is the minimum number of terms that form the root of the TKB. This parameter can be changed according to the requirements of the method. This ensures that the search is limited to relevant knowledge bases, reducing data processing time and increasing the accuracy of the results by improving the filtering of similar terms.

To optimise the multi-target search, the study employed the Louvain algorithm, which ensures the dynamic formation of term clusters. It factors in the modularity of the graph, which allows filtering relevant terms, reducing the amount of data processed, and improving the accuracy of the answer. The algorithm works in two stages: local modularity optimisation and agglomeration. Modularity measures the density of connections within graph communities compared to a random distribution.

The effectiveness of the proposed approach was tested on three datasets. K-Means clustering, hierarchical clustering, and DBSCAN (Density-Based Spatial Clustering of Applications with Noise) algorithms were employed for comparison (Sutramiani *et al.,* 2024). In the case of TKBs, the K-Means algorithm was used to cluster terms based on their semantic similarity, where the distance between terms is determined by a similarity metric such as cosine similarity. Additionally, the similarity of vector representations of terms was evaluated using the formula of the average adjacency matrix of the vectors of the input query terms and the TKB terms.

To test the effectiveness of this method based on the combined clustering algorithm, a software implementation was developed considering the technical features of the test data, namely their distribution and preparation for use. The source of the sample data was data from the Kaggle platform (Gabriel, 2020). Each data sample was presented in the form of a TKB implemented on the neo4j graph database and containing over 10,000 terms each.

Results and Discussion

One of the key aspects of developing a software solution for finding similar terms is to introduce a similarity coefficient between two terms. This coefficient allows mathematically estimating how close two terms are in their meaning within the same TKB. The formula for calculating this coefficient factors in the number of terms in the semantic chain and the weights of the links between them. However, term similarity is only one stage of the process. The ultimate purpose is to identify the set of terms that best reflect the context of the user's input query. This is especially significant when working with natural language when a user's message may contain more than one term. In this case, it is necessary to analyse vector representations of term sets, which allows comparing terms based on their semantic proximity. The vector representation of data in this context can be evaluated using a matrix that reflects the relationships between terms. An example of such an analysis can be given to find similarities between vector representations of terms from a TKB. This approach allows not only finding the most relevant terms but also assessing their semantic significance in the overall context.

To improve the search efficiency, it is proposed to use the coefficient of terms belonging to the TKB. This coefficient factors in the number and weight of links between terms, as well as the distance to the root term in the chain. It allows limiting the search to only relevant knowledge bases, which substantially reduces processing time and increases the accuracy of the results. Additionally, the study considered possible options for the data structure for the search, since terms that are linked by semantic relationships involve the use of network data structures. The study also highlighted graphs and adjacency matrices, which act as a mathematical representation of the similarity of terms among themselves. The graph data structure is widely used in graph databases, which is one of the best solutions in the field of TKB data processing (Yuehgoh *et al.,* 2024). For effective search in graphs, breadth-first and depth-first algorithms, clustering algorithms, and ranking algorithms for web-based information systems are used.

A multi-purpose search method that factors in the context of a term and its semantic group should not depend on the subject matter of the information technology in which it is expedient to use it, and therefore ranking algorithms are not relevant. Depth search and breadth search are more acceptable search algorithms, but have a series of disadvantages, such as high dependence on the search volume and organisation of data in the TKB.

Among the known algorithms, the most promising for data retrieval in TKBs is the Louvain algorithm (Sattar & Arifuzzaman, 2018). It is designed to find groups of nodes (clusters or communities) that have more internal connections with each other than with the rest of the graph. The main positive feature of this algorithm is its high efficiency in working with large networks of TKB data due to its scalability and speed of data search. The purpose of the algorithm is to maximise a metric called modularity. Modularity measures the quality of graph partitioning into communities by comparing the density of links within communities with a random distribution coefficient of such links, and its implementation is divided into two stages: local modularity optimisation and agglomeration. In this algorithm, modularity is a measure that determines how well a graph is divided into communities. A high modularity means that there are

considerably more connections within each community than between communities. The modularity formula considers not only the number of links, but also their expected number in a random graph with the same vertex degrees, which allows comparing the quality of clustering with a random distribution of links. Despite a series of advantages, the disadvantages include sensitivity to the initial state, namely the initial organisation of data in the TKB, and searching in local graph maxima, which may overlook small groups of terms among large groups of terms.

The core value of Louvain's algorithm lies in providing optimisation for multi-purpose term search by dynamically forming clusters of terms in the TKB. Consideration of semantic similarity and modularity allows filtering relevant terms for search, reducing the amount of data processed and improving the accuracy of the answer. This increases the efficiency of the multi-target search algorithm, ensuring accurate and fast detection of terms that match the user's query in large amounts of information.

The multi-target search method uses the Louvain algorithm with the use of not only the modularity metric, but also the metric of similarity of terms among themselves and the metric of similarity with the TKB. Thus, the multi-purpose term search method provides a comprehensive approach to query processing in terminological knowledge bases. It combines the analysis of semantic relationships between terms, evaluation of vector representations, and selection of relevant TKBs, as well as the modularity of term groups, which allows working efficiently with large amounts of data and ensuring high accuracy of results when searching in several TKBs. To perform a search for similar terms, it is necessary to enter the similarity coefficient between two terms, which is determined by formulas (1) and (2).

The factual finding of the similarity of terms is only part of the overall solution, as it involves only the preparation of the TKB data. The factual search is performed by identifying a fixed set of terms that best reflect the context of the user's input, namely the terms with the highest similarity coefficient. It should also be understood that most of the TKB data contains terms that include not only words, but also sentences containing more than one term, and therefore it is necessary to determine the similarity of vector representations of the term sets, which is represented in the form of a matrix. The similarity of vector representations is defined as the arithmetic mean of the adjacency matrix of the vector of terms of the input message and the vectors of terms of the data from the TKB, which is presented in the following formula:

$$S = \frac{1}{N*M}\sum_{i=1}^{N}\sum_{j=1}^{M} A_{ij}. \quad (4)$$

For instance, let us determine the similarity of vector representations for the vector of terms of the incoming message (T_1, T_2, T_3) and the vector representation of terms from the TKB (T_A, T_B, T_C, T_D, T_E). The calculation results are presented in Table 1.

Table 1. Example of searching for similar terms

	T_A	T_B	T_C	T_D	T_E
T_1	0.912	0.823	0.754	0.843	0.761
T_2	0.856	0.904	0.783	0.735	0.819
T_3	0.759	0.831	0.905	0.861	0.913

Source: developed by the authors

$$S = \frac{1}{15} * (0.912 + 0.823 + 0.754 + 0.843 + 0.761 + 0.856 + 0.90 + 0.783 + 0.735 + 0.819 + 0.759 + 0.831 + 0.905 + 0.861 + 0.913)$$

$$S = \frac{1}{15} * 12.458 = 0.831.$$

Considering the Louvain algorithm in terms of its application within a search within a single TKB, the first stage is local modularity optimisation. At this stage, the algorithm analyses each node and tries to move it to a neighbouring community if this leads to an improvement in modularity. The process consists of the following steps:

1. For each node, the algorithm calculates the change in modularity that will occur when this node is moved from its current community to one of the neighbouring communities.

2. If moving a node to a neighbouring community produces an increase in modularity, the node is moved to that community.

3. This process is repeated for all nodes in the graph until the condition of local modularity maximisation is met, i.e., when no more moves can improve modularity.

Notably, at each step, modularity is calculated using the formula 5 (Sattar & Arifuzzaman, 2018):

$$Q = \frac{1}{2m}\sum_{ij}\left[A_{ij} - \frac{k_i k_j}{2m}\right]\delta(c_i, c_j), \quad (5)$$

where Q is the modularity, m is the number of edges in the graph, k_i, k_j are the degrees of nodes i and j, A_{ij} is the element of the graph adjacency matrix, $\delta(c_i, c_j)$ is a function equal to 1 if nodes i, j belong to the same group and 0 if they belong to different groups. The purpose of the algorithm is to maximise the value of Q, which means that the graph is divided into communities in such a way that there are more links within each community than between different communities. This enables better cohesion of terms within a community and makes it easier to find relevant terms. Louvain's algorithm is an effective tool for working with large TKBs, as it can handle large networks of terms with numerous relationships. The main advantage of the algorithm is its scalability and speed. After forming clusters based on the modularity metric, the algorithm provides an opportunity to optimise the search by using semantic similarity

within each community. This allows the search to focus on relevant terms within each cluster, considerably reducing the amount of data to process. Louvain's algorithm has its drawbacks: it can be sensitive to the initial state and easily gets caught in local modularity maxima, which can lead to the formation of too large or small clusters, especially when working with large graphs. However, the use of additional metrics, such as cosine similarity, can reduce the effects of these limitations and improve the accuracy of results in multicomponent TKBs.

By using the Louvain algorithm and applying metrics to determine the similarity of terms among themselves, it is possible to achieve faster search results by reducing the number of search iterations within the graph data structure. After the software implementation, it is worth testing with the analogue methods used to search for terms in the TKB.

The main advantage of the K-Means algorithm is its speed and ability to process large amounts of data in a relatively small number of iterations. However, the algorithm has a series of limitations in the context of TKBs:

1. Sensitivity to the initial choice of K: The algorithm requires a predefined number of clusters (K), which can be difficult to predict in the context of dynamic TKB data.

2. Ignoring connectivity: K-Means does not consider the existence of relationships between terms and works solely based on distance to the centroid. This leads to the loss of information about the internal relationships between terms that are important in TKBs, which contradicts the purpose of the study, namely, multi-target search.

3. The presence of local minima: there is a possibility of a closed loop in the local minima, which will lead to the absence of an optimised solution, especially when there are many terms and complex semantic relationships.

Considering these limitations, the K-Means algorithm in TKB should be supplemented with the Louvain algorithm, which better accounts for the connectivity of terms and can determine the number of clusters dynamically. For clustering terms in a TKB, the Louvain algorithm has major advantages over K-Means because it uses a graph structure to distribute terms based on internal connectivity. The purpose of the Louvain algorithm is to maximise modularity, which considers the density of internal connections between terms. This provides a more natural distribution of data, especially in large graph structures where terms have a complex network of connections. At the same time, in tasks where clustering is based on semantic distance without complex relationships, K-Means can show high performance and be useful for the initial distribution of terms.

Hierarchical clustering and DBSCAN are also worth considering. Hierarchical clustering is based on the creation of a cluster tree structure. This approach allows analysing terms at multiple levels and identifying subclasses within large clusters, which can be useful for complex multilevel TKBs. DBSCAN creates clusters based on the density of points in space, which makes it useful for finding closely related groups of terms and isolating "noise" or terms that do not belong to any cluster. However, among the disadvantages of hierarchical clustering is its high computational complexity, which requires extensive computing resources, especially for large data sets, which are usually TKBs. This can create problems for scalability, as time and computational costs increase substantially with the number of terms. Another disadvantage is the lack of the ability to re-form clusters, namely, to change the structure after it has been built, which does not allow changing clusters when new data becomes available. This limits its use in dynamic TKBs where the database is constantly updated with new terms. Additionally, there is a disadvantage associated with low noise immunity, which can lead to the emergence of unwanted clusters, which is levelled by adding term filtering. Among the disadvantages of DBSCAN are the difficulty of working with multidimensional data, namely in graph data structures, where the dimensionality is determined by the number of edges, and the disregard for weak relationships between terms within even one TKB.

Considering the advantages and disadvantages of each method, a combined approach to clustering terms in TKBs was chosen, where K-Means is used for pre-clustering large amounts of data, while the Louvain algorithm is applied at the second stage to optimise internal links within each cluster. This strikes a balance between processing speed and clustering accuracy, ensuring that terms within each cluster are relevant. Firstly, K-Means is used to create initial clusters based on the semantic distance between terms, which reduces the amount of computation. Then, within each of these clusters, the Louvain algorithm is applied to refine the groups based on the relationships between the terms and improve the quality of the clustering. A diagram of the combined algorithm is presented in Figure 1. This approach not only improves the clustering accuracy but also enables efficient data processing in large TKBs, where the number of terms and their relationships can vary considerably.

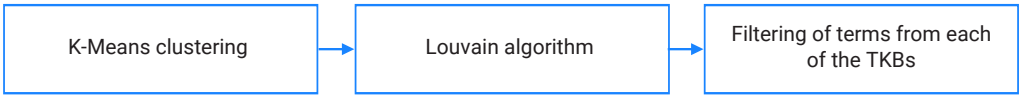

Figure 1. Combined clustering algorithm

Source: developed by the authors

A detailed description of the experimental results is presented in Table 2, where the developed method is called MultiSearch. The point of this experiment is to determine the optimised data distribution and the effectiveness of using iterations to find the corresponding clusters. Moreover, the cluster formation is based

on the fulfilment of the term relatedness condition and the determination of modularity by the Louvain algorithm. The hardware specifications for all methods are the same, and the software implementation was created using the Python 3.10 programming language and the TensorFlow package. The samples consist of prepared datasets formed in the form of graph databases. To reduce the impact of hardware and technical errors, a series of experiments was conducted 100 times. The data in the table are averages of all experiments.

Table 2. Experimental results

	Sample 1, s	Sample 2, s	Sample 3, s	Clusters	Iterations
DBSCAN	1.504	1.624	1.977	78	194/362/499
Hierarchical clustering	1.417	1.829	1.863	77	271/357/432
K-Means	1.105	1.272	1.715	84	175/282/335
MultiSearch	0.961	1.138	1.469	89	174/197/234

Source: developed by the authors

The experiments confirmed the effectiveness of the developed method when working with different datasets represented in graph databases. The findings demonstrated that the proposed multi-objective search method outperforms other clustering-based search methods, providing an average of 11.78-16.75% more efficient search while maintaining high quality cluster formation. The developed method proved its effectiveness in multiple tests, demonstrating stable results due to optimisation based on modularity and term affinity. The proposed combined clustering approach using K-Means and Louvain algorithms provided an effective balance between processing speed and clustering accuracy, especially for large graph databases.

One of the key stages in the formation of TKBs is the use of graph data structures and clustering. C. Li *et al*. (2022) described the use of intelligent search engines based on knowledge of graph structures, where each term is represented as a node in a graph, while the links between them are represented as edges. This allows building networks of links between terms, specifically for large databases where the number of links considerably exceeds the number of terms. The findings of the present study proved the effectiveness of using graph data structures for multi-criteria search, which was a convincing argument for using graph data structures in TKBs.

The use of graph structures greatly improves performance in multi-component systems, which is confirmed by the findings of S. George *et al*. (2019), where the graph structure showed high efficiency in processing large amounts of data due to its scalability. During the review of modern solutions in the field of term clustering, attention was focused on finding an efficient algorithm with scaling stability. Thus, S. Sattar & S. Arifuzzaman (2018) characterised the Louvain algorithm, designed to identify communities in large graphs, as one of the most effective clustering tools for term knowledge bases. This algorithm allows dividing a graph into a set of subgraphs, each of which is characterised by a high density of internal connections (modularity). Thus, Louvain's method provides efficient clustering that increases the relevance of searching in TKBs, since each cluster can be considered as a group of terms with strong semantic relationships. The disadvantages of the algorithm are its dependence on the initial state of the data in the graph and the difficulty in recognising smaller groups among large communities, which was emphasised by Y. Zhang *et al*. (2024). Considering the shortcomings of its application, in this study, their influence was not critical, since effective filtering of terms by the affinity criterion levelled the main drawback, namely, sensitivity to small groups in term clustering.

The multi-purpose search approach, which combines clustering and semantic similarity methods, is a modern trend that is gaining popularity due to its ability to simultaneously process several criteria. According to Y. Zhao & T. Wang (2021), this approach allows not only to account for the structured nature of the data, but also semantic relevance, which increases the search accuracy in dynamic knowledge bases. The main advantage is the ability to process queries considering many factors, such as data format, structure, and sample size, which makes it effective for various fields, including medicine and technical support. The use of artificial intelligence, specifically deep learning, greatly improves the accuracy of searching in TKBs, enabling the analysis and classification of large amounts of data automatically. H. Baqal & M. Sidiq (2024) noted that modern AI models can not only find relevant terms, but also learn based on previous queries, improving performance with each use. This is particularly relevant for chatbot applications, where systems can automatically update the knowledge base with new data and improve the accuracy of responses to users. Incorporating artificial intelligence into the term search process in a TKB provides systems with the ability to be adaptive, which is crucial in dynamic environments.

AI has made it possible to considerably improve the accuracy and speed of finding relevant data, which was best described by N.F. Lindemann (2024). The use of machine learning and deep learning techniques allows building complex models of semantic similarity between terms, which improves the quality of search results. AI algorithms can analyse large amounts of data, identify hidden relationships between terms and determine their relevance to a concrete user query. At the same time, multi-targeted search is becoming increasingly important, allowing

several factors to be factored in simultaneously when searching for information, such as semantic relationships between terms, the context of a user's query, and whether the terms belong to a particular subject area. This is crucial to ensure accurate and fast access to data in the face of the diversity of information stored in TKBs.

Intelligent systems capable of recognising, classifying, and interpreting data are becoming a prominent part of the modern information infrastructure. A. Yarovyi & D. Kudriavtsev (2021) focused on the use of neural networks and machine learning technologies for text processing tasks, namely classification and context detection. Additionally, the researchers proposed to combine the use of optimised semantic text analysis and recurrent neural network methods as one of the examples of effective data analysis and context detection. The relevance of such a method was confirmed in the context of developing chatbots specialising in the search for relevant information, as described by A. Morayo *et al.* (2024). Another striking example of intelligent systems is decision support systems, which also require fast and accurate term search in a large amount of information (Gupta & Singh, 2024). The method of multi-purpose term search in TKBs can be widely used in the development of intelligent information systems, as discussed by D. Beeram (2024), which help users quickly find the information they need and make informed decisions in various fields of activity. The multi-objective search method proposed in this study allows combining the strengths of various methods, ensuring high relevance and accuracy of the results.

Conclusions

The present study developed a method for multi-purpose term retrieval in terminological knowledge bases, which combines the analysis of semantic relations between terms and vector representations of terms. The proposed method allows accounting for various factors during the search, including the structure of the TKB, semantic similarity of terms, and the context of the user's query. This method is based on the idea of combining the K-Means clustering algorithm and the Louvain algorithm to optimise search processes, which greatly improved the accuracy and speed of search when working with large amounts of data. The use of modern algorithms, such as the Louvain algorithm for community detection and term clustering, helped to work effectively with large graph databases and ensure high search performance. Furthermore, it was found that the integration of the Louvain algorithm and term similarity coefficients greatly improves search results, reducing the amount of data processed and focusing the search on the most relevant terms. Therewith, the number of clusters formed indicates the advantage of using the combined approach, since a greater number of clusters with the term similarity coefficient as a filter for the appearance of unwanted terms indicates a greater diversity of the context of terms in the TKBs. The experiments demonstrated the effectiveness of the proposed method when working with several data samples implemented in graph databases. The obtained findings revealed that the multi-objective search method is competitive in comparison with other clustering-based search methods, being on average 11.78-6.75% faster than the best result. This method provides a comprehensive approach to query processing in TKBs and can be applied in various industries requiring fast and accurate access to specialised terminology. Further research could focus on identifying patterns and models of communication between data organisation structures in TKBs, including, apart from graph and network data structures, hash tables to improve the efficiency of multi-purpose search.

Acknowledgements

None.

Conflict of Interest

The authors declare no conflict of interest

References

[1] Abdykerimova, L., Abdikerimova, G.B., Konyrkhanova, A., Nurova, G., Bazarova, M., Bersugir, M., Kaldarova, M., & Yerzhanova, A. (2024). Analysis of the emotional coloring of text using machine and deep learning methods. *International Journal of Electrical and Computer Engineering (IJECE),* 14, article number 3055. doi: 10.11591/ijece.v14i3.pp3055-3063.

[2] Baqal, H., & Sidiq, M. (2024). Graph databases: Revolutionizing database design and data analysis. *Current Journal of Applied Science and Technology*, 43, 45-56. doi: 10.9734/cjast/2024/v43i114443.

[3] Beeram, D. (2024). Combining deep learning and heuristic search for efficient text summarization. *International Research Journal of Engineering and Technology (IRJET),* 11(8), 23-34.

[4] Bienvenu, M., Bourgaux, C., & Jean, R. (2024). Cost-based semantics for querying inconsistent weighted knowledge bases. In *Proceedings of the 21st international conference on principles of knowledge representation and reasoning* (pp. 167-177). Hanoi: CAI Organization. doi: 10.24963/kr.2024/16.

[5] Bourgaux, C., Guimarães, R., Koudijs, R., Lacerda, V., & Ozaki, A. (2024). Knowledge base embeddings: Semantics and theoretical properties. In *Proceedings of the 21st international conference on principles of knowledge representation and reasoning* (pp. 823-833). Hanoi: International Joint Conferences on Artificial Intelligence Organization. doi: 10.24963/kr.2024/77.

[6] Gabriel, A. (2020). Kensho derived Wikimedia dataset. Retrieved from https://www.kaggle.com/datasets/kenshoresearch/kensho-derived-wikimedia-data.

[7] George, S., Elayidom, M.S., & Santhanakrishnan, T. (2019). Semantic desktop search engine using graph database. *International Journal of Recent Technology and Engineering*, 8(1S2), 373-375.
[8] Gupta, A., & Singh, T. (2024). Study of various frameworks to develop intelligent chatbots. *International Journal of Innovative Science and Research Technology (IJISRT)*, 9(4), 2969-2978. doi: 10.38124/ijisrt/IJISRT24APR1290.
[9] Kaya, C., Kilimci, Z.H., Uysal, M., & Kaya, M. (2024). A review of metaheuristic optimization techniques in text classification. *International Journal of Computational and Experimental Science and Engineering*, 10(2). doi: 0.22399/ijcesen.295.
[10] Li, C., Liang, M., & Qiu, D. (2022). An intelligent search system based on knowledge graph. In *2022 International conference on artificial intelligence of things and crowdsensing (AIoTCs)* (pp. 66-70). Nicosia: IEEE. doi: 10.1109/AIoTCs58181.2022.00017.
[11] Lindemann, N.F. (2024). Chatbots, search engines, and the sealing of knowledges. *AI & Society.* doi: 10.1007/s00146-024-01944-w.
[12] Mohabir, S.E., & Joshi, Y.C. (2024). A bibliometric analysis of the knowledge base on multinational corporations' behavior. *SN Business & Economics,* 4, article number 105. doi: 10.1007/s43546-024-00705-7.
[13] Morayo, A., Samuel, J., Kennedy, O., Adeyinka, A., Adenugba, A., & Imhade, O. (2024). Development of an artificial intelligent health chatbot for improved telemedicine. In C. So In, N.D. Londhe, N. Bhatt & M. Kitsing (Eds.), *Information systems for intelligent systems. ISBM 2023. Smart innovation, systems and technologies* (Vol. 379, pp. 585-600). Singapore: Springer. doi: 10.1007/978-981-99-8612-5_48.
[14] Rathje, S., Mirea, D.-M., Sucholutsky, I., Marjieh, R., Robertson, C., & Van Bavel, J. (2024). GPT is an effective tool for multilingual psychological text analysis. *Proceedings of the National Academy of Sciences of the United States of America,* 121, article number e2308950121. doi: 10.1073/pnas.2308950121.
[15] Roy, S., Bharaty, A., Sarkar, S., Sehgal, M., & Panchal, R. (2024). A hybrid ensemble approach for short-text sentiment analysis integrating deep learning and traditional machine learning methods. *ResearchGate*. doi: 10.13140/RG.2.2.15182.88643.
[16] Sattar, N.S., & Arifuzzaman, S. (2018). Parallelizing Louvain algorithm: Distributed memory challenges. In *2018 IEEE 16th Intl conf on dependable, autonomic and secure computing, 16th intl conf on pervasive intelligence and computing, 4th intl conf on Big Data intelligence and computing and cyber science and technology congress (DASC/PiCom/DataCom/CyberSciTech)* (pp. 695-701). Athens: IEEE. doi: 10.1109/DASC/PiCom/DataCom/CyberSciTec.2018.00122.
[17] Simian, D., & Şerban, M.-E. (2024). Improving search query accuracy for specialized websites through intelligent text correction and reconstruction models. *Information*, 15, article number 683. doi: 10.3390/info15110683.
[18] Sutramiani, N., Arthana, I.M.T., Lampung, P.F., Aurelia, S., Fauzi, M., & Darma, I.W.A.S. (2024). The performance comparison of DBSCAN and K-Means clustering for MSMEs grouping based on asset value and turnover. *Journal of Information Systems Engineering and Business Intelligence*, 10, 13-24. doi: 10.20473/jisebi.10.1.13-24.
[19] Wu, L., Hu, J., Teng, F., Li, T. & Du, S. (2023). Text semantic matching with an enhanced sample building method based on contrastive learning. *International Journal of Machine Learning and Cybernetics*, 14, 3105-3112. doi: 10.1007/s13042-023-01823-8.
[20] Yarovyi, A. & Kudriavtsev, D. (2021). Multi-purpose search to determine the context of a text message based on the dictionary data structure. In *2021 IEEE 16th international conference on computer sciences and information technologies (CSIT)* (pp. 65-68). Lviv: IEEE. doi: 10.1109/CSIT52700.2021.9648803.
[21] Yuehgoh, F., Djebali, S., & Travers, N. (2024). Leveraging recommendations using a multiplex graph database. *International Journal of Web Information Systems*, 20(5). doi: 10.1108/IJWIS-05-2024-0137.
[22] Zhang, Y. *et al*. (2024). A materials terminology knowledge graph automatically constructed from text corpus. *Scientific Data*, 11, article number 600. doi: 10.1038/s41597-024-03448-0.
[23] Zhao, Y., & Wang, T. (2024). Knowledge base embeddings for a recommendation based on overlapping knowledge and graph learning. *Arabian Journal for Science and Engineering*. doi: 10.1007/s13369-024-09573-7.

Метод багатоцільового пошуку термів в термінологічній базі

Андрій Яровий
Доктор технічних наук, професор
Вінницький національний технічний університет
21021, вул. Хмельницьке шосе, 95, м. Вінниця, Україна
https://orcid.org/0000-0002-6668-2425

Дмитро Кудрявцев
Аспірант
Вінницький національний технічний університет
21021, вул. Хмельницьке шосе, 95, м. Вінниця, Україна
https://orcid.org/0000-0001-7116-7869

Анотація. У статті досліджувався метод багатоцільового пошуку термів у термінологічній базі знань, який базується на семантичному аналізі та використанні сучасних методів обробки природної мови. Розглянуто ключові фактори, що впливають на ефективність пошуку, зокрема структуру організації даних, формат і параметри даних, а також обсяг вибірки. Особлива увага була приділена семантичній подібності між термами, що дозволяє підвищити точність пошуку за рахунок векторних представлень та алгоритму Лувена. У статті також описано застосування косинусної подібності для кількісної оцінки подібності між термами. Крім того, оптимізовано процес пошуку шляхом фільтрації релевантних баз даних і динамічного визначення релевантних термів за допомогою метрики модульності. Виконано порівняльний аналіз наявних методів пошуку термів за визначеними факторами. Відзначено переваги та недоліки використання алгоритму Лувена у порівнянні з алгоритмами пошуку в графових структурах даних. Виконано ряд експериментів на вибірках даних, включаючи словникову структуру даних, графову та мережеву структуру даних. Проаналізовано використання логістичних обмежень для пошуку в мережевих структурах даних та відзначено можливість оптимізації за рахунок рівномірного та динамічного розподілу даних. Результати експериментів показали ефективність застосування комбінації алгоритму Лувена та мережевих структур даних в термінологічних базах знань. Подано приклади сфери застосування даного методу в інформаційних технологіях пошуку та обробки текстових даних. Розроблено схему архітектури програмного забезпечення із використанням програмного інтерфейсу та можливості інтеграції для веб-застосунків у вигляді пакету чи бібліотеки. Пропонований підхід продемонстрував ефективність у контексті інтелектуальних систем підтримки рішень і автоматизованих чат-ботів, що робить його особливо корисним для галузей, де критично важливий доступ до точних фахових термів. Розроблено базову версію програмного інтерфейсу для використання даного методу в інформаційних технологіях пошуку та аналізу даних для використання в пошукових системах

Ключові слова: термінологічна база знань; семантична подібність; алгоритм Лувена; векторні представлення; обробка природної мови