From human experts to machines: An LLM supported approach to ontology and knowledge graph construction*

Vamsi Krishna Kommineni[1,2,3][0000−0001−6168−3085], Birgitta König-Ries[1,2,4][0000−0002−2382−9722], and Sheeba Samuel[1,4][0000−0002−7981−8504]

[1] Heinz Nixdorf Chair for Distributed Information Systems, Friedrich Schiller University Jena, Jena, Germany
[2] German Centre for Integrative Biodiversity Research (iDiv) Halle-Jena-Leipzig, Leipzig, Germany
[3] Max Planck Institute for Biogeochemistry, Jena, Germany
[4] Michael Stifel Center Jena, Jena, Germany
(vamsi.krishna.kommineni, birgitta.koenig-ries,
sheeba.samuel)@uni-jena.de

Abstract. The conventional process of building Ontologies and Knowledge Graphs (KGs) heavily relies on human domain experts to define entities and relationship types, establish hierarchies, maintain relevance to the domain, fill the ABox (or populate with instances), and ensure data quality (including amongst others accuracy and completeness). On the other hand, Large Language Models (LLMs) have recently gained popularity for their ability to understand and generate human-like natural language, offering promising ways to automate aspects of this process. This work explores the (semi-)automatic construction of KGs facilitated by open-source LLMs. Our pipeline involves formulating competency questions (CQs), developing an ontology (TBox) based on these CQs, constructing KGs using the developed ontology, and evaluating the resultant KG with minimal to no involvement of human experts. We showcase the feasibility of our semi-automated pipeline by creating a KG on deep learning methodologies by exploiting scholarly publications. To evaluate the answers generated via Retrieval-Augmented-Generation (RAG) as well as the KG concepts automatically extracted using LLMs, we design a judge LLM, which rates the generated content based on ground truth. Our findings suggest that employing LLMs could potentially reduce the human effort involved in the construction of KGs, although a human-in-the-loop approach is recommended to evaluate automatically generated KGs.

Keywords: Knowledge Graphs · Ontology · Competency Questions · Large Language Models · Retrieval-augmented generation.

arXiv:2403.08345v1 [cs.CL] 13 Mar 2024

1   Introduction

In information organization and representation, ontologies stand as foundational frameworks for describing and structuring domain knowledge. These structured representations not only represent the entities and relationships within a domain but also serve as the foundation for constructing comprehensive knowledge graphs (KGs) [10]. KGs, in turn, offer a powerful mechanism for interlinking diverse pieces of information and facilitating sophisticated data analytics and reasoning. The domain knowledge encapsulated within ontologies constitutes a valuable asset for various knowledge-intensive applications. This comes at a price, though: Ontology and knowledge engineering represents a collaborative and interdisciplinary effort, demanding the time and expertise of multiple stakeholders [8,19]. The higher the expressiveness in the ontology language, the more intricate design choices need to be made throughout the construction process with additional developmental challenges including accuracy, scalability, and depth of knowledge captured. The conventional approach to constructing KGs typically involves gathering domain requirements through CQs (also proposed as the requirement specification in ontology development) collected from domain experts, collaborating with computer scientists and domain specialists to develop an ontology, transforming unstructured data into structured formats, and finally, populating the ontology to create the KG [9]. In this context, a pressing question emerges: How can we strike a balance between the resource-intensive nature of ontology construction and the imperative of leveraging domain knowledge effectively?

This research paper attempts to address this question by leveraging Large Language Models (LLMs) in Knowledge Graph Engineering, particularly focusing on minimizing the time and human effort involved in these processes. In recent years, the emergence of these models has revolutionized the landscape of natural language processing (NLP) and knowledge representation. Equipped with massive pre-trained parameters and advanced neural architectures, LLMs exhibit remarkable capabilities in understanding and generating human-like text across a spectrum of languages and domains [4,6]. As a result, they have gained immense popularity and adoption across diverse fields ranging from information retrieval to language translation.

Hence, we explore the (semi-)automatic construction of a KG, starting from collecting competency questions (CQs) to creating an ontology and to filling the data in the ontology, facilitated by the integration of LLMs. To test the feasibility of our approach, we apply this methodology to creating an ontology and KGs about deep learning (DL) methodologies extracted from scholarly publications in the biodiversity domain.

The choice of this example has been motivated by the increasing usage of DL in research. Documenting the provenance of DL results is indispensable to facilitate the reproducibility of these studies, a prerequisite to trust and validation of results. To accomplish this task effectively, information, such as models, architectures, hyperparameters, and other key details needs to be captured and stored in a structured representation.

2   Related work

LLMs have revolutionized knowledge engineering and NLP, showcasing human-level performance across diverse linguistic tasks [4,6]. With the increasing robustness of LLMs, their potential as a knowledge source in various applications such as KG completion, ontology refinement, and question answering has become evident [14,20,24]. Research has rapidly expanded to explore the application of LLMs, with recent papers providing surveys on the use of LLMs in KG engineering along with associated challenges [17,19]. Meyer et al. [17] presents a list of application areas for LLM-assisted KG engineering, including creating or enriching KG schemas/ontologies, populating KGs, etc. In their position paper, Pan et al. [19] present opportunities, visions, research topics, and challenges for LLMs for KGs and KGs for LLMs.

Recent studies have introduced methods for ontology creation [7,8], augmentation [26], completion [22], and learning [3] using LLMs. Cohen et al. [7] present a crawling approach to extract a KG from an LLM using 'subject-relation-object' statement formats. Funk et al. [8] focus on constructing a concept hierarchy for a given domain starting from a seed concept by querying LLMs. However, they consider only subconcept/is-a relation, but no other relations. In this work, we focus on reusing existing ontologies with all their concepts and relations. While there is limited research on the utilization of LLMs for completing KGs or ontologies [23], this trend appears to be changing rapidly. While LLMs offer new possibilities for ontology learning and development, they do not alter the fundamental need for expert collaboration and establishing consensus within a community. However, they may enhance the productivity of ontologists by simplifying ontology development and integrating into semi-automatic toolchains [18]. In our approach, we have reused the existing ontology in the domain and involved humans in the loop to validate and evaluate the LLM-generated content.

Most of the publications discussed above [8,17,18,22,26] are based on openAI's GPT 3.5[5] or 4[6]. However, the usage cost of OpenAI API models can escalate quickly if deployed for large-scale applications. In contrast, open-source LLMs offer transparency, model control, usage flexibility, and cost-effectiveness. As far as our knowledge extends, this represents the first approach to introduce a comprehensive (semi-)automated pipeline for constructing ontology and KGs by using open-source LLMs.

3   Methods

This section presents our (semi-)automatic pipeline for constructing KGs with six main components: Data Collection, CQ generation, Ontology creation, CQ answering, KG construction, and Evaluation (Figure 1). The prompts, code, results, and data used for the evaluation are publicly available at GitHub: `https://github.com/fusion-jena/automatic-KG-creation-with-LLM`. The

---

[5] `https://platform.openai.com/docs/models/gpt-3-5`
[6] `https://platform.openai.com/docs/models/gpt-4`

computational infrastructure to create and execute the pipeline is provided by Friedrich Schiller University Jena: Draco cluster, computer node with GPU accelerator, 1x NVIDIA A100, 80GB.
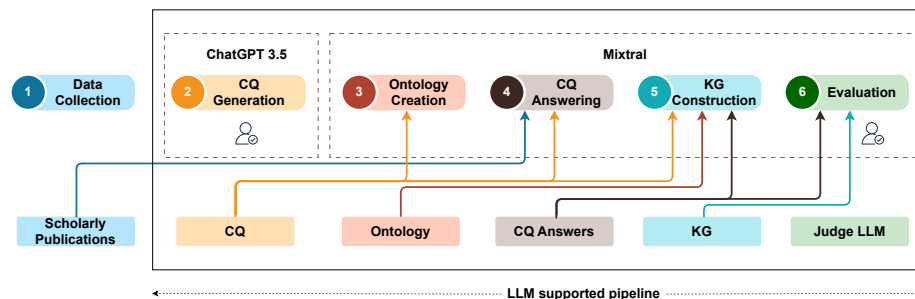
Fig. 1. The (semi-)automatic approach for constructing KGs. Human-in-the-loop is involved in the second and sixth stages of the LLM supported pipeline.

Data Collection: We reused a dataset generated in our prior research [2], where we conducted a systematic literature review to identify publications employing DL methods in biodiversity research based on keywords suggested by biodiversity experts [1]. Additionally, two domain experts curated a dataset of 61 publications and manually extracted reproducibility-related variables inspired by the current literature[7]. We used the first five of these 61 publications in this work to test our (semi-) automated pipeline.

CQ generation: To generate the CQs, we prompted ChatGPT-3.5 to get abstract-level questions to describe the provenance of the results of DL pipelines. Two human domain experts evaluated the CQs generated by the ChatGPT-3.5 web interface to enhance existing questions and add new ones.

After the CQ generation step, we conducted experiments using three open-source models: Llama 2-70B[8], 'Mixtral 8x7B'[9], and Falcon-40B[10]. Upon manual comparison of the content quality generated by these models, it became apparent that the outputs of Mixtral 8x7B were superior comparatively and provided the expected results. Hence, starting from the ontology creation component, we employed 'Mixtral 8x7B', a sparse mixture-of-experts network, which was inferred at a zero-shot setting. The hyperparameters of zero-shot Mixtral 8x7B include a temperature of (10^{-5}) and a maximum output token limit of 25,000 tokens. For the RAG pipeline, the chunk size for text splitting is set at 2,500 tokens, with a chunk overlap of 100 tokens.

---

[7] The dataset with reproducibility-related variables is available at `https://github.com/fusion-jena/Reproduce-DLmethods-Biodiv`

[8] `https://huggingface.co/meta-llama/Llama-2-70b-chat-hf`

[9] `https://mistral.ai/news/mixtral-of-experts/`

[10] `https://huggingface.co/tiiuae/falcon-40b`

Ontology creation: A two-step strategy was implemented to create the ontology from the LLM-generated human-verified CQs. In the first step, we aimed to extract all concepts and their relationships from the CQs. To achieve this, we included a set of instructions and an example CQ with expected output in the prompt. In the second step, we constructed an ontology for describing information on DL pipelines. This was achieved by providing an in-context example containing a basic ontology structure, utilizing the PROV-O ontology [13] as a foundational ontology for reuse, and incorporating the concepts and relationships extracted from the CQs.

CQ Answering: This component is the central pillar of the whole pipeline, particularly for KG construction. With this component, we retrieved answers for all the CQs using the RAG approach from the first five selected biodiversity scholarly publications from our dataset that employed DL methods. We then applied basic text processing techniques to refine the generated answers, eliminating redundant and repetitive content where applicable.

KG construction: To construct the KG, processed CQ answers along with their respective questions and the LLM-generated ontology were given as input to the LLM. With the prompt, we instructed the LLM to extract key entities, relationships, and concepts from the answers and map them onto the ontology to generate the KG.

Evaluation: Two key outputs produced by the LLM were evaluated in this step: the generated CQ answers and the KG concepts that were automatically extracted from these answers. To evaluate these outputs, we employed the LLM as a judge (instructed using a prompt) to assign scores to the generated content based on human evaluator-generated ground truth. Due to the time and human effort required for manual annotation, a human expert annotated only five scholarly articles with ground truth text for the CQs and assigned single words (Right, Wrong, and Partial) to evaluate the RAG-generated CQ answers. Consequently, these five publications are used in our complete pipeline to test the feasibility of our approach.

To assess the generated CQ answers, we asked the LLM to score the alignment between the ground truth and the generated answer on a scale from 0 to 10, where 0 represents no alignment and 10 represents maximum alignment. Then we classified scores greater than or equal to six as Right, scores less than three as Wrong, and the remaining scores as Partial. The automatically extracted KG concepts were also evaluated with the judge LLM, which checked whether the KG individual concepts appeared in the respective generated CQ answers. We used the ontology concepts to establish links between the KG individual concept and the corresponding generated CQ answers.

4 Results

The CQ generation step of the pipeline resulted in 40 CQs. These CQs investigate every step of the DL pipeline, including raw data sources, preprocessing techniques, model architectures, hyperparameter settings, software and hardware

choices, post-processing steps, security measures in handling sensitive data, and data biases alongside ethical considerations.
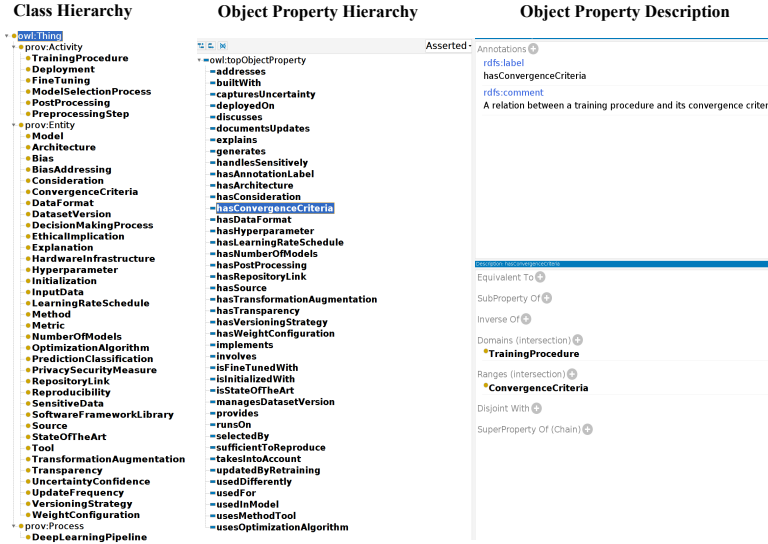

Fig. 2. An excerpt of the DLProv ontology generated by the LLM in our pipeline.

Based on these CQs, the Ontology generation step of the pipeline developed the DLProv Ontology reusing the PROV-O ontology [13]. It consists of 45 classes and 41 relationships with 365 axioms. See Figure 2 for an excerpt. However, concepts were added as subclasses to three PROV-O classes only, and no object properties were reused. In another experiment where we provided the MLSchema ontology[11] for reuse, only one class was reused and no object properties.

The CQ answering step of the pipeline generated answers for each CQ for the first five publications from our dataset. There were 42 disagreements between the human annotator and the LLM Judge out of 200 evaluated CQ answers during the evaluation of these RAG-generated answers.

Using the CQs, the answers from the CQs generated from the scholarly publications and the ontology, a KG was constructed from the method information of DL pipelines extracted from five scholarly publications using our pipeline. Listing 1.1 shows the excerpt of the KG describing the provenance of the results from the DL methods from the publication [12].

Listing 1.1. An excerpt of the KG generated by our (semi-) automated pipeline

```
dlprov:DeepLearningPipeline_1  rdf:type  dlprov:DeepLearningPipeline ;
```

---

[11] http://www.w3.org/2016/10/mls/

```
    dlprov:hasDataFormat dlprov:DataFormat_1 ;
    dlprov:hasDataFormat dlprov:DataFormat_2 ;
dlprov:DataFormat_1 rdf:type dlprov:DataFormat ;
    rdfs:label 'Audio Spectrogram'.

dlprov:DataFormat_2 rdf:type dlprov:DataFormat ;
    rdfs:label 'Image data'.
```

We have created multiple KGs in combination with two different prompts and two different RAG-generated CQ answers for five selected publications (Table 1). Our best KGs have successfully correlated 142 KG individuals with the CQ answers out of a total of 203 KG individuals across five KGs, each corresponding to a selected scholarly publication.

Publication / Combination of prompt and CQ answers	Prompt v1 and CQ answer v1	Prompt v1 and CQ answer v2	Prompt v2 and CQ answer v1	Prompt v2 and CQ answer v2
Klein et al. 2015 [12]	24.32	x	x	9.42
Khalighifar et al. 2021 [11]	85.71	x	x	68.06
Choe et al. 2021 [5]	73.21	64.41	59.26	64.29
Mahmood et al. 2016 [16]	91.53	81.48	82.76	77.55
Younis et al. 2020 [25]	66.67	x	67.51	61.67

Table 1. Percentage of KG individuals that are in line with respective CQ answers for different combinations of prompt and CQ answers. (x is the instance with no meaningful KG for that particular combination)

5   Discussion

Despite well-known problems such as hallucination, lack of critical thinking, outdated retrieval, and prompt sensitiveness [15,19,21], LLMs are increasingly being used for information retrieval. Initially, we tried to create a KG representing the method information of the DL pipeline from scholarly publications in a single step by providing the whole publication content by prompting the LLM. However, this approach provided little to no content related to KG. Then, we divided the whole process into six components, as discussed in Section 3.

The LLM-generated output produced in our pipeline contained excessive unnecessary explanation, which was refined using automatic text processing techniques, as demonstrated in the CQ Answering step, where redundant and repetitive segments of the generated content were eliminated. This helps to minimize the disagreements to 42 out of 200 between the human evaluator and LLM Judge. However, in most instances of disagreement, Judge LLM deemed the human-annotated 'partial' labelled CQ answers 'wrong'. Furthermore, there are instances where the generated KG format needs adjustments to ensure compatibility with libraries or SPARQL engines for querying purposes.

Although our semi-automated pipeline demonstrates the potential for automatic KG generation from provided content, the evaluation of the generated KG reveals a few limitations: (1) The consistency of the generated KG varies; the LLM assigns the rdfs:label for some KG individuals as 'Not Specified' if it cannot find related information in the provided content. In other instances, it simply omits KG individuals with no associated information. (2) Multiple KG individuals are created for the same ontology class by repeating the same value for the class due to LLM's struggle to capture the intact information from the provided content. In those cases, LLM fails to create meaningful KGs.

Prompt engineering has been pivotal at every stage of our pipeline, particularly in generating RAG CQ answers and KG generation. During KG construction, the outcome varied even with the change in the prompt input order of CQ answers and ontology. Throughout the entire pipeline, minor alterations in the prompt resulted in the LLM generating different answers most of the time, reflecting a well-known LLM issue that also persists in our pipeline. To address this challenge, we incorporated in-context examples wherever required, which helped to reduce the variability in the generated and hallucinated content. Initially, we describe the task briefly for writing an initial prompt. After reviewing the initial results, we added more details to the prompt via trial and error method to reach the optimal prompt, which in turn provided optimal expected answers.

We investigated different tools like Ragas[12] and Tonic ai[13] to evaluate the RAG-generated content; however, they function fully only when an OpenAI API key is provided. Therefore, we devised a judge LLM with open-source LLMs that can rate the generated content using ground truth values.

6   Conclusion

In this study, we have explored the use of open-source LLMs for the creation of ontologies and knowledge graphs. Our proposed pipeline follows established ontology engineering practices and shows the potential of LLMs acting as assistants (or co-pilots) to human experts. With this, ontology and KG creation require significantly lower effort and less semantic web expertise, making these powerful tools available for broader use. In our future work, on the one hand, we aim to take advantage of this in our own ontology and KG development work. On the other hand, we will continue improving the pipeline. To do so, we plan to run our pipeline on different hardware and evaluate the results using different open-source LLMs to discern potential variations in results. The prompt sensitivity of the LLM is high implying that even minor alterations in prompt design could significantly impact LLM outputs. Similarly, we anticipate varying results based on hardware usage due to differences in variable initializations across distinct hardware configurations, which can also impact LLM outputs. Furthermore, we will explore methods for mapping the generated ontology with other ML/DL ontologies.

---

[12] `https://docs.ragas.io/en/latest/index.html`
[13] `https://www.tonic.ai/validate`

References

1. Abdelmageed, N., Löffler, F., Feddoul, L., Algergawy, A., Samuel, S., Gaikwad, J., Kazem, A., König-Ries, B.: BiodivNERE: Gold standard corpora for named entity recognition and relation extraction in the biodiversity domain. Biodiversity Data Journal **10** (2022)
2. Ahmed, W., Kommineni, V.K., König-ries, B., Samuel, S.: How reproducible are the results gained with the help of deep learning methods in biodiversity research? Biodiversity Information Science and Standards **7** (2023)
3. Babaei Giglou, H., D'Souza, J., Auer, S.: LLMs4OL: Large language models for ontology learning. In: International Semantic Web Conference. pp. 408–427. Springer (2023)
4. Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J.D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al.: Language models are few-shot learners. Advances in neural information processing systems **33**, 1877–1901 (2020)
5. Choe, H., Chi, J., Thorne, J.H.: Mapping potential plant species richness over large areas with deep learning, MODIS, and species distribution models. Remote Sens. (Basel) **13**(13), 2490 (Jun 2021)
6. Chowdhery, A., Narang, S., Devlin, J., Bosma, M., Mishra, G., Roberts, A., Barham, P., Chung, H.W., Sutton, C., Gehrmann, S., et al.: Palm: Scaling language modeling with pathways. Journal of Machine Learning Research **24**(240), 1–113 (2023)
7. Cohen, R., Geva, M., Berant, J., Globerson, A.: Crawling the internal knowledge-base of language models. In: Findings of the Association for Computational Linguistics: EACL 2023, Dubrovnik, Croatia, May 2-6, 2023. pp. 1811–1824. Association for Computational Linguistics (2023)
8. Funk, M., Hosemann, S., Jung, J.C., Lutz, C.: Towards ontology construction with language models. In: Joint proceedings of the 1st workshop on Knowledge Base Construction from Pre-Trained Language Models (KBC-LM) and the 2nd challenge on Language Models for Knowledge Base Construction (LM-KBC) co-located with the 22nd International Semantic Web Conference (ISWC 2023), Athens, Greece, November 6, 2023. CEUR Workshop Proceedings, vol. 3577. CEUR-WS.org (2023)
9. Haussmann, S., Seneviratne, O., Chen, Y., Ne'eman, Y., Codella, J., Chen, C.H., McGuinness, D.L., Zaki, M.J.: Foodkg: a semantics-driven knowledge graph for food recommendation. In: The Semantic Web–ISWC 2019: 18th International Semantic Web Conference, Auckland, New Zealand, October 26–30, 2019, Proceedings, Part II 18. pp. 146–162. Springer (2019)
10. Hogan, A., Blomqvist, E., Cochez, M., d'Amato, C., Melo, G.D., Gutierrez, C., Kirrane, S., Gayo, J.E.L., Navigli, R., Neumaier, S., et al.: Knowledge graphs. ACM Computing Surveys (Csur) **54**(4), 1–37 (2021)
11. Khalighifar, A., Brown, R.M., Goyes Vallejos, J., Peterson, A.T.: Deep learning improves acoustic biodiversity monitoring and new candidate forest frog species identification (genus platymantis) in the Philippines. Biodivers. Conserv. **30**(3), 643–657 (Mar 2021)
12. Klein, D.J., McKown, M.W., Tershy, B.R.: Deep learning for large scale biodiversity monitoring. In: Bloomberg Data for Good Exchange Conference (2015)
13. Lebo, T., Sahoo, S., McGuinness, D., Belhajjame, K., Cheney, J., Corsar, D., Garijo, D., Soiland-Reyes, S., Zednik, S., Zhao, J.: PROV-O: The PROV ontology. W3C recommendation **30** (2013)

14. Liu, H., Perl, Y., Geller, J.: Concept placement using BERT trained by transforming and summarizing biomedical ontology structure. Journal of Biomedical Informatics **112**, 103607 (2020)
15. Lu, Y., Bartolo, M., Moore, A., Riedel, S., Stenetorp, P.: Fantastically ordered prompts and where to find them: Overcoming few-shot prompt order sensitivity. In: Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). pp. 8086–8098. Association for Computational Linguistics, Dublin, Ireland (May 2022)
16. Mahmood, A., Bennamoun, M., An, S., Sohel, F., Boussaid, F., Hovey, R., Kendrick, G., Fisher, R.B.: Automatic annotation of coral reefs using deep learning. In: OCEANS 2016 MTS/IEEE Monterey. IEEE (Sep 2016)
17. Meyer, L.P., Stadler, C., Frey, J., Radtke, N., Junghanns, K., Meissner, R., Dziwis, G., Bulert, K., Martin, M.: LLM-assisted knowledge graph engineering: Experiments with chatgpt. arXiv preprint arXiv:2307.06917 (2023)
18. Neuhaus, F.: Ontologies in the era of large language models - a perspective. Appl. Ontology **18**(4), 399–407 (2023)
19. Pan, J.Z., Razniewski, S., Kalo, J.C., Singhania, S., Chen, J., Dietze, S., Jabeen, H., Omeliyanenko, J., Zhang, W., Lissandrini, M., Biswas, R., de Melo, G., Bonifati, A., Vakaj, E., Dragoni, M., Graux, D.: Large Language Models and Knowledge Graphs: Opportunities and Challenges. Transactions on Graph Data and Knowledge **1**(1), 2:1–2:38 (2023)
20. Petroni, F., Rocktäschel, T., Riedel, S., Lewis, P., Bakhtin, A., Wu, Y., Miller, A.: Language models as knowledge bases? In: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP). pp. 2463–2473. Association for Computational Linguistics, Hong Kong, China (Nov 2019)
21. Razniewski, S., Yates, A., Kassner, N., Weikum, G.: Language models as or for knowledge bases. arXiv preprint arXiv:2110.04888 (2021)
22. Toro, S., Anagnostopoulos, A.V., Bello, S., Blumberg, K., Cameron, R., Carmody, L., Diehl, A.D., Dooley, D., Duncan, W., Fey, P., et al.: Dynamic retrieval augmented generation of ontologies using artificial intelligence (dragon-ai). arXiv preprint arXiv:2312.10904 (2023)
23. Veseli, B., Singhania, S., Razniewski, S., Weikum, G.: Evaluating language models for knowledge base completion. In: The Semantic Web - 20th International Conference, ESWC 2023, Hersonissos, Crete, Greece, May 28 - June 1, 2023, Proceedings. Lecture Notes in Computer Science, vol. 13870, pp. 227–243. Springer (2023)
24. Yao, L., Mao, C., Luo, Y.: KG-BERT: BERT for knowledge graph completion. arXiv preprint arXiv:1909.03193 (2019)
25. Younis, S., Schmidt, M., Weiland, C., Dressler, S., Seeger, B., Hickler, T.: Detection and annotation of plant organs from digitised herbarium scans using deep learning. Biodivers. Data J. **8**, e57090 (Dec 2020)
26. Zaitoun, A., Sagi, T., Wilk, S., Peleg, M.: Can large language models augment a biomedical ontology with missing concepts and relations? arXiv preprint arXiv:2311.06858 (2023)