Project Proposal and Literature Review

Alexandre Domingues Andrade
alexandrade@student.dei.uc.pt

1 Literature Review

1.1 Knowledge-Augmented Language Model Prompting for Zero-Shot Knowledge Graph Question Answering

The paper *Knowledge-Augmented Language Model Prompting for Zero-Shot Knowledge Graph Question Answering*[BAS23] introduces a new methodology to input prompting enriched with knowledge graphs, named KAPING. Because this solution augments only the input, it does not require model training, thus completely zero-shot. Because it does not require training, any model can use it, close or open source, without pre-training or training the model for a specific task and dataset or even be used to enrich black-box algorithms hidden behind APIs (like ChatGPT).

This paper's work was done in the context of a question-answering task, where questions are prompted to the LLM, and an answer is generated. This input enrichment method should be used in any question-answering bot or chat because the knowledge graphs have more up-to-date and accurate data than the LLMs, which represent a snapshot in time.

The main steps described in the paper are as follows: the entities are extracted from the input. Then, the extracted entities are matched with the entities of the Knowledge Graph, and the triplets are extracted. Triples are transformed into text, and the sequence ranking is done to extract the k more relevant sentences to enrich the input. Finally, the triples and the question are fed to the LLM, and an answer is returned.

The framework was compared with multiple methods. No knowledge; that is just an answer generator without input enrichment. Randomly extracted knowledge, the input is enriched with random knowledge. Popular knowledge extraction, the triples are ranked by their popularity. The last method is Generated knowledge, which is extracted from other LLMs. The framework outperforms all LM prompting methods on zero-shot in Knowledge Graph Question Answering. In one of the experiments, the results using entities labelled in the dataset were compared with results when using the Entity Linking model ReFinED to extract the entities. As expected, the results with the entity linking model were worse than the labelled entities because of the model performance. The framework outperformed the other baseline methods by up to 48% on average across multiple large language models of various sizes.

One limitation is the extraction of knowledge from multi-hop neighbourhoods. As stated in the paper, many times, the answer to the questions is outside the first-order neighbour, and it is necessary to retrieve 2-hop triples. The problem is that the quantity of triples grows exponentially when the order is raised; consequently, better triples ranking is necessary.

1.2 Knowledge-Augmented Language Model Verification

Despite the results obtained when prompting the LLMs with the KAPING, the model often does not generate the answer with the knowledge extracted, or the knowledge extracted does not reflect the truth; this is a general problem in knowledge augmentation approaches. In [Bae+23] from one of the authors, these problems are identified and described. The paper suggests an answer validator to resolve the problem of not using the extracted knowledge. The validator will evaluate if the answer generated by the model is correct or not. The method is named Knowledge Augmented Language Model Verification (KALMV).

Other fact-checking works are mentioned in the paper, but the differences are as follows: KALMV verifies the relevance of the retrieved knowledge before incorporating it into language models. In contrast, other methods assume the retrieved knowledge is always pertinent, which is not always true. It can detect when language models ignore the given knowledge and hallucinate answers are not grounded in the retrieved information; most other approaches assume the retrieved knowledge is accurately reflected in generated answers. It may only provide answers if validated as correct; the other fact-checking methods generally always provide an answer and refinements. Verifies both the knowledge retrieval and answer generation steps, and many existing works focus only on checking the factuality of the final generated answer.

The process of verification is done with a small and instruction-finetuned LM. The input of the verifier is the triplet of input question, retrieved knowledge, and generated answer; the output is A. the retrieved knowledge is not helpful to answer the question; B. the generated answer is not grounded in the retrieved knowledge; C. all the other cases. Two types of errors exist. Thus, each one has a different strategy. If a retrieval error is detected (i.e. the retrieved knowledge is irrelevant to the query), KALMV retrieves new knowledge from the external source. If a grounding error is

detected (i.e. the generated answer is not faithfully grounded in the retrieved knowledge), KALMV generates a new answer using the language model. The answer generation process is repeated until relevant knowledge is retrieved and correctly incorporated into the answer.

In Open-domain QA and Knowledge graph QA benchmarks, the KALMV significantly improves the performance of knowledge-augmented LMs on all datasets across different LM sizes. This framework, allied with KAPING or other state-of-the-art knowledge-injected LLMs, demonstrates a good improvement from previous methods to reduce hallucination and increase the factual quality of the generated answers.

2 Project Proposal

2.1 Description

End users' use of large language models (LLMs) has significantly increased since OpenAI announced the ChatGPT. The amount of trust users put in the models can lead to the spread of misinformation because the models' hallucinations mislead the users.

This project aims to improve the factual data provided by the LLMs in their responses, reduce the hallucination, and provide sources through knowledge injection from Knowledge Graphs such as Wikidata.

2.2 Goals

Knowledge must be injected somewhere in the neural network to improve the factual quality of the generated text. This project will inject knowledge into the input to be model agnostic. Performance and efficiency-wise, it is pretended that we should spend as few resources as possible. Consequently, the priority is to use zero-shot methods.

2.3 Approach

The input prompt engineering inspires the work done on [BAS23]; this project aims to improve the knowledge extraction technique from the Knowledge Graphs used in the paper, with a focus on one of its limitations, multi-hop data extraction.

Before feeding the user question to the PLM, this approach prompts the PLM with contextual information extracted from Knowledge Graphs. It is necessary to identify the entities in the sentence. The entities are extracted from the question using BLINK [Wu+20]. BLINK is an entity-linking algorithm that uses Wikipedia as the knowledge base (entity library). The returned entities have an ID and a description; this identification will be used to extract triplets from knowledge graphs and save them with the question. In the paper, one limitation was the knowledge extraction with multi-hop neighbours and discarding irrelevant facts.

Then, it is necessary to verbalise the extracted triplets. The paper used linear verbalisation; the same approach will be used because the results were good. The triplet extraction leads to an exaggerated amount of context, which is most irrelevant to the question, so ranking the information is necessary. One way of accomplishing that is by measuring the relevance of the sentences. BM25 [RZ09] appeared with groundbreaking results, and since then, multiple works have been done in this area; the RocketQA [Ren+21] dense retriever based on a pre-trained language model (PLM) will be used because of its achieved state-of-art results when compared to other methods.

At last, the k top sentences are prompted to the model and the question is fed.

2.4 Benchmarking

In order to evaluate the quality of the solutions, they will be benchmarked in Natural Questions Benchmark [Kwi+19].

2.5 Checkpoints

1. November 5
2. November 26

2.6 Data and Tools

The Natural Questions dataset will be used. The work will be done with Python and the PyTorch framework.

3 Ethics

In this document writing, a grammar corrector helped correct the text, and generative AI was used to rephrase sentences and summarise information.

References

[Bae+23] Jinheon Baek et al. *Knowledge-Augmented Language Model Verification*. 2023. arXiv: 2310.12836 [cs.CL]. URL: https://arxiv.org/abs/2310.12836.

[BAS23] Jinheon Baek, Alham Fikri Aji, and Amir Saffari. *Knowledge-Augmented Language Model Prompting for Zero-Shot Knowledge Graph Question Answering*. 2023. arXiv: 2306.04136 [cs.CL]. URL: https://arxiv.org/abs/2306.04136.

[Kwi+19] Tom Kwiatkowski et al. "Natural Questions: a Benchmark for Question Answering Research". In: *Transactions of the Association of Computational Linguistics* (2019).

[Ren+21] Ruiyang Ren et al. "RocketQAv2: A Joint Training Method for Dense Passage Retrieval and Passage Re-ranking". In: *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*. Ed. by Marie-Francine Moens et al. Online and Punta Cana, Dominican Republic: Association for Computational Linguistics, Nov. 2021, pp. 2825–2835. DOI: 10.18653/v1/2021.emnlp-main.224. URL: https://aclanthology.org/2021.emnlp-main.224.

[RZ09] Stephen Robertson and Hugo Zaragoza. "The Probabilistic Relevance Framework: BM25 and Beyond". In: *Foundations and Trends® in Information Retrieval* 3.4 (2009), pp. 333–389. ISSN: 1554-0669. DOI: 10.1561/1500000019. URL: http://dx.doi.org/10.1561/1500000019.

[Wu+20] Ledell Wu et al. "Zero-shot Entity Linking with Dense Entity Retrieval". In: *EMNLP*. 2020.