From Words to Wisdom: Automatically Generating Knowledge Graphs for Interpretable Educational AI

Johaun Hatchett[1,2], Debshila Basu Mallick[3], Richard G. Baraniuk[1,3]

[1]Department of Electrical and Computer Engineering, Rice University, USA
[2]Applied Physics Graduate Program, Smalley-Curl Institute, Rice University, USA
[3]OpenStax, Rice University, USA
jh152@rice.edu, db19@rice.edu, richb@rice.edu

Abstract

Large language models (LLMs) have emerged as powerful tools with vast potential across various domains. While they have the potential to transform the educational landscape with personalized learning experiences, these models face challenges such as high training and usage costs, and susceptibility to inaccuracies. One promising solution to these challenges lies in leveraging knowledge graphs (KGs) for knowledge injection. By integrating factual content into pre-trained LLMs, KGs can reduce the costs associated with domain alignment, mitigate the risk of hallucination, and enhance the interpretability of the models' outputs. To meet the need for efficient knowledge graph creation, we introduce *Words to Wisdom* (W2W), a domain-independent LLM-based tool that automatically generates KGs from plain text. With W2W, we aim to provide a streamlined KG construction option that can drive advancements in grounded LLM-based educational technologies.

Introduction

The recent emergence of *large language models* (LLMs) has introduced a new era of AI capabilities, including an unprecedented proficiency in complex language-based tasks (Brown et al. 2020; He et al. 2023). In the realm of education, LLMs have emerged as powerful tools, offering promises of personalized learning experiences, real-time feedback mechanisms, and automated assessment generation (Wang et al. 2022; Sonkar et al. 2023; Seßler et al. 2023). However, despite these models' remarkable performance, LLMs still pose significant challenges. First, the cost of aligning pre-trained LLMs to specific tasks and domains is significant. For those without the proper infrastructure or funds, state-of-the-art technologies still remain inaccessible. Another notable concern is LLMs' susceptibility to hallucination – a phenomenon where a model generates seemingly valid, but ultimately fictitious content (Rawte, Sheth, and Das 2023). In educational contexts, where accessibility, factual integrity, and explainability are paramount, these challenges can undermine the trust and utility of LLM-based systems.

Knowledge graphs (KGs) – structured representations of knowledge consisting of entities (nodes) and relations (directed edges) – offer a promising avenue for addressing the shortcomings of LLMs. KGs represent facts as triplets (s, r, o), where r denotes the relation between the subject entity s and object entity o. As logical structures, KGs facilitate many essential educational tasks including question answering (Huang et al. 2019) and inference (Sonkar, Katiyar, and Baraniuk 2022). With *retrieval-augmented generation* (Lewis et al. 2021), KGs can be used to ground even pretrained LLMs to factual sources, improving their accuracy and interpretability, while mitigating the risk of hallucination (Wu et al. 2023; Feng, Zhang, and Fei 2023). Constructing a KG, however, has traditionally been a labor-intensive process, requiring domain-expertise and/or crowdsourcing efforts (Clancy, Ilyas, and Lin 2019; Chaudhri et al. 2021). While a few recent works have explored LLM-based KG construction methods to address these shortcomings (Meyer et al. 2023; Carta et al. 2023; Hu et al. 2023), building accurate KGs at scale still remains a challenge.

In this paper, we introduce *Words to Wisdom* (W2W), an innovative pipeline leveraging OpenAI's GPT-3.5 for automatic KG construction from unstructured text. Unlike other LLM-based KG construction methods, W2W does not rely on highly-formatted schema or pre-defined ontologies, making its outputs accessible and intuitive even for those with minimal understanding of database engineering. By harnessing the synergies between LLMs and KGs, W2W aims to offer a cost-effective solution for KG construction that can help advance the frontier of AI-driven educational technologies.

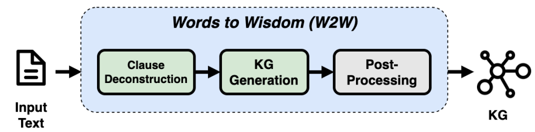

Figure 1: The W2W pipeline. Steps shown in green are processing steps using GPT-3.5. "Clause Deconstruction" is our novel contribution to existing automated knowledge graph (KG) construction frameworks.

System Overview

W2W is developed in Python 3.10, primarily using the `langchain` package for the LLM prompting modules and the `gradio` package for public web-hosting on Hugging Face Spaces.[1] The overall architecture is shown in Figure 1. Our system uses the `gpt-3.5-turbo` model checkpoint for text processing, with all model parameters at their default values. In the next paragraphs, we provide a description of each module of W2W.

Clause Deconstruction. For the initial module of W2W, we propose a novel text simplification step that breaks down complex sentences into distinct "units" of information. In natural language, a sentence comprises one or more clauses, each conveying an independent unit of information. Parsing the meaning of a sentence requires the ability to deconstruct these units. For instance, the complex sentence,

"Both dogs and cats chase squirrels, but fish do not,"

conveys three separate ideas: (i) *dogs chase squirrels*, (ii) *cats chase squirrels*, and (iii) *fish do not chase squirrels*. With our proposed simplification module, Clause Deconstruction, we prompt[2] GPT-3.5 to tease out the individual ideas from any sentence. The simplification ensures that our pipeline can handle sentences of varying lengths and complexities, and ensures that all essential information from the original text is explicitly- and concisely-written in the simplified text.

This unit-based deconstruction offers two crucial benefits. First, we note that sentences are simplified to a format that closely aligns with the representation of simple facts in KGs. Just as a KG triplet is composed of a subject entity s, a relation r, and an object entity o, so too is a clause comprised of a subject (the primary noun phrase), a relation (the linking verb phrase), and an object (the secondary noun phrase). This analogy suggests that our novel clause deconstruction step is well-suited to the automated KG extraction process.

The second benefit of Clause Deconstruction lies in its ability to mitigate hallucinations during the automatic KG construction process. By breaking down sentences into clearer, more concise substructures, our method helps to constrain spurious facts in the generated KG. This not only improves the overall quality of W2W's graphs, but also enhances their interpretability and reliability.

Knowledge Graph Extraction. W2W's core lies in its unique perspective on KG extraction. In our perspective, a KG triplet is nothing more than a specially-structured independent clause (e.g. (*dogs, chase, squirrels*)). Since the Clause Deconstruction step aligns the text with KG structure, we only need a simple prompt,[2] to translate from natural language syntax to KG syntax.

Our prompting strategy represents a departure from traditional KG construction approaches, which assume a graph-building perspective. In the traditional perspective, *entity extraction* defines the nodes of the graph, and *relationship selection* defines the edges of the graph (Clancy, Ilyas, and Lin 2019; Chaudhri et al. 2021). An advantage of our approach over the traditional approach is that we simplify these two tasks into a single process. By aligning our framework with the structure of natural language rather than the structure of graphs, we streamline the conversion between the two, allowing for a more intuitive and efficient KG extraction.

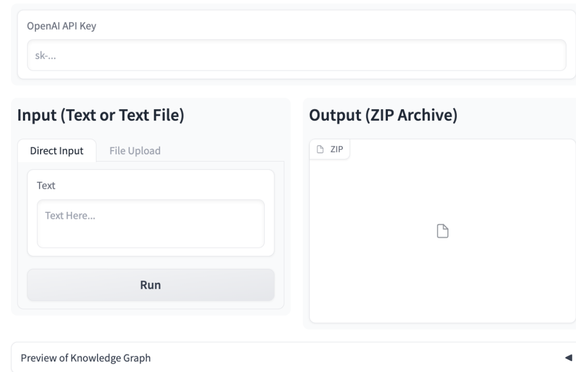

Figure 2: The W2W interface. At the top, we input an OpenAI API key. On the left, we provide plain text to be processed. After processing, a downloadable ZIP archive containing the generated knowledge graph and other metadata appears on the right. In the bottom-most panel, we provide a preview of the knowledge graph facts.

Post-Processing. W2W is a batch-wise processing pipeline, which introduces the possibility of undesired variations in capitalization, verb conjugation, and noun inflection across the generated KG elements. To limit these variations and improve the overall consistency of the graph, we implement a basic normalization: (i) lowercase all entities and relations; (ii) remove leading articles and "to be" verbs from entities and relations.

Demonstration

W2W's graphical user interface is shown in Figure 2. There are four primary panels:

- **OpenAI API Key:** An OpenAI API key is required to run W2W.
- **Input:** The user can either: (a) type text directly, or (b) upload a plain text document for processing.
- **Output:** After processing, W2W exports a ZIP archive containing: (i) the generated knowledge graph, (ii) the indexed text batches, and (iii) pipeline metadata.
- **Preview:** After processing, the generated knowledge graph can be inspected by expanding the accordion panel.

We also provide W2W as a command-line utility tool, allowing for customization of the pipeline, the LLM, and its prompts.

Conclusion

W2W offers a unique solution for quickly generating knowledge graphs that can be used to ground large language models. As future work, we will explore methods for quality assurance that further improve our graphs' consistency.

[1] https://huggingface.co/spaces/jhatchett/Words2Wisdom

[2] Prompts used in this work are available in our source code: https://github.com/johaun-hatchett/Words2Wisdom

References

Brown, T. B.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; Agarwal, S.; Herbert-Voss, A.; Krueger, G.; Henighan, T.; Child, R.; Ramesh, A.; Ziegler, D. M.; Wu, J.; Winter, C.; Hesse, C.; Chen, M.; Sigler, E.; Litwin, M.; Gray, S.; Chess, B.; Clark, J.; Berner, C.; McCandlish, S.; Radford, A.; Sutskever, I.; and Amodei, D. 2020. Language Models are Few-Shot Learners. arXiv:2005.14165.

Carta, S.; Giuliani, A.; Piano, L.; Podda, A. S.; Pompianu, L.; and Tiddia, S. G. 2023. Iterative Zero-Shot LLM Prompting for Knowledge Graph Construction. arXiv:2307.01128.

Chaudhri, V. K.; Boggess, M.; Aung, H. L.; Mallick, D. B.; Waters, A. C.; and Baraniuk, R. 2021. A Case Study in Bootstrapping Ontology Graphs from Textbooks. In *3rd Conference on Automated Knowledge Base Construction*.

Clancy, R.; Ilyas, I. F.; and Lin, J. 2019. Scalable Knowledge Graph Construction from Text Collections. In Thorne, J.; Vlachos, A.; Cocarascu, O.; Christodoulopoulos, C.; and Mittal, A., eds., *Proceedings of the Second Workshop on Fact Extraction and VERification (FEVER)*, 39–46. Hong Kong, China: Association for Computational Linguistics.

Feng, C.; Zhang, X.; and Fei, Z. 2023. Knowledge Solver: Teaching LLMs to Search for Domain Knowledge from Knowledge Graphs. arXiv:2309.03118.

He, Q.; Zeng, J.; Huang, W.; Chen, L.; Xiao, J.; He, Q.; Zhou, X.; Chen, L.; Wang, X.; Huang, Y.; Ye, H.; Li, Z.; Chen, S.; Zhang, Y.; Gu, Z.; Liang, J.; and Xiao, Y. 2023. Can Large Language Models Understand Real-World Complex Instructions? arXiv:2309.09150.

Hu, Y.; Zou, F.; Han, J.; Sun, X.; and Wang, Y. 2023. Llm-Tikg: Threat Intelligence Knowledge Graph Construction Utilizing Large Language Model.

Huang, X.; Zhang, J.; Li, D.; and Li, P. 2019. Knowledge Graph Embedding Based Question Answering. In *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining*, WSDM '19, 105–113. New York, NY, USA: Association for Computing Machinery. ISBN 9781450359405.

Lewis, P.; Perez, E.; Piktus, A.; Petroni, F.; Karpukhin, V.; Goyal, N.; Küttler, H.; Lewis, M.; tau Yih, W.; Rocktäschel, T.; Riedel, S.; and Kiela, D. 2021. Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks. arXiv:2005.11401.

Meyer, L.-P.; Stadler, C.; Frey, J.; Radtke, N.; Junghanns, K.; Meissner, R.; Dziwis, G.; Bulert, K.; and Martin, M. 2023. LLM-assisted Knowledge Graph Engineering: Experiments with ChatGPT. arXiv:2307.06917.

Rawte, V.; Sheth, A.; and Das, A. 2023. A Survey of Hallucination in Large Foundation Models. arXiv:2309.05922.

Seßler, K.; Xiang, T.; Bogenrieder, L.; and Kasneci, E. 2023. PEER: Empowering Writing with Large Language Models. In Viberg, O.; Jivet, I.; Muñoz-Merino, P. J.; Perifanou, M.; and Papathoma, T., eds., *Responsive and Sustainable Educational Futures*, 755–761. Springer Nature Switzerland. ISBN 978-3-031-42682-7.

Sonkar, S.; Katiyar, A.; and Baraniuk, R. 2022. NePTuNe: Neural Powered Tucker Network for Knowledge Graph Completion. In *Proceedings of the 10th International Joint Conference on Knowledge Graphs*, IJCKG '21, 177–180. New York, NY, USA: Association for Computing Machinery. ISBN 9781450395656.

Sonkar, S.; Liu, N.; Mallick, D. B.; and Baraniuk, R. G. 2023. CLASS: A Design Framework for building Intelligent Tutoring Systems based on Learning Science principles. arXiv:2305.13272.

Wang, Z.; Valdez, J.; Basu Mallick, D.; and Baraniuk, R. G. 2022. Towards Human-Like Educational Question Generation with Large Language Models. In Rodrigo, M. M.; Matsuda, N.; Cristea, A. I.; and Dimitrova, V., eds., *Artificial Intelligence in Education*, 153–166. Cham: Springer International Publishing. ISBN 978-3-031-11644-5.

Wu, Y.; Hu, N.; Bi, S.; Qi, G.; Ren, J.; Xie, A.; and Song, W. 2023. Retrieve-Rewrite-Answer: A KG-to-Text Enhanced LLMs Framework for Knowledge Graph Question Answering. arXiv:2309.11206.