Exploring Large Language Models for Knowledge Graph Completion

Anonymous EMNLP submission

Abstract

Knowledge graphs play a vital role in numerous artificial intelligence tasks, yet they frequently face the issue of incompleteness. In this study, we explore utilizing Large Language Models (LLM) for knowledge graph completion. We consider triples in knowledge graphs as text sequences and introduce an innovative framework called Knowledge Graph LLM (KG-LLM) to model these triples. Our technique employs entity and relation descriptions of a triple as prompts and utilizes the response for predictions. Experiments on various benchmark knowledge graphs demonstrate that our method attains state-of-the-art performance in tasks such as triple classification and relation prediction. We also find that fine-tuning relatively smaller models (e.g., LLaMA-7B, ChatGLM-6B) outperforms recent ChatGPT and GPT-4.

1 Introduction

Large knowledge graphs (KG) like FreeBase (Bollacker et al., 2008), YAGO (Suchanek et al., 2007), and WordNet (Miller, 1995) serve as a powerful foundation for numerous critical AI tasks, including semantic search, recommendation (Zhang et al., 2016), and question answering (Cui et al., 2017). A KG is generally a multi-relational graph with entities as nodes and relations as edges. Each edge is depicted as a triplet (*head entity*, relation, *tail entity*) (abbreviated as (h, r, t)), signifying the relationship between two entities, for instance, (*Steve Jobs*, founded, *Apple Inc.*). Despite their effectiveness, knowledge graphs remain incomplete. This issue leads to the challenge of *knowledge graph completion*, which aims to evaluate the plausibility of triples that are not present in a knowledge graph.

A significant amount of research has been dedicated to knowledge graph completion. One prevalent method is knowledge graph embedding (Wang et al., 2017). However, most knowledge graph embedding models solely rely on structural information from observed triple facts, leading to issues arising from the sparsity of knowledge graphs. A number of studies integrate textual information to enhance knowledge representation (Socher et al., 2013; Xie et al., 2016; Xiao et al., 2017; Wang and Li, 2016; Xu et al., 2017; An et al., 2018). KG-BERT (Yao et al., 2019) firstly employs the pre-trained language model BERT (Devlin et al., 2019) to encode prior knowledge and contextual information. The KG-BERT model was extended by several recent studies (Wang et al., 2021, 2022; Lovelace and Rose, 2022; Youn and Tagkopoulos, 2023) on both efficiency and performance, but the models used in these works are relatively small.

Recently, large language models (Zhao et al., 2023) like ChatGPT and GPT-4 (OpenAI, 2023) have gained significant attention. Researchers find that scaling pre-trained language models often leads to an improved model capacity on downstream tasks. These large-sized models show different behaviors from smaller models like BERT and display surprising abilities in solving a series of complex tasks.

In this study, we propose a novel method for knowledge graph completion using large language models. Specifically, we treat entities, relations, and triples as textual sequences and model knowledge graph completion as a sequence-to-sequence problem. We perform instruction tuning with open LLMs (LLaMA (Touvron et al., 2023) and ChatGLM (Du et al., 2022)) on these sequences for predicting the plausibility of a triple or a candidate entity/relation. The method achieves stronger performance in several KG completion tasks. Our source code is available at: `https://anonymous.4open.science/r/kg-llm-527B/`. Our contributions are summarized as follows:

- We propose a new language modeling method for knowledge graph completion. To the best of our knowledge, this is the first study to systematically investigate large language models for KG completion tasks.

- Results on several benchmarks show that our method achieves state-of-the-art results in triple classification and relation prediction. We also find that fine-tuning relatively smaller models (e.g., LLaMA-7B, ChatGLM-6B) can outperform recent ChatGPT and GPT-4.

2 Related Work

2.1 Knowledge Graph Completion

Comprehensive reviews of knowledge graph completion techniques have been carried out by (Wang et al., 2017) and (Ji et al., 2021). These techniques can be grouped into two categories based on their scoring functions for triple (h, r, t): translational distance models like TransE (Bordes et al., 2013) and semantic matching models like DistMult (Yang et al., 2015). Convolutional neural networks have also demonstrated promising results in knowledge graph completion (Dettmers et al., 2018; Nguyen et al., 2018; Nathani et al., 2019).

The methods mentioned above perform knowledge graph completion using only the structural information found in triples. However, incorporating various types of external information, such as entity types, logical rules, and textual descriptions, can enhance performance (Wang et al., 2017; Ji et al., 2021). For textual descriptions, Socher et al. (2013) initially represented entities by averaging the word embeddings in their names, with the embeddings learned from an external corpus. Wang et al. (2014a) suggested embedding entities and words in the same vector space by aligning Wikipedia anchors with entity names. Xie et al. (2016) employed convolutional neural networks (CNN) to encode word sequences in entity descriptions. There are also a number of studies in this line of works (Xiao et al., 2017; Wang and Li, 2016; Xu et al., 2017; An et al., 2018). Yao et al. (2019) proposed KG-BERT which improves the above methods with pre-trained language models (PLMs). Recently, Wang et al. (2021, 2022); Lovelace and Rose (2022) extended cross-encoder in KG-BERT to bi-encoder, which enhances the performance and inference efficiency. Similar to this work, KGT5 (Saxena et al., 2022) and KG-S2S (Chen et al., 2022) treat KG completion as sequence-to-sequence tasks. However, the pre-trained language models used in these studies are relatively small.

Compared with these methods, our method utilizes more powerful large language models with emergent abilities not present in small models such as in-context learning, instruction following, and step-by-step reasoning. These abilities are helpful for KG completion tasks.

2.2 LLMs with KG Completion

Recently, Zhao et al. (2023) presents a comprehensive survey of LLMs that describes knowledge completion as a basic evaluation task of LLMs. Two closely related studies (Xie et al., 2023; Zhu et al., 2023) evaluate ChatGPT and GPT-4 on a link prediction task in KG. Our study is inspired by these works, but we further provide more comprehensive results for KG completion and perform instruction tuning on three tasks.

3 Method

3.1 Knowledge Graph Completion Tasks

In this chapter, we describe the three tasks in knowledge graph completion: triple classification, relation prediction, and entity (link) prediction, and how to transform them into simple prompt questions for LLM to complete the tasks. The entire process is depicted in Figure 1.

Triple Classification. Given a triple (h, r, t), the task is to classify it as correct or incorrect. For example, given the triple <*Steve Jobs*, founded, *Apple Inc.*>, the task is to classify it as correct. The prompt formation would be "Is this true: Steve Jobs founded Apple Inc.?". And the ideal output of LLM would be "Yes, this is true."

Relation Prediction. Given a head entity and a tail entity, the task is to predict the relationship between them. For example, given the head entity "Steve Jobs" and the tail entity "Apple Inc.", the task is to predict that their relationship is "founded". The prompts formation would be "What is the relationship between Steve Jobs and Apple Inc.? Please choose your answer from: was born in | founded | is citizen of | . . . . . . | plays for." And the desired response would be "Steve Jobs founded Apple Inc."

Entity (link) Prediction. Given a head entity and a relationship, the task is to predict the tail entity related to the head entity. Given a tail entity and a relationship, the task is to predict the head entity. For example, given the head entity "Steve Jobs" and the relationship "founded", the task is to predict the tail entity "Apple Inc.". The prompts formation would be "Steve Jobs founded" for asking the tail entity and "What/Who/When/Where/Why founded

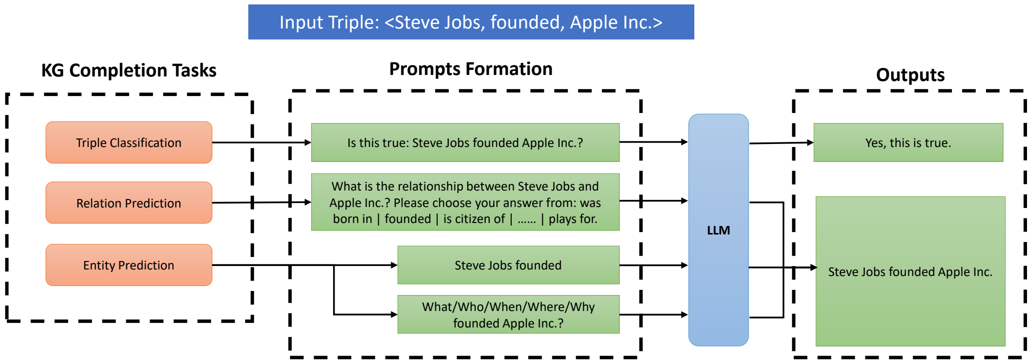

Figure 1: Illustrations of Large Langauge Models (LLMs) for Knowledge Graph (KG) Completion.

Apple Inc.?" for asking the head entity. The ideal response would be "Steve Jobs founded Apple Inc."

3.2 Instruction Turning LLM with KG (KG-LLM)

In order to align LLMs with KG triples, we introduce KG-LLM, which instruction turns the pre-trained LLM to process KG data using the specific factual question-answering prompt paradigm. Specifically, We fine-tune two open LLMs: ChatGLM-6B (Du et al., 2022) with P-tuning v2 (Liu et al., 2021) and LLaMA (version 1 and 2) (Touvron et al., 2023) with LoRA (Hu et al., 2021) using prompts and responses of training triples in a KG. We name our fine-tuned models KG-ChatGLM-6B and KG-LLaMA (7B and 13B). We also incorporate structural information into training and test instructions. Specifically, for the entity prediction task, we sample $K = 5$ neighbor entities (excluding the target entity) for the given entity and tell the model as in the Appendix.

4 Experiments

Dataset	# Ent	# Rel	# Train	# Dev	# Test
WN11	38,696	11	112,581	2,609	10,544
FB13	75,043	13	316,232	5,908	23,733
WN18RR	40,943	11	86,835	3,034	3,134
YAGO3-10	123,182	37	1,079,040	5,000	5,000

Table 1: Summary statistics of datasets.

4.1 Datasets and Settings

We ran our experiments on four widely used benchmark KG datasets: WN11 (Socher et al., 2013), FB13 (Socher et al., 2013), WN18RR and YAGO3-10 (Dettmers et al., 2018). Table 1 provides statistics of all datasets we used. We used the same entity and relation text descriptions as in (Yao et al.,

Method	WN11	FB13	Avg.
NTN (Socher et al., 2013)	86.2	90.0	88.1
TransE (Wang et al., 2014b)	75.9	81.5	78.7
TransH (Wang et al., 2014b)	78.8	83.3	81.1
TransR (Lin et al., 2015)	85.9	82.5	84.2
TransD (Ji et al., 2015)	86.4	89.1	87.8
TEKE (Wang and Li, 2016)	86.1	84.2	85.2
TransG (Xiao et al., 2016)	87.4	87.3	87.4
TranSparse-S (Ji et al., 2016)	86.4	88.2	87.3
DistMult (Zhang et al., 2018)	87.1	86.2	86.7
DistMult-HRS (Zhang et al., 2018)	88.9	89.0	89.0
AATE (An et al., 2018)	88.0	87.2	87.6
ConvKB (Nguyen et al., 2018)	87.6	88.8	88.2
DOLORES (Wang et al., 2018)	87.5	89.3	88.4
DKRL (BERT)	87.3	79.8	83.6
KG-BERT(a) (Yao et al., 2019)	93.5	90.4	91.9
KGT5	72.8	66.3	69.6
LLaMA-7B	21.1	9.1	15.1
LLaMA-13B	28.1	17.6	22.9
KG-LLaMA-7B	95.5	89.2	92.4
KG-LLaMA-13B	95.6	90.2	92.9
KG-LLaMA2-13B	**96.6**	**90.7**	**93.7**

Table 2: Triple classification accuracy (in percentage) for different methods. The baseline results with citations are obtained from corresponding papers.

Method	FB13-100
ChatGPT	0.90
GPT-4	**0.94**
LLaMA-7B	0.14
LLaMA-13B	0.16
KG-LLaMA-7B	0.93
KG-LLaMA-13B	**0.94**

Table 3: Triple classification accuracy on 100 test instances of FB13 for different LLMs.

2019). Due to the access limit of GPT-4, we randomly selected 100 test examples from FB13 and YAGO3-10 for evaluation, we name the subsets FB13-100 and YAGO3-10-100.

We compare KG-LLM with multiple KG embedding methods: TransE and its extensions TransH (Wang et al., 2014b), TransD (Ji et al., 2015), TransR (Lin et al., 2015), TransG (Xiao et al., 2016) and TranSparse (Ji et al., 2016), DistMult and its extension DistMult-HRS (Zhang et al., 2018). The neural tensor network NTN (Socher

Method	WN18RR	YAGO3-10	YAGO3-10-100
KG-BERT(a)	0.1102	–	–
StAR	0.2430	–	–
KGT5	0.1011	0.0484	0.12
KGLM	0.3050	–	–
ChatGPT	–	–	0.22
GPT-4	–	–	**0.24**
KG-ChatGLM-6B	0.1613	0.0455	0.11
LLaMA-7B	0.0849	0.0254	0.03
LLaMA-13B	0.0991	0.0276	0.01
KG-LLaMA-7B	0.2415	0.0782	0.16
KG-LLaMA-13B	0.2559	0.0872	0.13
KG-LLaMA2-13B	0.2682	0.0949	0.16
KG-LLaMA2-13B + Struct	**0.3151**	**0.1330**	0.22

Table 4: Entity (link) prediction Hits@1 for different methods. The baseline results with citations are obtained from corresponding papers.

Method	YAGO3-10	YAGO3-10-100
KG-BERT(b)	0.6816	–
KGT5	0.5714	0.60
ChatGPT	–	0.39
GPT-4	–	0.56
ChatGLM-6B	0.0658	0.07
KG-ChatGLM-6B	0.5662	0.58
LLaMA-7B	0.0348	0.13
LLaMA-13B	0.0040	0.01
KG-LLaMA-7B	**0.7028**	**0.71**
KG-LLaMA-13B	0.6968	0.64

Table 5: Relation prediction Hits@1 scores.

et al., 2013). CNN models: ConvKB (Nguyen et al., 2018). Contextualized KG embeddings: DOLORES (Wang et al., 2018). KG embeddings with textual information: TEKE (Wang and Li, 2016), DKRL (Xie et al., 2016) (BERT encoder), AATE (An et al., 2018). Pre-tained language models: KG-BERT (Yao et al., 2019), StAR (Wang et al., 2021), KGT5 (Saxena et al., 2022) and KGLM (Youn and Tagkopoulos, 2023). We also compare with ChatGPT and GPT-4.

For instruction tuning and inference of ChatGLM-6B, We used the default parameter settings in its public implementations. For LLaMA, we use the implementation in the Transformers Python library. More detailed settings can be found in our code. For KG completion models, we use the results in their original papers or reproduce the results using default configurations in their implementations. For KGT5, we use our prompts and responses for training, other settings are the same as its implementation. We input our designed prompts to the web interface of GPT-4 and ChatGPT to obtain results.

4.2 Results

Table 2 presents triple classification accuracy scores on WN11 and FB13. If the ground truth is true and the response contains affirmative words like "Yes" and "yes", or if the label is false and the response contains negative words like "No"/"no"/"not"/"n't", we label the response as correct. We find that LLaMA-7B and LLaMA-13B perform poorly on both WN11 and FB13. However, when instructed to process KG data, KG-LLaMA shows significant improvement compared to LLaMA. KG-LLaMA2-13B achieves the highest accuracy scores on the two KG data sets. Table 3 presents the accuracy scores of different LLMs on the 100 test instances of FB13. We manually label the response of different LLMs as correct or wrong. We find that KG-LLaMA performs well, the score is higher than ChatGPT and equal to GPT-4.

The link prediction hits@1 scores of various pre-trained language models on WN18RR and YAGO3-10 are presented in Table 4. The scores are the average for both head and tail entities. In the case of LLMs, the response is considered correct if it contains the label words. The results indicate a promising outcome in our paradigm, as KG-LLaMA shows significant improvements due to instruction turning. Incorporating structural information also improves the results by a large margin. Table 5 demonstrates that KG-LLaMA-7B produces the best relation prediction hits@1 on YAGO3-10, even better than GPT-4. KG-ChatGLM-6B also shows much better results. This indicates that instruction turning leads the LLM to extract knowledge stored in model parameters more efficiently.

Table 6 illustrates the differences in responses between LLM and KG-LLM given the same input. We found the answers of the original models are not satisfactory while instruction tuning can teach the models to answer like training triples and to be more aware of a fact.

The main reasons why KG-LLM performs well are: 1). LLMs contain more general knowledge compared with smaller pre-trained language models. 2). Instruction tuning fills the gap between the pre-trained weights in LLMs and KG triple descriptions.

5 Conclusions

In this work, we propose a novel method KG-LLM. Our approach attains SOTA performance in KG completion tasks such as triple classification and relation prediction. We plan to apply our KG-LLM as a knowledge-enhanced language model to other NLP tasks. Furthermore, we are going to explore more effective prompt engineering of LLM.

Limitations

Although our method has demonstrated promising results in KG completion using LLMs, it currently lacks the ability to handle KGs that lack textual names or descriptions for entities and relations. Additionally, we have not yet fully utilized the high-order KG structure information, which has the potential to significantly improve results, particularly in the entity prediction task.

References

Bo An, Bo Chen, Xianpei Han, and Le Sun. 2018. Accurate text-enhanced knowledge graph representation learning. In *NAACL*, pages 745–755.

Kurt Bollacker, Colin Evans, Praveen Paritosh, Tim Sturge, and Jamie Taylor. 2008. Freebase: a collaboratively created graph database for structuring human knowledge. In *SIGMOD*, pages 1247–1250.

Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. 2013. Translating embeddings for modeling multi-relational data. In *NIPS*, pages 2787–2795.

Chen Chen, Yufei Wang, Bing Li, and Kwok-Yan Lam. 2022. Knowledge is flat: A seq2seq generative framework for various knowledge graph completion. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 4005–4017.

Wanyun Cui, Yanghua Xiao, Haixun Wang, Yangqiu Song, Seung-won Hwang, and Wei Wang. 2017. KBQA: learning question answering over qa corpora and knowledge bases. *Proceedings of the VLDB Endowment*, 10(5):565–576.

Tim Dettmers, Pasquale Minervini, Pontus Stenetorp, and Sebastian Riedel. 2018. Convolutional 2d knowledge graph embeddings. In *AAAI*, pages 1811–1818.

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

Zhengxiao Du, Yujie Qian, Xiao Liu, Ming Ding, Jiezhong Qiu, Zhilin Yang, and Jie Tang. 2022. Glm: General language model pretraining with autoregressive blank infilling. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 320–335.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*.

Guoliang Ji, Shizhu He, Liheng Xu, Kang Liu, and Jun Zhao. 2015. Knowledge graph embedding via dynamic mapping matrix. In *ACL*, pages 687–696.

Guoliang Ji, Kang Liu, Shizhu He, and Jun Zhao. 2016. Knowledge graph completion with adaptive sparse transfer matrix. In *AAAI*.

Shaoxiong Ji, Shirui Pan, Erik Cambria, Pekka Marttinen, and S Yu Philip. 2021. A survey on knowledge graphs: Representation, acquisition, and applications. *IEEE transactions on neural networks and learning systems*, 33(2):494–514.

Yankai Lin, Zhiyuan Liu, Maosong Sun, Yang Liu, and Xuan Zhu. 2015. Learning entity and relation embeddings for knowledge graph completion. In *AAAI*.

Xiao Liu, Kaixuan Ji, Yicheng Fu, Weng Lam Tam, Zhengxiao Du, Zhilin Yang, and Jie Tang. 2021. P-tuning v2: Prompt tuning can be comparable to fine-tuning universally across scales and tasks. *arXiv preprint arXiv:2110.07602*.

Justin Lovelace and Carolyn Rose. 2022. A framework for adapting pre-trained language models to knowledge graph completion. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 5937–5955.

George A Miller. 1995. Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41.

Deepak Nathani, Jatin Chauhan, Charu Sharma, and Manohar Kaul. 2019. Learning attention-based embeddings for relation prediction in knowledge graphs. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4710–4723.

Dai Quoc Nguyen, Dat Quoc Nguyen, Tu Dinh Nguyen, and Dinh Phung. 2018. A convolutional neural network-based model for knowledge base completion and its application to search personalization. *Semantic Web*.

OpenAI. 2023. Gpt-4 technical report.

Apoorv Saxena, Adrian Kochsiek, and Rainer Gemulla. 2022. Sequence-to-sequence knowledge graph completion and question answering. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2814–2828.

Richard Socher, Danqi Chen, Christopher D Manning, and Andrew Ng. 2013. Reasoning with neural tensor networks for knowledge base completion. In *NIPS*, pages 926–934.

Fabian M Suchanek, Gjergji Kasneci, and Gerhard Weikum. 2007. Yago: a core of semantic knowledge. In *WWW*, pages 697–706. ACM.

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*.

Bo Wang, Tao Shen, Guodong Long, Tianyi Zhou, Ying Wang, and Yi Chang. 2021. Structure-augmented text representation learning for efficient knowledge graph completion. In *Proceedings of the Web Conference 2021*, pages 1737–1748.

Haoyu Wang, Vivek Kulkarni, and William Yang Wang. 2018. Dolores: Deep contextualized knowledge graph embeddings. *arXiv preprint arXiv:1811.00147*.

Liang Wang, Wei Zhao, Zhuoyu Wei, and Jingming Liu. 2022. Simkgc: Simple contrastive knowledge graph completion with pre-trained language models. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4281–4294.

Quan Wang, Zhendong Mao, Bin Wang, and Li Guo. 2017. Knowledge graph embedding: A survey of approaches and applications. *IEEE TKDE*, 29(12):2724–2743.

Zhen Wang, Jianwen Zhang, Jianlin Feng, and Zheng Chen. 2014a. Knowledge graph and text jointly embedding. In *EMNLP*.

Zhen Wang, Jianwen Zhang, Jianlin Feng, and Zheng Chen. 2014b. Knowledge graph embedding by translating on hyperplanes. In *AAAI*.

Zhigang Wang and Juan-Zi Li. 2016. Text-enhanced representation learning for knowledge graph. In *IJCAI*, pages 1293–1299.

Han Xiao, Minlie Huang, Lian Meng, and Xiaoyan Zhu. 2017. SSP: semantic space projection for knowledge graph embedding with text descriptions. In *AAAI*.

Han Xiao, Minlie Huang, and Xiaoyan Zhu. 2016. TransG: A generative model for knowledge graph embedding. In *ACL*, volume 1, pages 2316–2325.

Ruobing Xie, Zhiyuan Liu, Jia Jia, Huanbo Luan, and Maosong Sun. 2016. Representation learning of knowledge graphs with entity descriptions. In *AAAI*.

Xin Xie, Zhoubo Li, Xiaohan Wang, Yuqi Zhu, Ningyu Zhang, Jintian Zhang, Siyuan Cheng, Bozhong Tian, Shumin Deng, Feiyu Xiong, and Huajun Chen. 2023. Lambdakg: A library for pre-trained language model-based knowledge graph embeddings.

Jiacheng Xu, Xipeng Qiu, Kan Chen, and Xuanjing Huang. 2017. Knowledge graph representation with jointly structural and textual encoding. In *IJCAI*, pages 1318–1324.

Bishan Yang, Wen-tau Yih, Xiaodong He, Jianfeng Gao, and Li Deng. 2015. Embedding entities and relations for learning and inference in knowledge bases. In *ICLR*.

Liang Yao, Chengsheng Mao, and Yuan Luo. 2019. Kg-bert: Bert for knowledge graph completion. *arXiv preprint arXiv:1909.03193*.

Jason Youn and Ilias Tagkopoulos. 2023. KGLM: Integrating knowledge graph structure in language models for link prediction. In *Proceedings of the 12th Joint Conference on Lexical and Computational Semantics (*SEM 2023)*, pages 217–224, Toronto, Canada. Association for Computational Linguistics.

Fuzheng Zhang, Nicholas Jing Yuan, Defu Lian, Xing Xie, and Wei-Ying Ma. 2016. Collaborative knowledge base embedding for recommender systems. In *KDD*, pages 353–362. ACM.

Zhao Zhang, Fuzhen Zhuang, Meng Qu, Fen Lin, and Qing He. 2018. Knowledge graph embedding with hierarchical relation structure. In *EMNLP*, pages 3198–3207.

Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, et al. 2023. A survey of large language models. *arXiv preprint arXiv:2303.18223*.

Yuqi Zhu, Xiaohan Wang, Jing Chen, Shuofei Qiao, Yixin Ou, Yunzhi Yao, Shumin Deng, Huajun Chen, and Ningyu Zhang. 2023. Llms for knowledge graph construction and reasoning: Recent capabilities and future opportunities.

A Example Input

An example input for LLM relation prediction from YAGO3-10: "What is the relationship between Sergio Padt and Jong Ajax? Please choose your answer from: is known for|is citizen of|has currency|has child|deals with|has academic advisor|has gender|wrote music for|acted in|died in|has capital|works at|lives in|is affiliated to|has musical role|is located in|happened in|has official language|created|has won prize|influences|is politician of|is connected to|owns|graduated from|was born in|is leader of|exports|is interested in|participated in|directed|imports|edited|has neighbor|has website|is married to|plays for."

Two example inputs for LLM entity prediction with structural information from YAGO3-10 are: "Giving the neighbors of Joseph Bologna: Transylvania 6-5000 (1985 film)|Boynton Beach Club|Emmy Award|male|Sins (TV miniseries). Complete the fact: Joseph Bologna acted in" and "What/Who/When/Where/Why is affiliated to Arsenal F.C.? The neighbors of Arsenal F.C. :Darragh Ryan|Leslie Jones (footballer)|Andrew Devine|Gilles Grimandi|Ray Kennedy.".

Input	Label	Output	
Is this true: Everett T Moore profession Librarian?	True	GPT-4	I cannot verify specific personal information about individuals who are not public figures.
		ChatGPT	I'm sorry, but I don't have enough information to confirm whether Everett T Moore is a librarian or not.
		LLaMA-7B	I was wondering if anyone could tell me if this is true.
		LLaMA-13B	I'm a librarian at Everett T Moore Middle School in San Antonio, TX.
		KG-LLaMA-7B	Yes, this is true.
		KG-LLaMA-13B	Yes, this is true.
		ChatGLM-6B	Yes, Everett T Moore is a profession Librarian.
		KG-ChatGLM-6B	Yes, this is true.
Josip Škorić has gender	male	GPT-4	Josip Škorić is a male name. Josip is a Croatian form of the name Joseph, which is a masculine name.
		ChatGPT	Josip Škorić is a human being and therefore has a gender. However, I couldn't find any information about the gender of Josip Škorić.
		LLaMA-7B	Josip Škorić has gender issues. I'm a 22-year-old guy from Croatia, who has gender issues.
		LLaMA-13B	Josip Škorić has undergone gender reassignment surgery.
		KG-LLaMA-7B	male
		KG-LLaMA-13B	male
		ChatGLM-6B	As an AI language model, I do not have access to information about specific individuals' genders or their personal identity, including JosipŠkorić's.
		KG-ChatGLM-6B	male

Table 6: Examples outputs from different LLMs. The first line is taken from FB13-100 and the second line is from YAGO3-10-100.