ШЕБЫРЕВА А. А.

ПРОЕКТИРОВАНИЕ СИСТЕМЫ АВТОМАТИЗАЦИИ НАПОЛНЕНИЯ ОНТОЛОГИЙ С ИСПОЛЬЗОВАНИЕМ ДАННЫХ ИЗ ОТКРЫТЫХ ИСТОЧНИКОВ

УДК 004.89, ГРНТИ 28.23.01
Статья поступила в редакцию 20.05.2026

Проектирование системы автоматизации наполнения онтологий с использованием данных из открытых источников

Designing an Ontology Filling Automation System Using Open-Source Data

А. А. Шебырева

Национальный исследовательский университет «Высшая школа экономики»; г. Пермь

A. A. Shebyreva

National Research University Higher School of Economics (HSE-Perm), Perm

В статье предложена методика автоматизации построения онтологий на основе интеграции большой языковой модели (LLM) с многоаспектной онтологией. Разработан прототип ИИ-ассистента, выполняющий извлечение схемы онтологии и экземпляров из неструктурированных текстов, автоматическую валидацию и генерацию OWL-файлов. Экспериментальная апробация на примере построения онтологии сердечно-сосудистых заболеваний подтвердила работоспособность методики. Результаты показывают возможность снижения требований к квалификации специалистов.

The article proposes a method for automating the construction of ontologies based on the integration of a large language model (LLM) with a multi-aspect ontology. A prototype of an AI assistant has been developed that extracts ontology schema and instances from unstructured texts, performs automatic validation, and generates OWL files. The experimental validation of the method using the construction of an ontology for cardiovascular diseases has confirmed its effectiveness. The results demonstrate the potential for reducing the requirements for specialist qualifications.

Ключевые слова: *онтология, большие языковые модели, LLM, автоматизация, многоаспектная онтология, извлечение знаний*

Keywords: *ontology, large language models, LLM, automation, multi-aspect ontology, knowledge extraction*

Введение

Онтологии широко применяются в различных предметных областях: в медицине [1, 2], образовании [3], финансовой сфере [4], нефтегазовой отрасли

[5], программной инженерии [6, 7] и т. д. Однако процесс построения и наполнения онтологий остаётся сложным и многоэтапным процессом, требующим высокой квалификации специалистов: они должны быть профессионалами одновременно в предметной области и в области инженерии знаний [8].

Существуют разные подходы к автоматизации отдельных этапов разработки онтологий. Метод MASK (Method of Analysis and Structuring Knowledge) основан на интервьюировании экспертов и компиляции существующих документов, однако не исключает субъективный фактор и требует значительных ручных трудозатрат [9]. Использование UML-диаграмм позволяет визуализировать структуру знаний, но ограничено для представления сложных семантических отношений и требует последующего преобразования в онтологические языки [10, 11]. Метод автоматического построения онтологических моделей с древовидной структурой концептов обеспечивает обработку больших объёмов текстов на основе статистических показателей, однако качество результата сильно зависит от репрезентативности исходного корпуса [12]. Подходы на основе реляционных баз данных, метод METEOR позволяют автоматически преобразовывать структурированные данные в онтологии, но применимы только при наличии готовых баз данных [13, 14].

Особый интерес представляют методы, основанные на использовании генеративного искусственного интеллекта. Плагин для редактора Protégé на основе GPT-3 позволяет преобразовать предложения на естественном языке в аксиомы OWL [15]. Аналогичные подходы демонстрируют возможность использования LLM для извлечения концептов и связей из неструктурированных текстов [8]. Однако применение LLM сопряжено с серьёзными проблемами: логические ошибки, «галлюцинации», неоднозначность интерпретации [8, 16]. Существующие решения автоматизируют только отдельные этапы процесса, не устраняя необходимость участия инженеров по знаниям.

Таким образом, актуальной является разработка методики, позволяющей компенсировать ограничения LLM за счёт формальной логики и правил, обеспечивающей воспроизводимость результатов и снижение порога входа для экспертов предметных областей.

Методика

Предлагается методика, которая основывается на интеграции большой языковой модели с многоаспектной онтологией (базой знаний), которая выступает ядром системы. Многоаспектная онтология включает четыре компонента:

1. Онтология источников данных (Рисунок 1) хранит информацию о типах источников, их метаданных и статусе обработки.

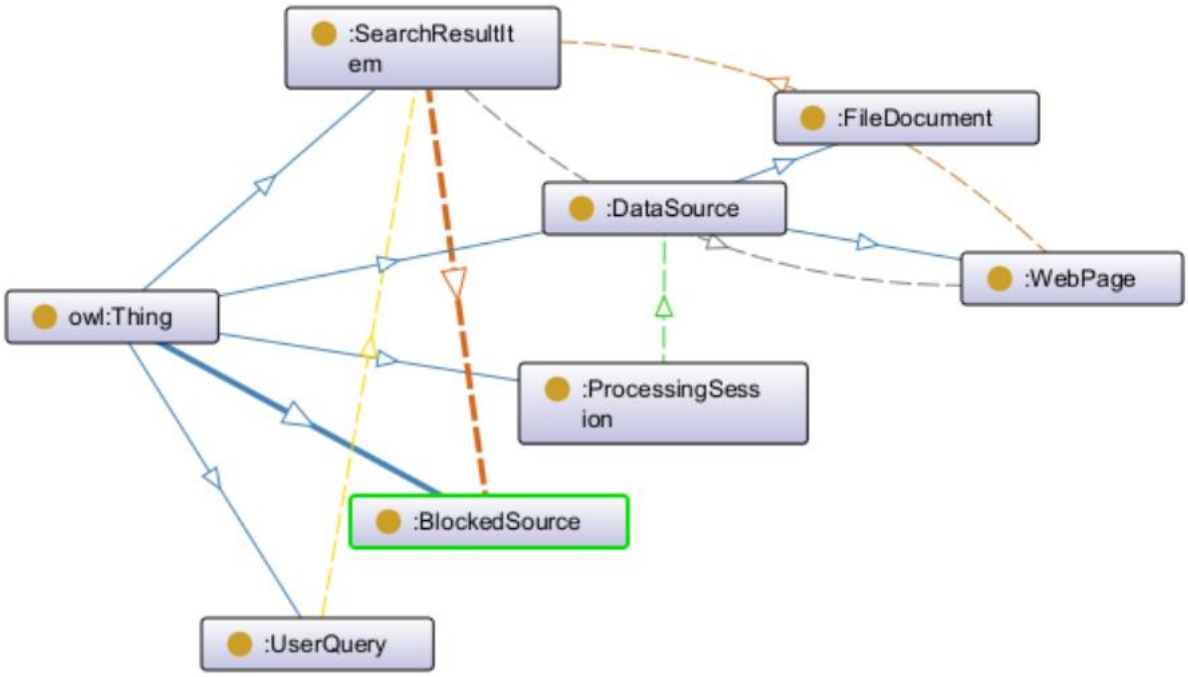


Рисунок 1. Онтология источников данных

2. Онтология промптов (Рисунок 2) хранит шаблоны запросов к языковой модели, их версии и связь с конкретными задачами (извлечение схемы, извлечение экземпляров, валидация, ручная коррекция).

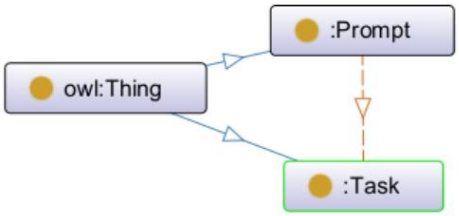


Рисунок 2. Онтология промптов

3. Онтология валидации содержит формализованные правила логической согласованности, полноты и корректности извлечённых элементов онтологии, а также сведения о выявленных ошибках, статусах проверки и связи с сессиями генерации. Онтология предполагает следующие классы: ValidationRun – запись одной сессии проверки (привязана к конкретной онтологии предметной области и моменту времени); Error – обнаруженная ошибка; ErrorType – тип ошибки (дублирование, галлюцинация, несуществующий элемент, зацикленность); CorrectionAction – действие для исправления ошибки (автоматическая или ручная коррекция).

4. Репозиторий пользовательских задач выступает в качестве онтологии для накопления истории работы пользователей: диалогов, запросов, ответов системы. Построенная целевая онтология (онтология предметной области) сохраняется как экземпляр класса TargetOntology. Репозиторий обеспечивает возможность связать целевую онтологию с исходными данными, используемыми промптами, сессией валидации.

Репозиторий включает классы:

- DialogueMessage – каждое отдельное сообщение между пользователем и ИИ-ассистентом,
- GeneratedArtfact – абстрактный класс для всех результатов работы системы (JSON-ответ, схема онтологии, список экземпляров, отчет о валидации, owl-файл),
- TargetOntology – конкретная owl-онтология, сохраняется как экземпляр с мета-данными,
- UsedSource – ссылки на источники данных, которые были использованы для генерации целевой онтологии.

Проектирование архитектуры системы основано на подходе, ориентированном на знания, где ядром платформы выступает многоаспектная онтология, обеспечивающая семантическую интеграцию данных, их согласованность и возможность повторного использования. Архитектура системы построена по трёхуровневому принципу:

1. Уровень представления реализован в виде веб-интерфейса для взаимодействия пользователя с системой.

2. Прикладной уровень выполняет основные процессы обработки данных, включает модули загрузки и предобработки данных, взаимодействия с LLM, извлечения знаний, валидации, оркестрации процессов, генерации онтологии.

3. Уровень знаний представляет собой основу в виде многоаспектной онтологии, которая включает онтологию источников данных, онтологию промптов, онтологию валидации и репозиторий.

Апробация: построение онтологии сердечно-сосудистых заболеваний

Для апробации предложенной методики проведён эксперимент по построению онтологии в предметной области «Сердечно-сосудистые заболевания». В качестве источников данных использовались статьи из Интернет[1][2]. Источники были сохранены в онтологию источников данных с указанием URL, даты загрузки и статуса обработки (Рисунок 3).

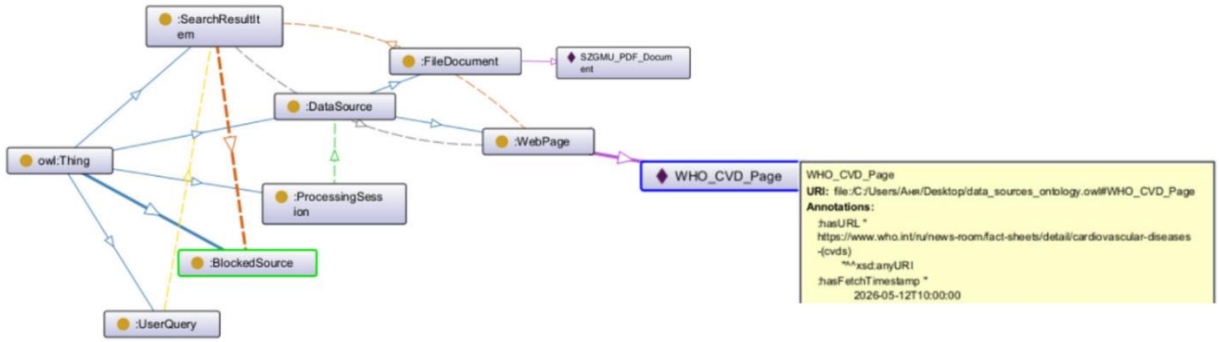


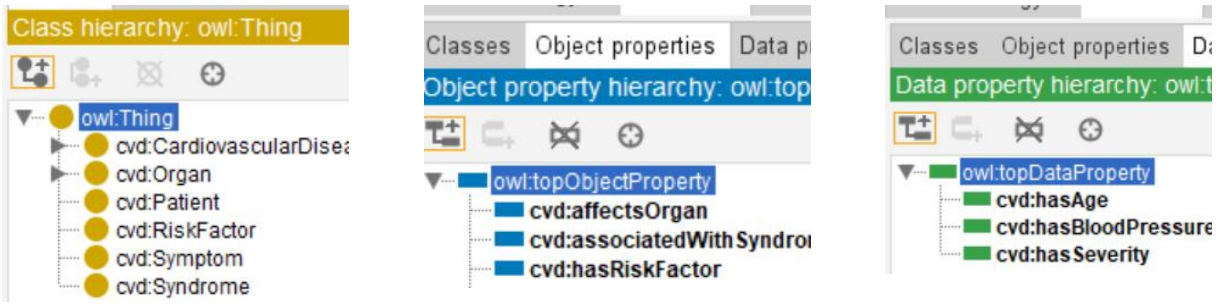
Рисунок 3. Фрагмент онтологии источиников данных

[1] https://www.who.int/ru/news-room/fact-sheets/detail/cardiovascular-diseases-(cvds)
[2] https://szgmu.ru/i/pdo_kaf/115/osnovnie_sindromi_i_simptomi.pdf.

Для извлечения знаний использован шаблон промпта из онтологии промптов с задачей на извлечение. Промпт был ранее сохранен в онтологии промптов, сформулирован следующим образом:

«Ты – инженер по знаниям. Проанализируй источники по теме классификации сердечно-сосудистых заболеваний. Извлеки структуру онтологии: 1. Классы (понятия) и их иерархию. 2. Data properties (атрибуты), которые могут иметь разные значения для разных экземпляров. 3. Экземпляры – конкретные примеры из текста с их значениями свойств. Ответ предоставь строго в формате JSON. Используй информацию только из текста. На основе полученного JSON создай файл в формате OWL.»

В результате работы LLM (модель Llama 3.3 70B через API Groq) была извлечена схема онтологии, включающая: классы (Рисунок 4, а), объектные свойства (Рисунок 4, б) свойства данных (Рисунок 4, в).



а *б* *в*

Рисунок 4. Фрагмент онтологии
а – классы, б – object properties, в – data properties

Фрагмент сгенерированной онтологии в редакторе Protégé показал корректное выделение иерархий и связей.

Далее был выполнен процесс ручной коррекции: был удалён класс Patient – как не относящийся непосредственно к теме с помощью запроса на ручную коррекцию: «Для извлеченной схемы удали класс cvd:Patient и все object properties, data properties, которые с ним связаны. Не меняй другие элементы онтологии». Система успешно обновила структуру классов в репозитории пользовательских задач, а промпт коррекции был сохранён в онтологию промптов для повторного использования.

На заключительном этапе были выполнены запросы на извлечение экземпляров и генерация итоговой OWL-онтологии. Результат был загружен в Protégé для визуализации (Рисунок 5).

Эксперимент подтвердил, что предложенная методика позволяет автоматизировать процесс извлечения знаний из неструктурированных источников и генерации формальной онтологии при минимальном участии эксперта.

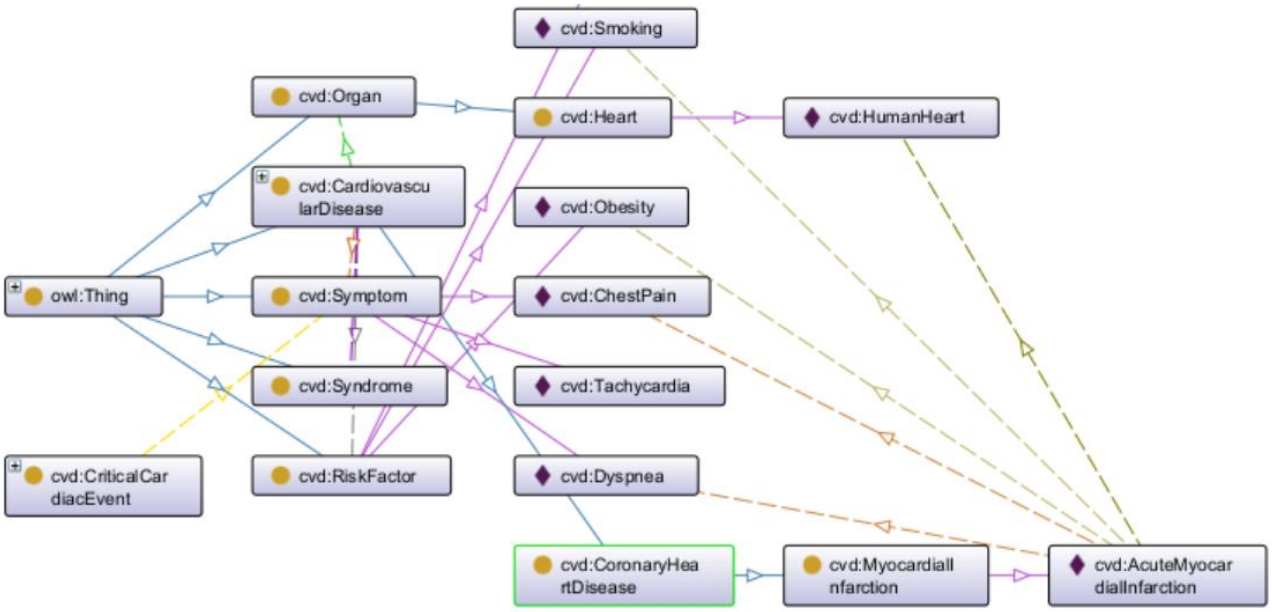


Рисунок 5. Фрагмент целевой онтологии

Заключение

Предлагаемая методика, основанная на интеграции больших языковых моделей и многоаспектной онтологии, позволяет снизить порог вхождения для экспертов предметных областей, не требую от них навыков программирования.

Благодаря использованию онтологии промптов обеспечивается воспроизводимость и повторное использование шаблонов запросов. Онтология источников данных фиксирует происхождение знаний и позволяет автоматически исключать недостоверные источники. Онтология валидации закладывает правила, выявляет ошибки и инициирует автоматическую или ручную коррекцию. Репозиторий пользовательских задач накапливает полную историю сессий, что даёт возможность адаптировать систему к новым предметным областям на основе успешного опыта.

Экспериментальная апробация подтвердила работоспособность методики: система автоматически извлекла схему и экземпляры из неструктурированных текстов, провела валидацию, позволила выполнить целенаправленную ручную коррекцию и сохранила все результаты в репозитории.

Таким образом, предложенная методика может служить основой для создания практических систем. Направления дальнейших исследований включают проведение экспериментов с привлечением экспертов из различных предметных областей для количественной оценки снижения трудоёмкости и качества генерируемых онтологий, доработку механизмов автоматической валидации, а также адаптацию методики для работы с другими открытыми языковыми моделями.

Список использованных источников и литературы

1. Zeshan, F. Medical ontology in the dynamic healthcare environment / F. Zeshan, R. Mohamad // Procedia Computer Science. – 2012. – Vol. 10. – P. 340-348.

2. Hu, J. Development and application of Chinese medical ontology for diabetes mellitus / J. Hu, Z. Huang, X. Ge, Y. Shen, Y. Xu, Z. Zhang // BMC Medical Informatics and Decision Making. – 2024. – Vol. 24, № 1. – P. 18.

3. Zouri, M. An ontology-based approach for curriculum mapping in higher education / M. Zouri, A. Ferworn // Proceedings of 2021 IEEE 11th Annual Computing and Communication Workshop and Conference (CCWC). Las Vegas : IEEE, 2021. – P. 141-147.

4. Amaral, G. Towards an ontology network in finance and economics / G. Amaral, T. P. Sales, G. Guizzardi // Enterprise Engineering Working Conference. Cham : Springer International Publishing, 2021. – P. 42-57.

5. Abolhassani, N. A Data Mesh Adaptable Oil and Gas Ontology Based on Open Subsurface Data Universe (OSDU) / N. Abolhassani, A. Tudor, S. Paul // Proceedings of the 15th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management (IC3K 2023) – Vol 2: KEOD. – P. 29-39. – DOI: 10.5220/0012160000003598.

6. Хорошевский, В. Ф. Проектирование систем программного обеспечения под управлением онтологий: модели, методы, реализации // Онтология проектирования. – 2019. – Т. 9, № 4 (34). – С. 429-448.

7. Разумовский, А. Г. Использование онтологий в разработке программного обеспечения / А. Г. Разумовский, М. Г. Пантелеев // Инженерия знаний и технологии семантического веба. – 2011, № 2. – С. 88-95.

8. Шебырева, А. А. Исследование возможностей автоматизации построения онтологий для информационных систем, управляемых знания. Искусственный интеллект в решении актуальных социальных и экономических проблем XXI века – 2025. – С. 68-73.

9. Matta, N. How to capitalize knowledge with the MASK method ? / N. Matta, J. L. Ermine, G. Aubertin, J.-Y. Trivin // Proceedings IJCAI 2001 Workshop on Knowledge Management and Organizational Memories. – 2001, № 6. – P. 1-13.

10. Kogut, P. UML for ontology development / P. Kogut, S. Cranefield, L. Hart, M. Dutra, K. Baclawski, M. Kokar, J. Smith // The Knowledge Engineering Review. – 2002. – Vol. 17, № 1. – P. 61-64. – DOI: 10.1017/S0269888902000358.

11. Vo, M. Transformation of UML class diagram into OWL Ontology / M. Vo, L. Kh, K. Khoang // Journal of Information and Telecommunication. – 2019. – Vol. 4, №1. – P. 1-16 – DOI: 10.1080/24751839.2019.168668.

12. Чалая, Л. Э., Метод автоматического построения онтологических моделей с древовидной структурой концептов / Л. Э. Чалая, А. В. Чижевский // Автоматизированные системы управления и приборы автоматики. – 2015. – № 173. – С. 32-42.

13. Ben Mahria, B. A novel approach for learning ontology from relational database: from the construction to the evaluation / B. Ben Mahria, I. Chaker, A. Zahi // Journal of Big Data. – 2021. – Vol. 8, № 1. – DOI: 10.1186/s40537-021-00412-2.

14. Лещева, И. А. Метод автоматизированного наполнения баз знаний онтологического типа [Электронный ресурс] : дис. … канд. техн. наук : 2.3.5 / И. А. Лещева; Санкт-Петербургский гос. ун-т. – СПб., 2022. – URL: https://disser.spbu.ru/files/2022/disser_leshcheva.pdf (дата обращения: 01.03.2026).

15. Mateiu, P. Ontology engineering with large language models / P. Mateiu, A. Groza // 2023 25th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing (SYNASC). – IEEE. – 2023. – P. 226-229.

16. Beutel, G. Artificial hallucination: GPT on LSD? / G. Beutel, E. Geerits, J. T. Kielstein // Critical Care. – 2023. – Vol. 27, №. 1.:148. – DOI: 10.1186/s13054-023-04425-6.

List of references

1. Zeshan, F., Mohamad, R. Medical Ontology in the Dynamic Healthcare Environment // *Procedia Computer Science*. – 2012. – Vol. 10. – P. 340-348.

2. Hu, J., Huang, Z., Ge, X., Shen, Y., Xu, Y., Zhang, Z. Development and Application of Chinese Medical Ontology for Diabetes Mellitus // *BMC Medical Informatics and Decision Making*. – 2024. – Vol. 24, No. 1. – P. 18.

3. Zouri, M., Ferworn, A. An Ontology-Based Approach for Curriculum Mapping in Higher Education // *Proceedings of 2021 IEEE 11th Annual Computing and Communication Workshop and Conference (CCWC)*. Las Vegas: IEEE, 2021. – P. 141-147.

4. Amaral, G., Sales, T.P., Guizzardi, G. Towards an Ontology Network in Finance and Economics // *Enterprise Engineering Working Conference*. Cham: Springer International Publishing, 2021. – P. 42-57.

5. Abolhassani, N., Tudor, A., Paul, S. A Data Mesh Adaptable Oil and Gas Ontology Based on Open Subsurface Data Universe (OSDU) // *Proceedings of the 15th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management (IC3K 2023)* – Vol. 2: KEOD. – P. 29-39. DOI: 10.5220/0012160000003598.

6. Khoroshevsky, V.F. Design of Software Systems under Ontology Management: Models, Methods, Implementations // *Ontology of Designing*. – 2019. – Vol. 9, No. 4 (34). – P. 429-448.

7. Razumovsky, A.G., Panteleev, M.G. The Use of Ontologies in Software Development // *Knowledge Engineering and Semantic Web Technologies*. – 2011, No. 2. – P. 88-95.

8. Shebyreva, A.A. Investigation of Opportunities for Automating the Construction of Ontologies for Knowledge-Driven Information Systems // *Artificial Intelligence in Solving Current Social and Economic Problems of the 21st Century – 2025*. – P. 68-73.

9. Matta, N., Ermine, J.L., Aubertin, G., Trivin, J.-Y. How to Capitalize Knowledge with the MASK Method? // *Proceedings IJCAI 2001 Workshop on Knowledge Management and Organizational Memories*. – 2001, No. 6. – P. 1-13.

10. Kogut, P., Cranefield, S., Hart, L., Dutra, M., Baclawski, K., Kokar, M., Smith, J. UML for Ontology Development // *The Knowledge Engineering Review*. – 2002. – Vol. 17, No. 1. – P. 61-64. DOI: 10.1017/S0269888902000358.

11. Vo, M., Kh, L., Khoang, K. Transformation of UML Class Diagram into OWL Ontology // *Journal of Information and Telecommunication*. – 2019. – Vol. 4, No. 1. – P. 1-16. DOI: 10.1080/24751839.2019.168668.

12. Chalaya, L.E., Chizhevsky, A.V. A Method for Automatic Construction of Ontological Models with a Tree Structure of Concepts // *Automated Control Systems and Instrumentation*. – 2015. – No. 173. – P. 32-42.

13. Ben Mahria, B., Chaker, I., Zahi, A. A Novel Approach for Learning Ontology from Relational Database: From the Construction to the Evaluation // *Journal of Big Data*. – 2021. – Vol. 8, No. 1. DOI: 10.1186/s40537-021-00412-2.

14. Leshcheva, I.A. A Method for Automated Filling of Ontological Knowledge Bases [Electronic resource]: Diss. ... Cand. Tech. Sci.: 2.3.5 / I.A. Leshcheva; St. Petersburg State University. – St. Petersburg, 2022. – URL: https://disser.spbu.ru/files/2022/disser_leshcheva.pdf (accessed: 01.03.2026).

15. Mateiu, P., Groza, A. Ontology Engineering with Large Language Models // *2023 25th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing (SYNASC)*. – IEEE. – 2023. – P. 226-229.

16. Beutel, G., Geerits, E., Kielstein, J.T. Artificial Hallucination: GPT on LSD? // *Critical Care*. – 2023. – Vol. 27, No. 1.:148. DOI: 10.1186/s13054-023-04425-6.