UDC: 004.82:004.89 https://doi.org/10.15407/jai2024.04.089

A. Litvin[1], V. Kaverinsky[2], D. Simonov[3]
[1,3]V. M. Glushkov Institute of Cybernetics of NAS of Ukraine, Ukraine
40, Akademika Glushkova Avenue, Kiev, 03187
[3]Frantsevich Institute for Problems of Materials Science of NAS of Ukraine, Ukraine
3, Omelyan Pritsaka street, Kiev, 03142
[1]litvin_any@ukr.net
[2]insamhlaithe@gmail.com
[3]denys.symonov@gmail.com
[1]http://orcid.org/0000-0002-5648-9074
[2]https://orcid.org/0000-0002-6940-579X
[3]https://orcid.org/0000-0002-6648-4736

THE USE OF LARGE LANGUAGE MODELS IN COMBINATION WITH THE ONTOLOGICAL APPROACH FOR THE SYNTHESIS OF NATURAL LANGUAGE TEXT

Abstract. This research presented an approach based on the application of structured prompt instructions to large language models (LLM). Methodological foundations for using an ontology-driven system of structured prompts in interaction with ChatGPT were developed. The ChatGPT system allows expanding the knowledge base by obtaining new information from existing knowledge units based on a set of contexts. Thus, methods for generating meaningful responses in natural language were considered using a large language model using ontological approaches and natural language contexts.

Using the proposed methodology, the OntoChatGPT system was developed, which effectively extracts entities from contexts, classifies them and generates appropriate responses. An experiment on the reverse synthesis of natural language sentences based on their ontological representation using large language models allows to clearly demonstrate the effectiveness of using the concept of large language models in dialogue systems. The study highlights the versatility of the methodology, emphasizing its applicability not only to ChatGPT but also to other chatbot systems based on LLMs, such as Google's Bard utilizing the PaLM 2 LLM. The implementation of this technology is demonstrated using the Ukrainian language.

Keywords: ontology engineering; prompt engineering; prompt-based learning; meta-learning; ChatGPT; OntoChatGPT; chatbot; ontology-driven information system.

Introduction

An approach based on the application of structured instructions-prompts for large language models formed using meta-ontology was created to solve a wide range of problems of natural language text analysis, including obtaining answers based on contexts selected with the help of appropriate formal queries from the ontological knowledge base.

Nowadays, there are many virtual assistants that use natural language processing. One of these, ChatGPT (which is based on such basic models as GPT OpenAI GPT-3, GPT-3.5, and GPT-4 [1] and has been customized for conversational applications using both supervised and reinforcement learning methods), marked a significant breakthrough in the field of artificial intelligence, particularly in the areas of natural language processing (NLP) and understanding (NLU).

ChatGPT is a general-purpose dialog system. It performs certain tasks for processing textual information: determining intentions, named entities, building logical conclusions, searching for information, and even style correction and machine translation tasks, etc.

The ChatGPT model provides the ability to interact through an API, which involves sending commands in natural language. English is the preferred language for such commands and instructions. This is because English is the primary language for ChatGPT.

Research has shown that formatting instructions in JSON is an effective approach to provide clear and comprehensive model commands.

The urgency of the problem is due to the fact that large models such as ChatGPT seem to work very efficiently, but their knowledge base is not comprehensive. Training such models requires a large set of labeled data and significant computing resources. Thus, the task is to create a methodological foundation for using an ontologically guided system of structured prompts in interaction with ChatGPT. The system allows to expand the knowledge base by obtaining new information from existing knowledge units based on a certain set of contexts.

The subject of the research is to expand the knowledge base of large ChatGPT language models by using an ontological approach to generate meaningful text.

Research methods. Prompting instruction engineering for large language models. Determination of criteria for the quality of systems by metric: cosine similarity index.

Problem statement

At first glance, large language models such as ChatGPT seem to work very efficiently and are capable of performing a wide range of tasks. But, at the current time, they are not without certain drawbacks. Large language models do have a broad knowledge base, but it is not comprehensive. There are many narrow subject areas that are not represented in their current form or are poorly represented. This is usually encountered when working with ChatGPT, when questions/tasks begin to relate to narrow scientific and technical areas where general knowledge is insufficient or outdated (updating the knowledge base of models is a long and costly process that cannot happen as often as we would like). Such models have mechanisms for retraining and fine-tuning, but they also face certain difficulties. Re-training requires a large set of labeled data, the creation and preparation of which is difficult and painstaking. It also requires significant computing resources and takes a lot of time. Thus, especially in conditions of limited resources and the presence of tasks that can be solved in other ways, large language models become too much of a “big gun” to “shoot at sparrows”. In addition, not all natural languages are currently represented in such models at the same level. They work most efficiently with English, which is the language with the largest data sets. With less common and less popular languages, the results are somewhat more modest. Thus, in some cases, it seems easier and more appropriate to create a simpler model, including a rule-based model.

Moreover, by leveraging the strengths of different methods, there are broad prospects for combining approaches that may, in certain cases, achieve even greater efficiency in their use. Thus, the task is to create formal models (both informational and functional) and develop methodological foundations to use of an ontologically controlled system of structured prompts in interaction with ChatGPT. This system allows for the provision of information and the expansion of the knowledge base by obtaining new information from existing knowledge units based on a specific set of contexts. Logical deduction suggests obtaining new information based on established facts, rules and logical principles. This allows the system to draw logical conclusions and make connections between different pieces of information. By using deductive reasoning, the system can expand its understanding and create additional units of knowledge that have not been explicitly provided. The process of deriving new information units from previously known ones plays a crucial role in increasing the system's knowledge and improving its overall functionality. It allows the system to make intelligent inferences, discover hidden relationships, and provide the user with more complete and valuable information.

By developing formal models, this research provides a structured base for organising and representing knowledge in a systematic way. Such models, which cover both informational and functional aspects, lay the foundation for effective integration of the ontological approach with ChatGPT. The combined system provides a sophisticated dialog interaction that incorporates inferences and uses contextual information to provide meaningful responses.

Analysis of the last publications

The instructions-prompts provided by ChatGPT must be structured in a clear and precise structure. Since the number of tokens that ChatGPT can process is limited, the instructions should be short, but at the same time informative. Experimental evidence [2] – [5] has revealed that one of the effective approaches is to present them in the JSON format. Additionally, the instructions must convey enough information to accurately instruct ChatGPT. This ensures that the system's responses match the intentions of the user's expectations.

The ChatGPT system offers an API that only accepts commands in natural language. Although ChatGPT API supports several languages, but in order to form accurate and clear instructions, it is advisable to use English. These approaches improve the interaction between the user and the system, facilitating more accurate and meaningful answers, compared to the train-of-thought reasoning technique [4], [6] - [9].

An example set of instructions can be found in the public repository «Mr. Ranedeer: Your personalized AI Tutor!» [2] aimed at converting GhatGPT into a virtual tutor. Thus, ChatGPT can be adapted to work as a virtual tutor in various subject areas covered by its knowledge base.

GhatGPT system provides the ability to connect third-party plugins [10] – [12]. The GhatGPT system provides the ability to connect third-party plugins. This opens up a new direction of research, referred to as prompt engineering and meta-learning [6], [8], [13] – [15].

The main goal of prompt engineering is to solve the problem of guiding ChatGPT to appropriate answers, especially in problems that require logical inference. Designing instruction prompts involves carefully choosing specific words, phrase structures, and their order to elicit the desired AI behavior. However, to date, the question of the most effective strategy for developing instructions-prompts remains open, with some approaches described in the [5], [16].

While mechanisms such as model training and fine tuning exist in ChatGPT, they can be expensive and require large, carefully selected datasets.

The use of ontologies or metaontologies as a repository of system behaviour rules is discussed in [16]. In this approach, the ontology serves as a decision-making module, guiding ChatGPT on how to process certain types of data and present them in the user interface. By incorporating ontologies to specify ChatGPT behavior and using structured prompts, we can develop a powerful ontology-driven system. This system enhances ChatGPT's ability to adapt and learn in specific domains by exploiting the flexibility of prompts and capitalizing on the knowledge stored in the ontology.

The fundamental concepts of information systems with ontology-driven architecture are discussed in detail in [17], [18]. An ontology-driven information system includes several key elements:

– knowledge base (usually represented as a finite set of systematically integrated knowledge bases in specific subject areas);
– output mechanism;
– application processing subsystem;
– interfaces (UI, API) for user interaction and/or integration with the external environment.

However, it is important to note that the issues associated with these methods, such as ensuring accurate entity extraction and generating accurate SPARQL queries, are complex and beyond the scope of the current research [19] – [21].

In summary, in a classic sense, an ontology serves as the main repository of information in the system, and not as a container for rules and instructions for organizing the work of the system [22, 23, 24]. Nevertheless, the considered approach includes elements of ontological engineering. For example, such methods as extraction of named entities, analysis of the associated context, and automatic generation of formal SPARQL queries [22] from natural language phrases provided by the user.

The aim of research

Develop methods for generating meaningful natural language answers involving a large language model using ontological approaches and natural language contexts.

The main material

Using a large language model as a component of an ontologically guided dialog system.

The presented methodology can be divided into two key components, each of which has a separate purpose in the development of the OntoChatGPT system. This system enables the provision of information and inference (according to the definition given in «The explanatory ontograph dictionary for knowledge engineering» – expanding the knowledge base by deriving new information from existing knowledge units; this process includes various operations, with logical deduction being a notable case) based on a specific set of contexts, functioning as a dialogue system. Logical deduction involves inferring new information based on established facts, rules, and logical principles.

First, a prompt-based meta-learning technique was developed to generate structured prompts for ChatGPT. Structured prompts are used to instruct ChatGPT's meta-learning process, allowing it to generate more contextual and accurate answers. We delve into the methodology behind the development and implementation of these prompts, highlighting their importance in improving ChatGPT's conversational capabilities.

The second part of the methodology focuses on the development of an automatic ontology-driven dialog system that combines ChatGPT and structured prompts. The main idea of this system is to bring together specific subject areas and related contexts that may contain domain-related information that is not fully covered in the ChatGPT knowledge base. These contexts are stored in a database, such as MongoDB or a relational database, and are associated with sets of named entities with their own ontological structure. In addition, sentiment analysis can be used to classify contexts. The binding of named entities to their respective contexts includes semantic components that specify the entity's role in the context.

These additional features aim to increase the relevance and clarity of the selected context for further processing. To automate these processes, we use our previously developed tools [17, 18, 25, 26] and incorporate pre-trained BERT-based transformer models such as [27].

Formation and use of structured prompt instructions. ChatGPT proves to be a valuable resource for semantic analysis and extraction of named entities from user-supplied phrases. Structured prompt instructions are created specifically for this purpose. In addition, ChatGPT is used to analyze the intentions expressed in user phrases. The identified intentions and their corresponding named entities, as well as a list of contexts selected from the knowledge base, are then provided as input to ChatGPT. These inputs are accompanied by appropriate structured prompts that explain the information to be retrieved and the desired presentation format.

One of the features of the system is the flexibility of its structured prompts for ChatGPT. Instead of hard-coded prompts, they are dynamically generated based on the specific situation using instructions provided in the form of a meta-ontology. This meta-ontology describes the fields that should be included in the JSON (or XML) structure and the corresponding prompt phrases that should be inserted. Each instruction or structured prompt for ChatGPT has its own set of fields and predefined values that can be included in it. In addition, the prompt contains a template structure for the answer, which ensures consistency and simplifies further processing. The generation of prompt phrases uses proven techniques from [7, 12] to produce efficient and coherent instructions.

The prompt can contain various possible intentions, such as “quantity”, “method of action”, “object”, “subject”, “action”, “place”, “direction”, “place of action”, “conditions”, “instrument”, “cooperation”, “relation”, “reason”, “sequence”, “origin”, etc. These intents represent semantic categories and provide a basis for understanding the user request. In addition, the structured prompt contains fields for the information to be provided, the language used, and other technical details related to input and output. Named entity detection is performed by ChatGPT using a different instruction. The result should provide lemmatized words grouped by entities, indicate the type of each group (noun or verb), and specify the main word in each group.

The results of such an analysis can then be used in the respective dialog information system to implement the processing of selected contexts in order to synthesize the answer in accordance with the intentions expressed by the user in the text transmitted to the system.

The concept of the OntoChatGPT dialog system, which includes a combination of an ontological approach and a large language model. To provide a visual representation of the overall system diagram, we provide a context/container C4 model diagram, as shown in Figure 1. This diagram provides a comprehensive overview of the system architecture, showing the interactions between the various components and their relationships. It serves as a visual guide to understanding the basic structure and functionality of the OntoChatGPT system.

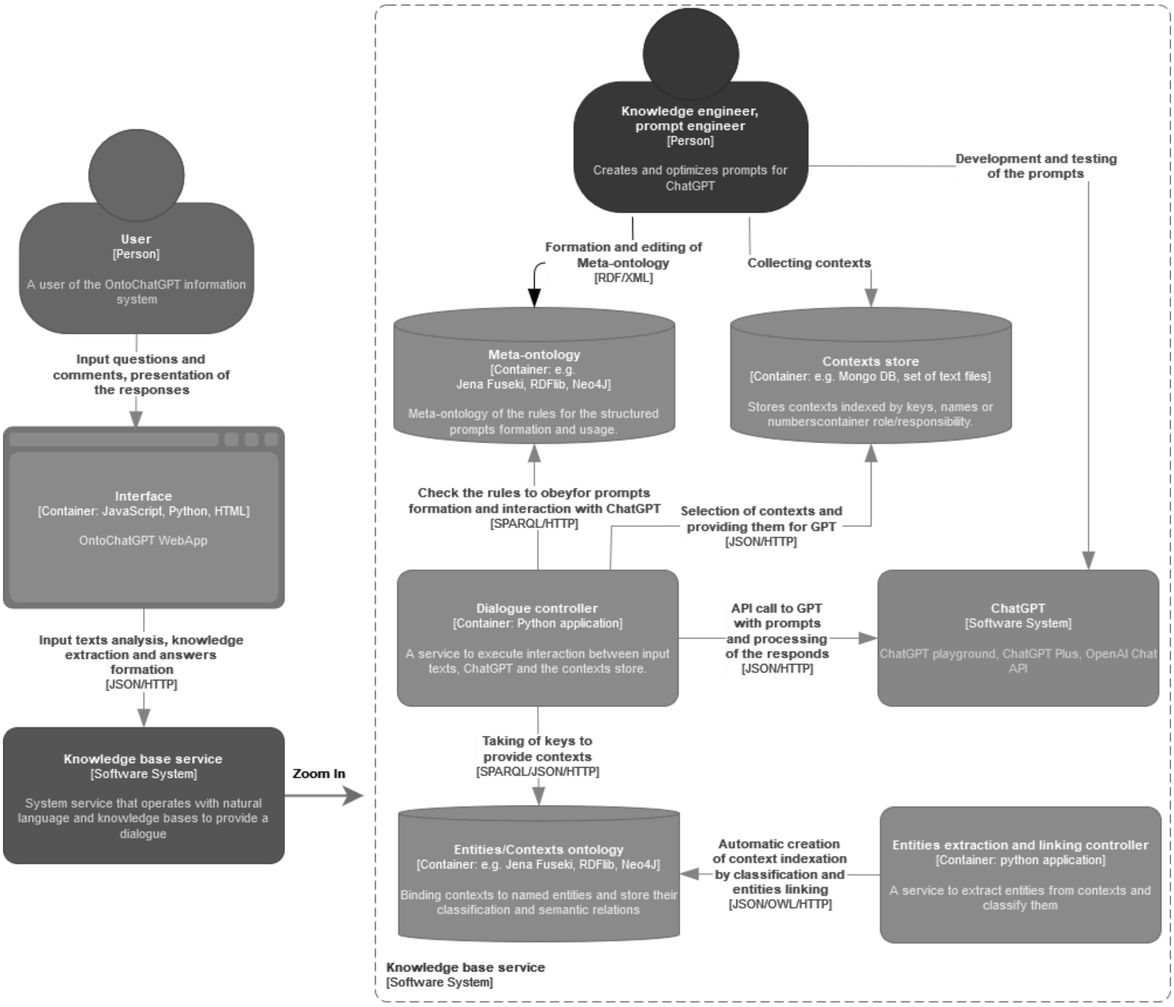

Fig. 1. Context/container C4 model diagram of the OntoChatGPT information system.

Discussing

To confirm the results obtained in the work, an experiment was conducted aimed at reverse synthesis of natural language sentences based on their ontological representation using a large language model.

The essence of the experiment was as follows. From the test ontology created on the basis of the text «Склад обчислювальної системи», individual sentences and their corresponding pairs of entities with semantic categories that combine them within a given sentence were extracted using a Cypher query. Then, with the help of a special instruction-prompt, the big language model (ChatGPT) was given the task of creating a grammatically correct sentence in Ukrainian based on a set of entity pairs given the specified semantic relations between them. The instruction-prompt provides an explanation of the model, what exactly the available semantic categories represent, in particular, in the aspect of creating a sentence. The original sentence itself was not submitted to the system.

The result was the generated sentence and the model's own estimate of the probability that the sentence was reproduced correctly.

10 sentences from the specified text were used for the test.

To compare the sentence formed on the basis of the ontological representation with the original one, the cosine similarity index was applied.

The values of the quantitative estimates characterizing the closeness of the sentence formed with the help of a large language model to the original are shown in Table 1. The table compares the values of the obtained cosine similarity under the conditions of different methods of vector representation of the analyzed sentences (original and formed).

Table 1. Quantitative assessments of the quality of reverse synthesis of sentences from ontological representation

Estimating the probability of correct reproduction of the ChatGPT phrase		Cosine similarity					
		Model xx_ent_wiki_sm		Model uk_core_news_lg		Method tf-idf	
Mean value ± confidence interval	Variation interval	Mean value ± confidence interval	Variation interval	Mean value ± confidence interval	Variation interval	Mean value ± confidence interval	Variation interval
0,845 ±0,037	0,75 – 0,90	0,8716 ±0,0335	0,8193 – 0,9722	0,8108 ±0,1224	0,4067 – 0,9653	0,2927 ±0,1718	0,0607 – 0,7745

The table shows that the language vectorization models xx_ent_wiki_sm and uk_core_news_lg lead to fairly high values of cosine similarity (0.8716 and 0.8108, respectively). At the same time, a simpler vectorization method based on tf-idf yields significantly lower average values and a large range of variation.

A combined system called OntoChatGPT was created that uses large language models, including the ChatGPT API, and an ontological knowledge base for meta-learning and improving such models. The peculiarity of this system is the presence of a meta-ontology that creates structured prompt instructions for a large language model. This approach significantly enhances the capabilities of a strong language model by providing it with access to selected pieces of specialized information from a narrow subject area, which increases the content and relevance of the answers provided by the mode.

Conclusions

A combined system called OntoChatGPT was created that uses large language models, including the ChatGPT API, and an ontological knowledge base for meta-learning and improving such models. Thepeculiarity of this system is the presence of a meta-ontology that creates structured prompt instructions for a large language model. This approach significantly enhances the capabilities of a strong language model by providing it with access to selected pieces of specialized information from a narrow subject area, which increases the content and relevance of the answers provided by the model.

An experiment was conducted on the reverse generation of natural language sentences based on their semantic representation in an ontological knowledge base, for which a large language model (ChatGPT) was used and corresponding instructions-promts were created for it. Although the results of the experiment show a fairly high cosine similarity between the original and generated sentences and the general meaning of the original is generally preserved, there is a noticeable distortion of the form and style of the phrase. This approach confirms the relevance of text generation systems based on ontologies with rules and templates, which can provide better results when solving this particular task compared to large language models.

References

1. Models - OpenAI API [Electronic source] // OpenAI Platform, 2023, Source access mode: https://platform.openai.com/docs/models/overview. (accessed Jun. 01, 2023).

2. JushBJJ,"JushBJJ/Mr.-Ranedeer-AI-Tutor: A GPT-4 AI Tutor Prompt for customizable personalized learning experiences.," GitHub, Jun. 01, 2023. [Online]. Available at: https://github.com/JushBJJ/Mr.-Ranedeer-AI-Tutor (accessed Jun. 01, 2023).

3. {Structured} Prompt, "Structured JSON Prompts are even better in GPT-4.," {Structured} Prompt. [Online]. Available at: https://structuredprompt.com/structured-json-prompts-are-even-better-in-chatgpt-4/.

4. GPT 4 is Smarter than You Think: Introducing SmartGPT. [Online Video]. Available at: https://www.youtube.com/watch?v=wVzuvf9D9BU

5. JushBJJ, "Mr. Ranedeer,"JushBJJ's Substack, May 24, 2023. [Online]. Available at: https://jushbjj.substack.com/p/mr-ranedeer (accessed Jun. 01, 2023).

6. K. Hebenstreit, R. Praas, L. P. Kiesewetter, and M. Samwald,"An automatically discovered chain-of-thought prompt generalizes to novel models and datasets, " arXiv, May 04, 2023. doi: 10.48550/arXiv.2305.02897.

7. T. Kojima, S. S. Gu, M. Reid, Y. Matsuo, and Y. Iwasawa, "Large language models are zero-shot reasoners, » arXiv, Jan. 29, 2023. doi: 10.48550/arXiv.2205.11916.

8. J. Wei et al., "Chain-of-thought prompting elicits reasoning in large language models,"arXiv, Jan. 10, 2023. doi: 10.48550/arXiv.2201.11903.

9. S. Wiegreffe, J. Hessel, S. Swayamdipta, M. Riedl, and Y. Choi,"Reframing human - AI collaboration for generating free-text explanations,"arXiv, May 04, 2022. https://doi.org/10.18653/v1/2022.naacl-main.47.

10. OpenAI, "ChatGPT plugins," OpenAI blog, Mar. 23, 2023. [Online]. Available at: https://openai.com/blog/chatgpt-plugins.

11. OpenAI, "OpenAI plugins API," OpenAI Platform, Jun. 01, 2023. [Online]. Available at: https://platform.openai.com/docs/plugins/introduction (accessed Jun. 01, 2023).

12. S. Panda and N. Kaur, "Revolutionizing language processing in libraries with SheetGPT: an integration of Google Sheet and ChatGPT plugin,"Library Hi Tech News, 2023, https://doi.org/10.1108/LHTN-03-2023-0051.

13. S. R. Moghaddam and C. J. Honey, "Boosting theory-of-mind performance in large language models via prompting,"arXiv, Apr. 26, 2023. doi: 10.48550/arXiv.2304.11490.

14. D. Hendrycks et al., "Measuring massive multitask language understanding," arXiv, Jan. 12, 2021. doi: 10.48550/arXiv.2009.03300.

15. L. C. Magister, J. Mallinson, J. Adamek, E. Malmi, and A. Severyn,"Teaching small language models to reason," arXiv, Jun. 01, 2023. doi:10.48550/arXiv.2212.08410.

16. Y. Zhou et al., "Large language models are human-level prompt engineers," arXiv, Mar. 10, 2023. doi: 10.48550/arXiv.2211.01910.

17. Palagin A. V., "Architecture of ontology-controlled computer systems," Cybern Syst Anal., vol. 42, no. 2, 2006, pp. 254–264. doi: 10.1007/s10559-006-0061-z.

18. Bossche M. V., Ross P., MacLarty I., Nuffelen B. V., Pelov N., "Ontology driven software engineering for real life applications," Proceedings of the 3rd Intl. Workshop on Semantic Web Enabled Software Engineering. 2007. Available at: https://www.semanticscholar.org/paper/Ontology-Driven-SoftwareEngineering-for-Real-Life-BosscheRoss/aabbe8ecd227bd931b44da8cea2aa8d2d1f76519.

19. Ochieng, "PAROT: Translating natural language to SPARQL," Expert Systems with Applications: X, vol. 5, p. 100024, 2020, https://doi.org/10.1016/j.eswax.2020.100024.

20. S. Shaik, P. Kanakam, S. Mahaboob Hussain, D. Suryanarayana,"Transforming natural language query to SPARQL for Semantic Information Retrieval," International Journal of Engineering Trends and Technology - IJETT, vol. 41, no. 7, pp.; 347-350, 2016. https://doi.org/10.14445/22315381/IJETT-V41P263.

21. J. Lehmann and L. Bühmann, "AutoSPARQL: Let users query your knowledge base," The Semantic Web: Research and Applications, G. Antoniou, M. Grobelnik, E. Simperl, B. Parsia, D. Plexousakis, P. De Leenheer, and J. Pan, Eds., in Lecture Notes in Computer Science. Berlin, Heidelberg: Springer, 2011, pp. 63–79. https://doi.org/10.1007/978-3-642-21034-1_5.

22. Litvin A., Velychko V., Kaverinsky V., "A new approach to automatic ontology creation from the untagged text on the natural language of inflective type," Proceedings of the International conference on software engineering "Soft Engine 2022", NAU, Kyiv Ukraine, 2022, pp. 37 – 45.

23. Litvin A., Velychko V., Kaverinsky V., "Development of natural language dialogue software systems," Information Theories and Applications 28. 2021, pp. 233 – 270. doi: 10.54521/ijita28-03-p03.

24. Kaverynskyi V. V., Litvin A. A., Velychko V. Yu., "Tree-based semantic analysis method for natural language phrase to formal query conversion," Radio Electronics, Computer Science, Control, no. 2, 2021, pp. 105 – 113. doi: 10.15588/1607-3274-2021-2-11.

25. WebProtégé [Electronic source] // Biomedical Informatics Research Group – Source access mode: https://webprotege.stanford.edu/.

26. Curé O., Blin G., "RDF database systems: triples storage and SPARQL query processing," First edition. Amsterdam; Boston: Morgan Kaufmann, 2015.

27. LanguageTool API NLP UK [Електронний ресурс] // Corpus of modern Ukrainian language. 2023. https://github.com/brownuk/nlp_uk.

The article has been sent to the editors 25.11.24.
After processing 10.12.24.
Submitted for printing 30.12.24.