

Cover Letter

We thank the reviewers for their engagement with our work, and appreciate the constructive feedback provided. All suggestions have been reflected as revisions in the manuscript. This cover letter addresses the revisions we made and how they relate to the reviewers' comments.

Reviewer 1

Comment #1: The paper does not highlight the educational impact of this work which is of importance to the workshop.

As LLM-based educational technologies continue to be developed, it is important that they are grounded to factual content to ensure they are explainable and reliable. With retrieval-augmented generation, for example, factual sources can be injected into LLMs' responses, and improve their reasoning skills. Our framework provides a cost-effective solution for quickly generating knowledge graphs, which, as logical structures, natively support a variety of tasks that are useful in educational contexts, including question answering and reasoning. To this end, the conversational and generative skills of LLMs can be grounded with the factual and explainable properties of KGs to create better educational technologies.

Comment #2: LLMs are prone to hallucination. If we provide longer and more complex sentences, how would this system perform?

In our experiments, hallucinations typically take the form of mislabeling KG triplets, or providing otherwise irrelevant triplets. By including the proposed clause deconstruction step, we noticed an improvement in the quality of knowledge graph components. We believe this is due to the fact that the simplification significantly reduces the complexity of the input sentences. Whereas complex sentences have lengthy dependencies between ideas, the short sentences we generate clearly distinguish the relationships in the text. With clear relationships, the LLM seems to easily identify the components of the knowledge graph. On our poster, we will include a comparative evaluation that shows the quality of knowledge graphs is higher when using our proposed pipeline than when using other knowledge graph construction pipelines.

Comment #3: How would the proposed system perform for incorporating new knowledge- and how would it handle conflicting relationships between entities?

Our proposed system is a batch-wise processing pipeline; a new, separate graph is created for each input. Since new knowledge can only be added as a new input, a new graph would need to be created. However, the union of these graphs would represent the combined knowledge that they comprise. Since our knowledge graphs are represented in CSV format, graphs can be combined by concatenating their files.

We have left the investigation of methods for quality assurance, including conflict resolution for future work. There are various approaches for automated quality assurance on knowledge

graphs, including committee-based methods, and graph-based methods. We plan to provide a comparative evaluation of the effectiveness of each method on the knowledge graphs our framework generates.

Comment #4: Related work regarding using LLMs for KG construction does not seem to be highlighted.

Addressing the comment, we would like to emphasize that several recent works have indeed explored the utilization of LLMs in KG construction tasks. Specifically, Meyer et al. (2023) investigated the capabilities of ChatGPT in various knowledge graph engineering tasks, such as knowledge extraction from PDFs, SPARQL query generation, and knowledge graph exploration and visualization. Furthermore, works by Hu et al. (2023) and Carta et al. (2023) have proposed pipelines for generating knowledge graphs from textual data, employing methodologies that include entity extraction followed by relation extraction. To provide a more substantial motivation of our own work and its contributions, we have updated our manuscript to include these works.

Reviewer 2

Comment #1: Necessity of clause deconstruction step is not clear from the paper. Same output may be generated using only the prompt in the knowledge graph extraction section. If there is any performance gain with this option, it should have been mentioned in the paper.

While we acknowledge that the Knowledge Graph Extraction prompt can similarly generate triplets from a given text, omitting the Clause Deconstruction step misses out on its key advantages. The primary purpose of clause deconstruction is to simplify complex sentences into their individual substructures. By teasing out each fact of a sentence before converting to knowledge graph syntax, we ensure that all facts are accurately represented. We have updated the manuscript to discuss the importance of this simplification step. Additionally, our poster will showcase the performance gain that comes from implementing this step.

Comment #2: In the demonstration there is a option of sentence batch size. Effect of sentence batch size is not mentioned.

Since the initial version of the paper, we have updated our text batching algorithm to be based on the number of words rather than the number of sentences. However, the purpose and effect of batching is still the same: the batch size specifies how much text to process in each forward pass of our system. It can be seen as controlling the "granularity" of the processing we propose. For example, processing sentence-by-sentence may ensure that each sentence is given equal attention, but the relationship between sentences will be lost. On the other hand, larger batches may better capture lengthy relationships across sentences, but may come at the cost of certain facts if the LLM decides to only pick out the most important ones.

In the graphical demonstration, this parameter is omitted altogether. However, using the command line version of our tool, the batch size, as well as other pipeline and LLM

configuration parameters can be fully customized. This decision was made to keep the graphical user interface intuitive, while allowing deeper analysis and experimentation.

Comment #3: There is no conclusion/discussion section.

We acknowledge the reviewer's suggestion to include a summative discussion of our work. Our manuscript has been updated accordingly.