Anchored Semantics: Augmenting Ontologies via Competency Questions, Self-Attention, and Predictive Graph Learning

Shengqi Li and Amarnath Gupta[000−0003−0897−120X]

University of California San Diego, La Jolla, CA 92093, USA
{shl142,a1gupta}@ucsd.edu

Abstract. We propose a framework that enriches ontologies by leveraging competency questions and distant supervision. The process begins by using an LLM to extract domain-relevant entities from the questions, followed by incremental refinement through short definitions anchored to a predefined dictionary. These entities and their hierarchies, along with associated queries, are embedded using a fine-tuned Llama3.2:1b and further processed through a self-attention mechanism to create unified representations. A directed acyclic graph models the dependencies between entities, with additional nodes derived from frequent co-occurrences in queries. A Graph Attention Network (GAT) is used for stable link prediction, discovering latent semantic relationships. These links are then labeled with specific relation types using a fine-tuned RoBERTa module. Evaluations using datasets from HPC training sessions and OpenAlex abstracts show significant improvements in link prediction and ontology enrichment over standard GAT and GraphSage baselines.

Keywords: Ontology Augmentation · Competency Questions · Distant Supervision · Graph Embedding.

1 Introduction

Ontologies serve as crucial formal knowledge representations that bridge the gap between human conceptual understanding and machine-processable data structures in modern AI systems. They provide a mathematically rigorous framework for encoding domain expertise, ensuring semantic interoperability across diverse, heterogeneous data by bridging them to a network of related concepts, and answer complex queries that require connecting information across multiple domains. Since they represent domain knowledge, ontologies are often used as a part of a knowledge graph and enables better explainability in AI-based tasks like answering natural language questions.

Many of the early ontologies (e.g., biomedical ontologies such as the Gene Ontology) were constructed by domain experts via human processes, and were regularly maintained and updated as new terminology and new uses emerged[10, 16], but such top-down development is slow, non-scalable, and insufficiently agile. Data-driven methods that generate ontologies from text accelerate development

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97635-3_23

but often sacrifice quality and ignore downstream user needs; we therefore require a bottom-up, logically consistent, vocabulary-aligned methodology that can evolve with user demands. Recent advances in LLMs present new opportunities: several studies generate or augment ontologies directly from competency questions—manually or via rule-based techniques[6]—offering deterministic consistency yet limited adaptability, while transformer-based approaches cannot fully capture CQ variability[4]; other work fine-tunes GPT-3 to translate natural language into OWL Functional Syntax[11] or leverages zero- and few-shot learning for ontology alignment[2, 8]. Although these enhance expressiveness and matching, our approach differs by integrating extracted CQ topics with distant supervision and deep-learning-based structural prediction to deliver a more automated, scalable ontology augmentation.

We represent user demands via a set of competency questions (CQs), where CQs are natural-language questions that the completed ontology should answer, which have been shown to help resolve ontology defects by introducing entities and relationships the ontology does not capture[3]. We adopt a setting where a consistent but incomplete, task-agnostic ontology exists for some domain, and a CQ bank, obtained from prospective users, is used to computationally extend it while maintaining our design criteria.

In this paper, we investigate how generative models—specifically LLMs, which hold great promise in transfer learning[1] can be effectively used in bottom-up ontology construction using CQs as a guideline (cf.[13]). We propose a framework that leverages an LLM to extract key domain-relevant entities and their relationships from CQs to enhance an existing ontology, systematically addressing both content and structural heterogeneity. Finally, a GAT integrates the explicit links provided by CQs and infers latent semantic relationships between entities, thereby augmenting the expressiveness of the ontology. Specifically, this paper makes the following contributions to CQ-driven ontology augmentation.

- We formalize the problem of ontology augmentation by incorporating competency questions, addressing both content and structural heterogeneity.
- We propose an innovative framework that utilizes LLMs for entity extraction and recursive definition generation, coupled with a multi-head self-attention mechanism to fuse multiple feature modalities.
- We design a comprehensive graph-based model that integrates the question-entity, entity-category, and inferred entity-entity relationships, thus enriching the ontology's expressiveness.
- We empirically demonstrate the effectiveness of our approach on real-world datasets, showing notable improvements in ontology coverage and link prediction accuracy.

2 Our Approach

Our automated ontology construction process has two inputs — an existing ontology O that needs to be augmented and a set of competency questions Q obtained from users. We perform a semantic analysis of each question $q \in Q$,

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97635-3_23

together with $V(O)$, the verbalized version of the ontology. The final ontology is generated in a 3-stage process. In the *Entity Extraction and Definition* stage (Section2.1), semantic entities are extracted from $Q \cup V(O)$ and composed into an initial `subClassOf`DAG. Next, in the *Embedding Generation and Fusion* stage (Section2.2), each entity's label, hierarchy path, and definition embeddings are combined via self-attention into a holistic representation. Finally, in the *Graph-based Link Prediction and Labeling* stage (Section2.3), a Graph Attention Network identifies new relations and a classifier assigns each predicted edge its relation type.

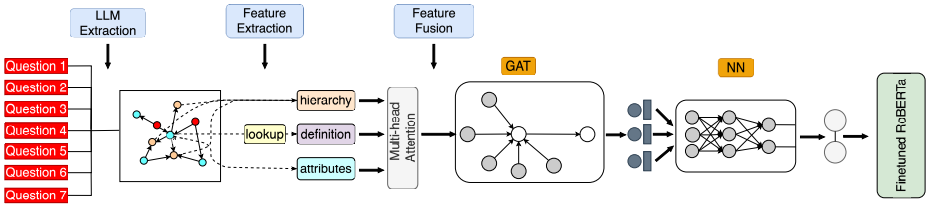

Fig. 1. The primary architecture of the system

2.1 Entity Extraction and Definition

Our process begins with two inputs—an existing ontology O and a set of competency questions Q. We first verbalize O into natural language using an ontology verbalizer[15] and concatenate that text with each $q \in Q$. An LLM, prompted via few-shot examples, parses this combined text to extract candidate entities (noun phrases) along with their hierarchical category paths up to the root node `Entity`. For each candidate:

- We consult a distant-supervision dictionary of known definitions. If found, the entity is marked "grounded."
- Otherwise, the LLM is prompted to generate a concise, one-sentence definition (cf.[13]). Any new terms in that definition are recursively extracted and resolved until all entities map to dictionary entries.
- As post-processing, we remove cycles, duplicate definitions, and acronyms to ensure the result is a clean `subClassOf` DAG where vertices are entities and edges capture dependency relationships from the definition process.

2.2 Embedding Generation and Fusion

Each entity e is represented by three facets—its label n, its hierarchy path $\mathcal{C}(e)$, and its definition $d(e)$. We fine-tune a Llama3.2-1b model via LoRA [9] to encode each facet into vectors $\mathbf{v}_n$, $\mathbf{v}_C$, and $\mathbf{v}_d$, drawing training pairs from co-occurrence of entities in OpenAlex abstracts, with positive pairs share a CQ;

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97635-3_23

negative pairs have their nearest common ancestor at least three levels above. Rather than using these vectors independently, we stack them into a matrix and apply a single self-attention layer, whose parameters are trained using the same distant-supervision data mentioned in the earlier encoding stages.

Although there are well-known graph-only embedding approaches, including RDF2Vec[14], OWL2Vec*[5], TransE/DistMult variants. They rely exclusively on triple walks or translation objectives and ignore the rich textual and hierarchical signals in our augmented ontology. In contrast, we obtain embeddings that capture intrinsic semantics, taxonomic context, and relational differences by fine-tuning Llama3.2:1b on our domain corpus and competency questions.

2.3 Graph-based Link Prediction and Labeling

We construct an augmented graph $G' = (V, E \cup E_r)$ by adding co-occurrence edges

$$E_r = \{(e_i, e_j) \mid \mathcal{Q}(e_i) \cap \mathcal{Q}(e_j) \neq \emptyset\}.$$

A Graph Attention Network (GAT) then propagates each $\mathbf{v}_e$ over its neighborhood:

$$\alpha_{ij} = \mathrm{softmax}_j\big(\mathrm{LeakyReLU}(a^\top [W\mathbf{v}_{e_i} \| W\mathbf{v}_{e_j}])\big), \quad \mathbf{v}'_{e_i} = \sigma\Big(\sum_{e_j \in \mathcal{N}(e_i)} \alpha_{ij}\, W\mathbf{v}_{e_j}\Big).$$

An MLP over $[\mathbf{v}'_{e_i} \| \mathbf{v}'_{e_j}]$ then scores link existence. For each predicted link, we form an input by concatenating the text of e_i, e_j, and their shared question q_{ij}, and classify it into one of 21 relation types, combining existing HPC labels with a few new predicates via a lightweight fine-tuned classifier.

3 Experiments

3.1 Experiment Design

Datasets Our primary dataset for ontology augmentation consists of 1,127 competency questions collected from SDSC HPC training sessions. These questions are structured according to the HPC-fair ontology to model relationships within the HPC domain. After extracting the data using GPT-4o, we applied rigorous post-processing steps refining the dataset into a well-structured graph representation.

To fine-tune the LLM for embedding generation, we supplemented our dataset with additional data from a publication resource, OpenAlex. The data collection process began by extracting pairs of entities from the graph constructed using competency questions. We classified entity pairs as relevant if they originated from the same competency question and non-relevant if they shared a common ancestor at a hierarchical distance of three steps.

For each entity pair, we searched for paper abstracts where both keywords co-occurred. In practice, we randomly selected 200 entities from the graph, collecting k_1 (e.g., 3) relevant and an equal number of non-relevant associated words. For each pair of words, we retrieved n (10 in our experiments) abstracts of articles from OpenAlex.

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97635-3_23

Table 1. Dataset Statistics

Competency Questions Dataset	
Questions	1,127
Entities	9,590
Categories	4,954
Question–Entity edges	28,521
Entity–Category edges	17,492
Category–Category edges	9,133
OpenAlex Paper Abstracts	
Word pairs	1,200
Paper abstracts	12,000

3.2 Link Prediction

Which pair of entities is related? To evaluate the effectiveness of our ontology in capturing semantic relationships between entities, we compare our results with pure GAT and GraphSage [7] without self-attention embedding fusion.

The GAT we are using here is trained on the same data as the distant supervision employed in the previous steps. The training labels are derived from the following logic:

- Two entities are *related* if they share at least one competency question,
- Two entities are *not-related* if their closest common ancestor is at least three steps above either node in the hierarchy.

Instead of random sampling part of the graph as training set and the rest as validation and testing set, we consider the case of incremental ontology augmentation. The original graph is used as the training set, while we split a subset of competency questions and corresponding entities as the testing set. Positive samples are selected using the same logic as training labels, and an equal number of negative samples are selected using only the new data. The training-test ratio is set to 8:2.

Table 2. Link Prediction Performance Comparison

Method	**Accuracy**	**F1**	**AUC**
Logistic Regression	0.8665	0.8712	0.8735
GAT	0.9232	0.9235	0.9657
GraphSage	0.9295	0.9294	0.9681
Ours	0.9653	0.9659	0.9830

The result in Table 2 indicates that our method outperforms both conventional GAT and GraphSage models. By incorporating a self-attention layer, our model can selectively integrate and weigh the diverse information provided by the augmented embeddings, capturing subtle semantic nuances and relationships

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97635-3_23

essential for link prediction. It is worth noting that even a Logistic Regressor can reach 0.8665 accuracy, because the embeddings already capture semantic and structural information and separate the dataset, making it easier for simple models to perform well. We explore this further in the *Ablation Study* section.

3.3 Label Prediction

Which relationship does this edge belong to? In this subsection, we solved this problem by finetuning a RoBERTa model. We prepare a dataset comprising every entity pair identified by the GAT module, each record containing the pair, their shared competency questions, and the target relationship. Using few-shot GPT-4o prompting, we generate up to three candidate labels per pair (3,098 labels over 16,455 pairs), embed them with our fine-tuned Llama3.2:1b, cluster by cosine similarity, and manually refine to 27 final labels.

After finetuning, the model achieved an accuracy of 0.7356, precision of 0.7222, recall of 0.7356, and F1 score of 0.7207.

The performance of relationship label classifiers is suboptimal as it shows patterns of errors. The errors arise from contextual ambiguity, where multiple labels may apply, and confusion among semantically similar relations (e.g., causation vs. usage). To enhance accuracy, it is crucial to create a dataset that includes additional contextual details and competency questions to offer a deeper understanding of connections between entities.

4 Discussion

4.1 Embedding Component Contribution

The results clearly show that all these factors, intrinsic, hierarchy, and definition, have a positive impact on total performance, and integration of these provides the best results. In Table 3, we observe that when each of these factors is taken in isolation, intrinsic, hierarchy, and definition give the same results. But combining three components with a self-attention mechanism resulted in better results. For example, hierarchy and definition as a pair provide a higher degree of precision and F1 scores than other pairs of two factors, implying that these factors support each other well.

4.2 Ablation Study

Embeddings play an important role in our ontology augmentation framework. As shown in Table 2, a simple logistic regressor with finetuned llama3.2:1b can already reach 0.8665 accuracy. In this ablation study, we employed three different encodings to evaluate their performance using our model. The result of link prediction is shown in Fig 2. From this figure:

– The finetuned Llama3.2:1b performs better than vanilla Llama3.2:1b, indicating the finetune process can capture domain-specific differences between entities.

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97635-3_23

Table 3. Performance Metrics for Different Component Combinations

Component Combination	Accuracy	F1 Score	AUC
Intrinsic	0.9295	0.9294	0.9681
Hierarchy	0.9317	0.9318	0.9761
Definition	0.9281	0.9293	0.9765
Intrinsic + Hierarchy	0.9511	0.9521	0.9785
Intrinsic + Definition	0.9461	0.9475	0.9767
Hierarchy + Definition	0.9598	0.9603	0.9813
All Three (Intrinsic, Hierarchy, Definition)	0.9653	0.9659	0.9830

– Finetuned and vanilla Llama3.2:1b outperforms traditional embedding models like word2vec[12], indicating LLM has a deeper understanding of the underlying structure in our augmented ontology.

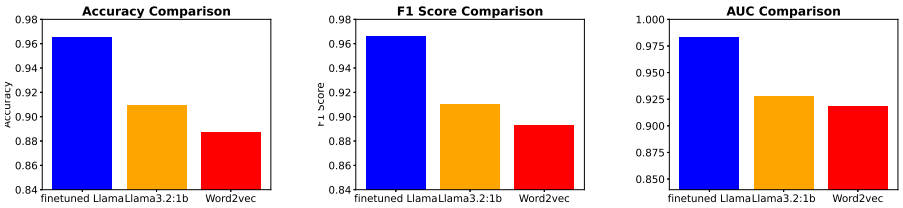

Fig. 2. Performance of various embedding models

5 Conclusion

This paper proposes a new paradigm for competency question based ontology enrichment using large language models, followed by high-level embedding fusion using a self-attention mechanism and graph attention network for robust link prediction. Our experiments demonstrate that the proposed method outperforms conventional methods such as GAT and GraphSage with improved accuracy and link prediction results. Despite some challenges in relationship label prediction, largely due to inherent vagueness and fine-grained semantic overlap, promising results indicate the potential to integrate deep learning and graph-based approaches to enhance the semantic density and structural coherence of ontologies. Future work will focus on enhancing the label prediction process by leveraging more contextual information, with the ultimate goal of further advancing the state-of-the-art in ontology enrichment.

Acknowledgments. This study was was partially funded by USDA Grant 2024-68015-41700.

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97635-3_23

Disclosure of Interests. Neither author has any competing interest at this time.

References

1. Achiam, J., Adler, S., Agarwal, S., Ahmad, L., Akkaya, I., Aleman, F.L., Almeida, D., Altenschmidt, J., Altman, S., Anadkat, S., et al.: Gpt-4 technical report. arXiv preprint arXiv:2303.08774 (2023)
2. 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)
3. Bezerra, C., Freitas, F., Santana, F.: Evaluating ontologies with competency questions. In: 2013 IEEE/WIC/ACM International Joint Conferences on Web Intelligence (WI) and Intelligent Agent Technologies (IAT). vol. 3, pp. 284–285. IEEE (2013)
4. Bosselut, A., Rashkin, H., Sap, M., Malaviya, C., Celikyilmaz, A., Choi, Y.: Comet: Commonsense transformers for automatic knowledge graph construction. arXiv preprint arXiv:1906.05317 (2019)
5. Chen, J., Hu, P., Jimenez-Ruiz, E., Holter, O.M., Antonyrajah, D., Horrocks, I.: Owl2vec*: Embedding of owl ontologies. Machine Learning **110**(7), 1813–1845 (2021)
6. Gangemi, A., Lippolis, A.S., Lodi, G., Nuzzolese, A.G.: Automatically drafting ontologies from competency questions with frodo. In: Towards a Knowledge-Aware AI, pp. 107–121. IOS Press (2022)
7. Hamilton, W., Ying, Z., Leskovec, J.: Inductive representation learning on large graphs. Advances in neural information processing systems **30** (2017)
8. Hertling, S., Paulheim, H.: Olala: Ontology matching with large language models. In: Proceedings of the 12th Knowledge Capture Conference 2023. pp. 131–139 (2023)
9. Hu, E.J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., Chen, W., et al.: Lora: Low-rank adaptation of large language models. ICLR **1**(2), 3 (2022)
10. Keet, C., Mahlaza, Z., Antia, M.: Claro: a data-driven cnl for specifying competency questions. ArXiv **abs/1907.07378** (2019)
11. Mateiu, P., Groza, A.: Ontology engineering with large language models. 2023 25th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing (SYNASC) pp. 226–229 (2023). https://doi.org/10.1109/SYNASC61333.2023.00038
12. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781 (2013)
13. Petroni, F., Rocktäschel, T., Lewis, P., Bakhtin, A., Wu, Y., Miller, A.H., Riedel, S.: Language models as knowledge bases? arXiv preprint arXiv:1909.01066 (2019)
14. Ristoski, P., Paulheim, H.: Rdf2vec: Rdf graph embeddings for data mining. In: The Semantic Web–ISWC 2016: 15th International Semantic Web Conference, Kobe, Japan, October 17–21, 2016, Proceedings, Part I 15. pp. 498–514. Springer (2016)
15. Zaitoun, A., Sagi, T., Peleg, M.: Generating ontology-learning training-data through verbalization. In: Proceedings of the AAAI Symposium Series. vol. 4, pp. 233–241 (2024)
16. Zhao, Y., Vetter, N., Aryan, K.: Using large language models for ontoclean-based ontology refinement. ArXiv **abs/2403.15864** (2024). https://doi.org/10.48550/arXiv.2403.15864

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97635-3_23