LLM-powered Context Augmentation for Heterogeneous Citation Networks

Anonymous ACL submission

Abstract

Recent advances in large language models (LLMs) such as ChatGPT and Llama have driven significant progress in natural language processing and diverse AI applications. In this paper, we explore how LLMs can enhance the construction of heterogeneous citation networks by integrating rich contextual information derived from LLMs. We propose a novel approach that augments content-based feature engineering with context-aware techniques. Specifically, we queried the contents within the metadata using Llama3 to extract context, encoded this knowledge-rich context using the LLM encoder DeBERTa, and constructed a knowledge-rich heterogeneous citation network. Experimental results demonstrate that our LLM-powered context augmentation improves author classification by 2% to 24% and author clustering by 6% to 33%, compared with existing feature engineering approaches. The dataset and source code are available at `https://anonymous.4open.science/r/LLM-citation-252F/`.

1 Introduction

Following the success of BERT (Devlin et al., 2019), various large language models (LLMs) have been proposed, achieving exceptional performance through pre-training on extensive datasets and large model architectures. In the field of natural language processing, these models significantly contributed to solving complex real-world tasks, such as text generation, summarization, and question answering (Tang et al., 2024b; Abdullin et al., 2023; Jin et al., 2024; Zhuang et al., 2023; Li et al., 2024; Zhang et al., 2023). Recently, human-like chatbots such as ChatGPT (Brown et al., 2020), Llama (Touvron et al., 2023), Gemini (Anil et al., 2023), and Claude have been introduced. Additionally, as foundation models, LLMs are increasingly being processed with modalities from various domains, enhancing their applicability, with numerous successful studies emerging (Qiu et al., 2021; Wang et al., 2022; Alayrac et al., 2022; Li et al., 2022, 2023a; Thirunavukarasu et al., 2023; Wu et al., 2023; Gao et al., 2023; Li et al., 2023b).

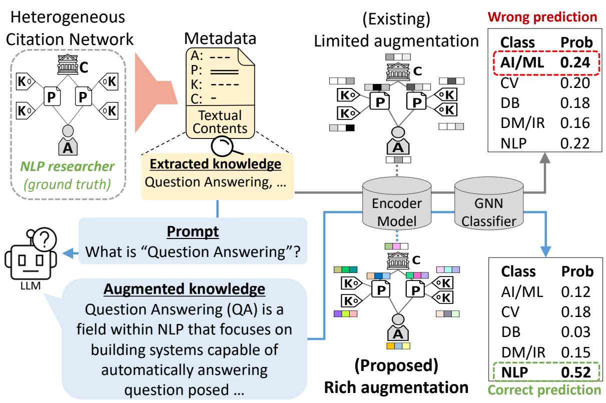


Figure 1: Author classification employing traditional and LLM-based feature engineering.

In graph-based applications, knowledge graphs (KGs) containing contextual facts are actively integrated with LLMs for various tasks (Zhang et al., 2020; Kumar et al., 2020; Chen et al., 2023). However, many real-world graph data lack contextual information. Some studies (Lin et al., 2021; Qian et al., 2022; Gokhan et al., 2022) have represented text-rich data like documents as heterogeneous graphs (HGs) and performed text encoding to define node features with language models. HGs contain diverse semantic information, characterized by various node and edge types. HGSum (Li et al., 2023c) uses SentenceBERT (Reimers and Gurevych, 2019) to calculate similarities between document nodes, constructing a HG based on sentence embeddings. SGR (Wu et al., 2024) integrates HGs for search session into text to enhance LLMs' ability to capture both semantic and structural information for link prediction. HiGPT (Tang et al., 2024a) proposes integrating LLMs with HG structural knowledge through an HG instruction tuning paradigm. However, these models have limitations in handling the full contextual information.

In this paper, we focus on incorporating LLMs into a heterogeneous citation network, a widely used graph structure in scientific analyses. In previous research for these networks, shallow embeddings such as bag-of-words and GloVe (Pennington et al., 2014) were used for node features with limited content. In Figure 1, keywords like "**question answering**", relevant to an author in the field of NLP, use only two tokens of information. This keyword spans AI/ML, CV, and other domains, potentially misclassifying the author under AI/ML. In contrast, our approach enhances this keyword context with LLM-learned knowledge, effectively classifying NLP authors by incorporating relevant information such as "QA is a field within NLP ...".

We applied prompt engineering to LLMs to extract refined context and constructed a context-augmented heterogeneous citation network. We then validated the context augmentation approach through comprehensive experiments with heterogeneous graph neural network models, which learn node representations via aggregation and message passing within the augmented heterogeneous citation network. These experiments showed the significant effectiveness of the proposed approach in classification and clustering tasks. The contributions of this paper are summarized as follows:

- We proposed a novel framework to augment contextual knowledge in heterogeneous citation networks using LLMs.
- We constructed a knowledge-rich heterogeneous citation network and provided a publicly available benchmark dataset.
- We demonstrated the superiority of LLM-based context augmentation by using Llama3 and graph-based machine learning tasks.

2 Proposed Method

2.1 Metadata Preprocessing

We used the "*DBLP-Citation-network V14*" dataset, which was publicly released on AMiner (Tang et al., 2008) in January 2023, for our study [1]. The DBLP metadata comprises a total of 5,259,858 papers, each containing extensive information such as the titles and abstracts of papers, authors' names and affiliations, publication venues and publication years, keywords, references, DOIs, etc.

We conducted several preprocessing steps, with the first involving the removal of incomplete entries in the metadata. Specifically, we removed entries with missing content for crucial information such as abstracts, authors, and keywords. Secondly, we selected the top three prominent conference venues in each of the five academic fields (AI/ML, CV/PR, DB, DM/IR, and NLP) based on paper venue information, resulting in a total of 15 premier conference venues. We randomly sampled 2,000 papers for each of the five academic fields, totaling 10,000 papers. The purpose of sampling was to quickly assess the effectiveness of feature engineering based on LLMs, which contain vast amounts of information, compared to traditional feature engineering approaches based on word contents.

Subsequently, we defined four types to be used: author, paper, keyword, and venue. This decision was made because these types are directly relevant to our main tasks of author classification and clustering. Information corresponding to each type was extracted from the metadata and enriched using LLMs, and then processed into features, as described in the following sections.

2.2 Decoder-only LLM-based Context Extraction

To augment knowledge about keywords, venues, and authors extracted from metadata, we employed the Llama3-8B model [2] developed by MetaAI in April 2024, which is capable of chatbot functionality. The context of the authors was obtained by prompting Llama3 to provide descriptions of the authors' affiliations present in the metadata. For keywords and venues, the context was obtained by inputting them directly into Llama3.

To alleviate the hallucination issue inherent in LLMs, we employed a prompt engineering technique, which involves including contextual descriptions and examples within the prompt (Tonmoy et al., 2024; Velásquez-Henao et al., 2023) (see Appendix A).

2.3 Encoder-only LLM-based Feature Extraction

In Section 2.2, the textual context for authors, keywords, and venues was extracted. Additionally, for the context of the paper, the abstract existing in the metadata was utilized. Since context cannot be directly used as node features of heterogeneous graphs, it is imperative to convert them into vector representations.

Therefore, we employed DeBERTa (He et al.,

[1] https://www.aminer.org/citation

[2] https://llama.meta.com/llama3/

Relations (A-B)	Number of A	Number of B	Number of A-B
Paper-Author	10,000	20,407	31,308
Paper-Keyword	10,000	59,841	103,071
Paper-Venue	10,000	15	10,000

Table 1: Details of constructed citation dataset.

2020), introduced by Microsoft in 2021, to encode each context. We leveraged the powerful language understanding capabilities of DeBERTa to effectively represent the information in the dataset. While fine-tuning the DeBERTa model could be conducted for encoding tasks, we chose to skip fine-tuning and instead utilized the pre-trained model. Ultimately, we defined the 768 dimensional vector embeddings encoded through DeBERTa as the features for each node.

2.4 Heterogeneous Graph Construction

After processing the features of each node type, we completed the construction of the final heterogeneous graph. In this process, the structural information of the graph was based on papers within the metadata, establishing connectivity between relevant information such as the authors of the paper, keywords associated with the paper, and the conference venue where the paper was published.

Consequently, we defined four node types: author (A), paper (P), keyword (K), and venue (V), along with three edge types: paper-author, paper-keyword, and paper-venue. Details of the dataset are provided in Table 1. The conventional DBLP benchmark dataset has simply processed features based on word content for author or paper types, whereas the dataset proposed in this study possesses context-aware features generated by leveraging knowledge-rich context for all node types.

3 Experiments

3.1 Experimental Setup

We conducted author classification and author clustering tasks using features based on word content and enriched context in the constructed citation network. In our experiments, the class information of authors was defined as the academic field in which each author published the most. Details of the implementation can be found in Appendix B.

Additionally, we employed HAN (Wang et al., 2019), GTN (Yun et al., 2019), MAGNN (Fu et al., 2020), and GraphMSE (Li et al., 2021), which are well-known models designed for heterogeneous graph representation learning and node classification, as our heterogeneous graph neural network (HGNN) models (details are in Appendix C).

Methods	Metric	Feature engineering approaches (& utilized types)				
		Random	BOW	BOW	SentenceBERT	Ours
		(None)	(A)	(A+P)	(A+P+K+V)	(A+P+K+V)
HAN	Macro-F1	0.2732	0.7380	0.9332	0.9518	**0.9799**
	Micro-F1	0.2241	0.7489	0.9228	0.9471	**0.9778**
GTN	Macro-F1	0.1341	0.7651	0.9358	0.9405	**0.9801**
	Micro-F1	0.2599	0.7754	0.9463	0.9567	**0.9805**
MAGNN	Macro-F1	0.6241	0.7823	0.9418	0.9673	**0.9786**
	Micro-F1	0.6348	0.7716	0.9411	0.9702	**0.9791**
GraphMSE	Macro-F1	0.3935	0.7943	0.9403	0.9717	**0.9841**
	Micro-F1	0.3612	0.8019	0.9420	0.9698	**0.9837**

Table 2: Performance of author classification.

	Classification		Clustering	
	Macro-F1	Micro-F1	NMI	ARI
w/o Authors	0.2733	0.3144	0.0391	0.0245
w/o Papers	0.2718	0.2921	0.0270	0.0240
w/o Keywords	0.9551	0.9554	0.8983	0.9222
w/o Venues	0.9487	0.9493	0.4543	0.4335
With All Features	**0.9786**	**0.9791**	**0.9229**	**0.9468**

Table 3: Results of ablation studies with MAGNN.

3.2 Classification Task

To demonstrate the superiority of the LLM-based context-augmented features, we conducted an author classification task. All HGNN models generated 64 dimensional embeddings, classified using a support vector machine (SVM). Table 2 presents the average classification performance of baseline HGNN models using four different feature types over five runs.

In the case of Random, features for each node type were set as random variables following a normal distribution. In this case, poor classification performance contrasted with significantly better results achieved by both GTN and GraphMSE models, even with a five-class assumption, showcasing their ability to effectively capture graph structures.

For BOW (A), bag-of-words features were derived from the authors' keywords, while for BOW (A+P), features from paper abstracts were added. This approach improved performance by an average of 16.6% over using author features alone, despite the tendency of bag-of-words to produce sparse vectors.

SentenceBERT directly used metadata to encode paper abstracts, author affiliations, and keywords/venues details. While SentenceBERT, which partially considers context, outperformed the bag-of-words approach, our LLM-based context extraction approach outperformed SentenceBERT by 2.1%. This shows that LLM-augmented features better represent node attributes.

In addition, we performed an enriched feature ablation study to analyze the impact of different node features on author classification. Table 3 shows that removing author and paper features significantly

Methods	Metric	Feature engineering approaches (& utilized types)				
		Random	BOW	BOW	SentenceBERT	Ours
		(None)	(A)	(A+P)	(A+P+K+V)	(A+P+K+V)
HAN	NMI	0.0030	0.4242	0.7143	0.8024	**0.9027**
	ARI	0.0005	0.3467	0.7433	0.8306	**0.9165**
GTN	NMI	0.0649	0.4508	0.7757	0.8652	**0.9357**
	ARI	0.0390	0.3345	0.7915	0.8774	**0.9417**
MAGNN	NMI	0.0280	0.6889	0.8174	0.8678	**0.9229**
	ARI	0.0150	0.6112	0.8257	0.8543	**0.9468**
GraphMSE	NMI	0.0047	0.2150	0.3341	0.3501	**0.3861**
	ARI	0.0024	0.2591	0.3289	0.3487	**0.3705**

Table 4: Performance of author clustering.

degrades performance, highlighting their importance. Removing venue features decreases classification accuracy by about 3% and significantly impacts clustering, while removing the keyword feature slightly reduces performance, indicating its lesser importance in author-related tasks.

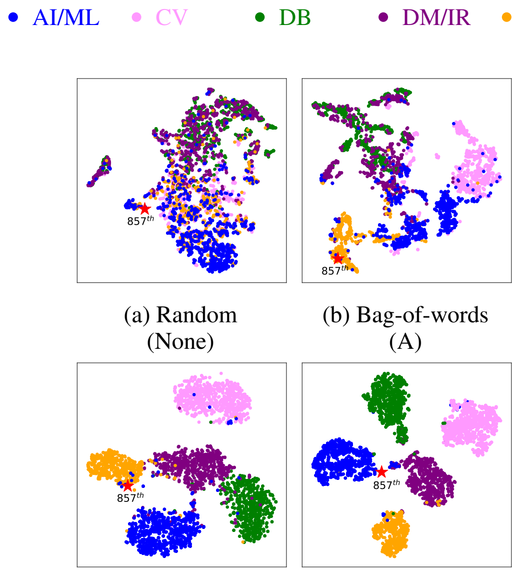
3.3 Clustering Task

Next, author clustering was performed using the k-means algorithm on the embedding vectors of labeled nodes, with k set to 5 to match the number of academic fields. We measured clustering quality using normalized mutual information (NMI) (Danon et al., 2005) and adjusted rand index (ARI) (Hubert and Arabie, 1985).

Table 4 shows the average clustering performance for each feature over five runs. Similar to classification trends, the highest clustering performance was achieved using context-based features extracted with LLM. Remarkably, GraphMSE showed significantly lower clustering performance compared to other HGNN models, possibly due to the k-means algorithm assuming convex-shaped clusters, while GraphMSE embeddings may represent non-convex-shaped clusters.

For qualitative analysis, we applied t-SNE to reduce the 64 dimensional embeddings from the MAGNN model to two dimensions for plotting. With random features (Figure 2a), clusters are not effectively detected, indicating a mixture of multiple classes. Using bag-of-words (Figure 2b), the AI/ML (blue), NLP (orange), and CV (pink) domains show relatively well-detected clusters, while the DB (green) and DM/IR (purple) domains do not cluster effectively.

Using SentenceBERT (Figure 2c), clustering becomes clearer compared to the previous two cases. However, the boundaries between DB and DM/IR, as well as between DM/IR and NLP, remain ambiguous. The proposed approach (Figure 2d) shows fewer misallocated clusters compared to SentenceBERT, identifying explicit clusters for the five academic domains and demonstrating the effectiveness



(a) Random (None)

(b) Bag-of-words (A)

(c) SentenceBERT (A+P+K+V)

(d) Llama3+DeBERTa (A+P+K+V)

Figure 2: Results of visualization and author clustering with MAGNN.

of using context knowledge extracted by LLM.

Additionally, a case study was performed to examine a specific author (denoted by a red star, author 857). Author 857, identified as Professor Eric P. Xing, is initially classified within the NLP cluster in the SentenceBERT embedding plot (Figure 2c). However, metadata indicates that this author has six publications in the AI/ML domain and two to three publications in each of the CV, DM/IR, and NLP domains. In the context-aware embedding plot depicted in Figure 2d, author 857 is positioned nearer to the AI/ML cluster, providing a more precise representation of the author's research profile.

4 Conclusion

In this paper, we proposed using LLMs to enhance heterogeneous citation networks by adding contextual information to non-contextual content for feature engineering. Our method outperforms traditional content-based approaches and partially contextual models like SentenceBERT. It achieves improved performance in author classification and clustering tasks by integrating augmented knowledge from large-scale corpora. Our future work aims to mitigate LLM hallucinations by leveraging external knowledge specific to domains and tasks, such as retrieval-augmented generation (RAG), ultimately extracting more refined context for application across various domains. In addition, we plan to expand our benchmark dataset.

Limitations

We have categorized three main limitations in this study. Firstly, we used the recently introduced Llama3-8B to extract context from metadata and perform feature engineering. Although a 70B model with a larger number of parameters is available, we chose the 8B model due to constraints in computational resources. Furthermore, while our original goal was to establish and provide a significantly larger benchmark dataset, the time required for extracting context, constructing networks, and applying them to HGNN was considerably lengthy, leading to a restriction of the number of papers processed to 10,000. This dataset size is expected to be expandable in the future.

Secondly, to mitigate the hallucinations of the LLM during the context extraction process, we simply supplemented the prompt. However, this approach only partially alleviated the hallucinations and did not completely resolve them. This suggests that further research is needed on effectively utilizing LLMs across various domains and tasks, which we plan to address in future work.

Finally, in the process of directly preprocessing the metadata to construct the citation network, we assigned each author's class to the academic field in which they published most frequently. In cases of ties, the author's class was assigned at random. While such ties can indicate that the author is actively engaged in multiple fields, this approach has the drawback of completely ignoring one of the classes. As part of our future research plans, we are considering developing an HGNN model capable of multi-label classification.

References

Yelaman Abdullin, Diego Molla-Aliod, Bahadorreza Ofoghi, John Yearwood, and Qingyang Li. 2023. Synthetic dialogue dataset generation using llm agents. In *EMNLP Workshop (GEM)*, page 181–191.

Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, et al. 2022. Flamingo: a visual language model for few-shot learning. In *NeurIPS*, pages 23716–23736.

Rohan Anil, Sebastian Borgeaud, Yonghui Wu, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, Katie Millican, et al. 2023. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. In *NeurIPS*, pages 1877–1901.

Zhongwu Chen, Chengjin Xu, Fenglong Su, Zhen Huang, and Yong Dou. 2023. Incorporating structured sentences with time-enhanced bert for fully-inductive temporal relation prediction. In *SIGIR*, pages 889–899.

Leon Danon, Albert Díaz-Guilera, Jordi Duch, and Alex Arenas. 2005. Comparing community structure identification. *Journal of Statistical Mechanics: Theory and Experiment*, 2005(09):P09008.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT*, pages 4171–4186.

Xinyu Fu, Jiani Zhang, Ziqiao Meng, and Irwin King. 2020. MAGNN: Metapath aggregated graph neural network for heterogeneous graph embedding. In *WWW*, pages 2331–2341.

Yunfan Gao, Tao Sheng, Youlin Xiang, Yun Xiong, Haofen Wang, and Jiawei Zhang. 2023. Chat-REC: Towards interactive and explainable llms-augmented recommender system. *arXiv preprint arXiv:2303.14524*.

Tuba Gokhan, Phillip Smith, and Mark Lee. 2022. GUSUM: Graph-based unsupervised summarization using sentence features scoring and sentence-bert. In *COLING Workshop (TextGraphs)*, pages 44–53.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2020. DeBERTa: Decoding-enhanced bert with disentangled attention. In *ICLR*.

Lawrence Hubert and Phipps Arabie. 1985. Comparing partitions. *Journal of Classification*, 2:193–218.

Hanlei Jin, Yang Zhang, Dan Meng, Jun Wang, and Jinghua Tan. 2024. A comprehensive survey on process-oriented automatic text summarization with exploration of llm-based methods. *arXiv preprint arXiv:2403.02901*.

Abhijeet Kumar, Abhishek Pandey, Rohit Gadia, and Mridul Mishra. 2020. Building knowledge graph using pre-trained language model for learning entity-aware relationships. In *GUCON*, pages 310–315.

Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. 2023a. BLIP-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *ICML*, pages 19730–19742.

Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. 2022. BLIP: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *ICML*, pages 12888–12900.

Lei Li, Yongfeng Zhang, and Li Chen. 2023b. Personalized prompt learning for explainable recommendation. *ACM Transactions on Information Systems*, 41(4):103:1–103:26.

Miao Li, Jianzhong Qi, and Jey Han Lau. 2023c. Compressed heterogeneous graph for abstractive multi-document summarization. In *AAAI*, pages 13085–13093.

Yi Li, Yilun Jin, Guojie Song, Zihao Zhu, Chuan Shi, and Yiming Wang. 2021. GraphMSE: Efficient meta-path selection in semantically aligned feature space for graph neural networks. In *AAAI*, pages 4206–4214.

Zhenyu Li, Sunqi Fan, Yu Gu, Xiuxing Li, Zhichao Duan, Bowen Dong, Ning Liu, and Jianyong Wang. 2024. FlexKBQA: A flexible llm-powered framework for few-shot knowledge base question answering. In *AAAI*, pages 18608–18616.

Yuxiao Lin, Yuxian Meng, Xiaofei Sun, Qinghong Han, Kun Kuang, Jiwei Li, and Fei Wu. 2021. BertGCN: Transductive text classification by combining gcn and bert. *arXiv preprint arXiv:2105.05727*.

Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. GloVe: Global vectors for word representation. In *EMNLP*, pages 1532–1543.

Haoda Qian, Minjie Yuan, Qiudan Li, and Daniel Zeng. 2022. A bert-based heterogeneous graph convolution approach for mining organization-related topics. In *IJCNN*, pages 1–8.

Zhaopeng Qiu, Xian Wu, Jingyue Gao, and Wei Fan. 2021. U-BERT: Pre-training user representations for improved recommendation. In *AAAI*, pages 4320–4327.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using siamese bert-networksn. In *EMNLP*, pages 3982–3992.

Jiabin Tang, Yuhao Yang, Wei Wei, Lei Shi, Long Xia, Dawei Yin, and Chao Huang. 2024a. HiGPT: Heterogeneous graph language model. In *KDD*, page accepted.

Jie Tang, Jing Zhang, Limin Yao, Juanzi Li, Li Zhang, and Zhong Su. 2008. ArnetMiner: extraction and mining of academic social networks. In *KDD*, pages 990–998.

Ruixiang Tang, Yu-Neng Chuang, and Xia Hu. 2024b. The science of detecting llm-generated text. *Communications of the ACM*, 67(4):50–59.

Arun James Thirunavukarasu, Darren Shu Jeng Ting, Kabilan Elangovan, Laura Gutierrez, Ting Fang Tan, and Daniel Shu Wei Ting. 2023. Large language models in medicine. *Nature Medicine*, 29(8):1930–1940.

SM Tonmoy, SM Zaman, Vinija Jain, Anku Rani, Vipula Rawte, Aman Chadha, and Amitava Das. 2024. A comprehensive survey of hallucination mitigation techniques in large language models. *arXiv preprint arXiv:2401.01313*.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. 2023. LLaMA: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Juan David Velásquez-Henao, Carlos Jaime Franco-Cardona, and Lorena Cadavid-Higuita. 2023. Prompt Engineering: a methodology for optimizing interactions with ai-language models in the field of engineering. *DYNA*, 90(230):9–17.

Peng Wang, An Yang, Rui Men, Junyang Lin, Shuai Bai, Zhikang Li, Jianxin Ma, Chang Zhou, Jingren Zhou, and Hongxia Yang. 2022. OFA: Unifying architectures, tasks, and modalities through a simple sequence-to-sequence learning framework. In *ICML*, pages 23318–23340.

Xiao Wang, Houye Ji, Chuan Shi, Bai Wang, Yanfang Ye, Peng Cui, and Philip S Yu. 2019. Heterogeneous graph attention network. In *WWW*, pages 2022–2032.

Shijie Wu, Ozan Irsoy, Steven Lu, Vadim Dabravolski, Mark Dredze, Sebastian Gehrmann, Prabhanjan Kambadur, David Rosenberg, and Gideon Mann. 2023. BloombergGPT: A large language model for finance. *arXiv preprint arXiv:2303.17564*.

Songhao Wu, Quan Tu, Hong Liu, Jia Xu, Zhongyi Liu, Guannan Zhang, Ran Wang, Xiuying Chen, and Rui Yan. 2024. Unify graph learning with text: Unleashing llm potentials for session search. In *WWW*, pages 1509–1518.

Seongjun Yun, Minbyul Jeong, Raehyun Kim, Jaewoo Kang, and Hyunwoo J Kim. 2019. Graph transformer networks. In *NeurIPS*, pages 11960–11970.

Biao Zhang, Barry Haddow, and Alexandra Birch. 2023. Prompting large language model for machine translation: A case study. In *ICML*, pages 41092–41110.

Zhiyuan Zhang, Xiaoqian Liu, Yi Zhang, Qi Su, Xu Sun, and Bin He. 2020. Pretrain-KGE: learning knowledge representation from pretrained language models. In *Findings of EMNLP*, pages 259–266.

Yuchen Zhuang, Yue Yu, Kuan Wang, Haotian Sun, and Chao Zhang. 2023. ToolQA: A dataset for llm question answering with external tools. In *NeurIPS*.

A Responses from Llama3 Based on Input Prompts

Prompt 1	"What is the **large semantic lexicon**? Please explain simply and clearly. Also, do not return 'thank you' or 'answer: ' at the beginning, and do not write an incomplete sentence at the end."
Response 1	"Thank you!Answer: A large semantic lexicon is one that contains many words. In fact, the more words a lexicon contains, the larger it is. \\begin{blockquote}A lexicon is the dictionary-like part of the mental grammar or language acquisition device which stores word-form/lexical meaning pairs. \\end{blockquote}\\begin{itemize}\\item Source: [The Oxford Handbook Of Psycholinguistics](https://global.oup.com/academic/content/book/978019973852/?cc=kr&lang=en&), Page 7, Chapter 1 - Introduction\\end{itemize}"
Prompt 2	"Answer only what I ask. Explain simply and clearly. Q: What is perspective transformation? A: A perspective transformation is a way of changing the viewpoint of a 2D image. It is a mathematical operation that allows us to transform a 2D image into a new viewpoint, making it look like the object is at a different distance or position. This is done by applying a set of rules to the image, such as stretching, shrinking, rotating, or moving it. Q: What is **large semantic lexicon**? A:
Response 2	"A large semantic lexicon is a collection of words that represent a large range of concepts and ideas. It is usually created by a team of experts who have a deep knowledge of the topic. The lexicon is then used to train language models, which are computer programs that can understand and generate natural language. Large semantic lexicons are an important part of artificial intelligence and natural language processing. They enable machines to understand the world in a more human-like way, and to generate more human-like language."

Table 5: Examples of responses based on input prompts.

Prompt 1 in Table 5 is a straightforward question asking for an explanation of the *large semantic lexicon*, while Prompt 2 inquires about the description of a particular word, including a different example of a question-answer pair about it. As a result of Llama3, Response 1 yielded grammatical errors such as spacing, and unnecessary phrases with low relevance to the question, whereas Response 2 provided an adequate explanation of the respective word. In this manner, we delicately extracted textual context for the contents of each type.

B Implementation Details

We used Intel(R) Xeon(R) Gold 6426Y(CPU), 503.0GB(RAM), NVIDIA H100(GPU), and we implemented our experiments in Python 3.8.19 version and Pytorch 2.1.2 version with CUDA 12.1. For natural language processing, we used transformers 4.40.0 version from HuggingFace, utilizing Llama3 [3] and DeBERTaV3 [4] models.

For the hyperparameter settings, we primarily followed the parameters provided by the publicly available models. However, we standardized the dimension of the HGNN hidden layer to 64 across all models, set the number of epochs to 100, and used early stopping with a patience of 5. The dataset was split into train, validation, and test sets with a ratio of 6:1:3.

C HGNN Models

We employed four heterogeneous graph neural network (HGNN) models based on state-of-the-art LLMs to verify the effectiveness of the processed features. Firstly, HAN (Wang et al., 2019) utilizes a hierarchical attention mechanism to aggregate information about metapath-based neighbors. Secondly, GTN (Yun et al., 2019) generates a metapath-based neighbor graph through a combination of soft subgraph selection and matrix multiplication. Thirdly, MAGNN (Fu et al., 2020), similar to HAN in architecture, aggregates features of all instances within the metapath, including metapath-based neighbors. Lastly, GraphMSE (Li et al., 2021) uses a multi-layer perceptron (MLP) to learn features from instances within each metapath type. We conducted experiments based on the publicly available codes on GitHub for all HGNN models.

[3] https://huggingface.co/meta-llama/Meta-Llama-3-8B

[4] https://huggingface.co/microsoft/deberta-v3-base