Using Knowledge Graphs and Agentic LLMs for Factuality Text Assessment and Improvement

Linda Kwan, Pouya G. Omran and Kerry Taylor

Australian National University, Canberra ACT 2601, AU

Abstract

This paper addresses the challenge of assessing and enhancing the factual accuracy of texts generated by large language models (LLMs). Existing methods often rely on self-reflection or external knowledge sources, validating statements individually and rigidly, thus missing a holistic view. We propose a novel approach utilizing a comprehensive knowledge graph (KG), such as Wikidata, to assess and improve the factuality of generated texts. Our method dynamically retrieves and integrates relevant facts during the assessment process, providing a more interconnected and accurate evaluation. Integrating KG with LLM capabilities enhances the overall factual integrity, leading to more reliable AI-generated content. Our results demonstrate improvements in factual accuracy, highlighting the effectiveness of our approach.
Submission type: **Poster**

Keywords

Knowledge Graph, Large Language Model, Agentic LLM, LLM Evaluation

1. Introduction and Background

The rapid advancements in large language models (LLMs) have transformed natural language processing, enabling these models to understand and generate human-like text with remarkable accuracy. Despite their impressive capabilities, assessing and enhancing the factual accuracy of texts produced by LLMs remains a significant challenge. Texts generated by these models, including popular applications like ChatGPT, can contain inaccuracies and misinformation, posing risks to users who might accept generated content as factual without verification. This problem underscores the need for robust methods to assess and improve the factuality of texts produced by generative models. Existing approaches to address this issue rely on either LLMs' self-reflection or external knowledge sources like knowledge graphs (KGs)[1, 2, 3, 4, 5, 6, 7, 8, 9, 10]. Self-reflection is restrictive as it lacks an ultimate source of truth and depends on the LLMs' inherent knowledge. External knowledge methods validate individual statements, which limits their effectiveness due to a local perspective and rigid matching processes during the entity linking or predicate alignment phase.

To overcome these limitations, we propose a novel approach using a comprehensive knowledge graph, such as Wikidata [11], to assess and enhance the factual accuracy of texts generated by LLMs. The enhancer agent integrates a general-purpose KG with LLM capabilities for named entity recognition and fact extraction, using an LLM encoder and vectorization for soft

Posters, Demos, and Industry Tracks at ISWC 2024, November 13–15, 2024, Baltimore, USA
linda.h.kwan@outlook.com (L. Kwan); P.G.Omran@anu.edu.au (P. G. Omran); kerry.taylor@anu.edu.au (K. Taylor)

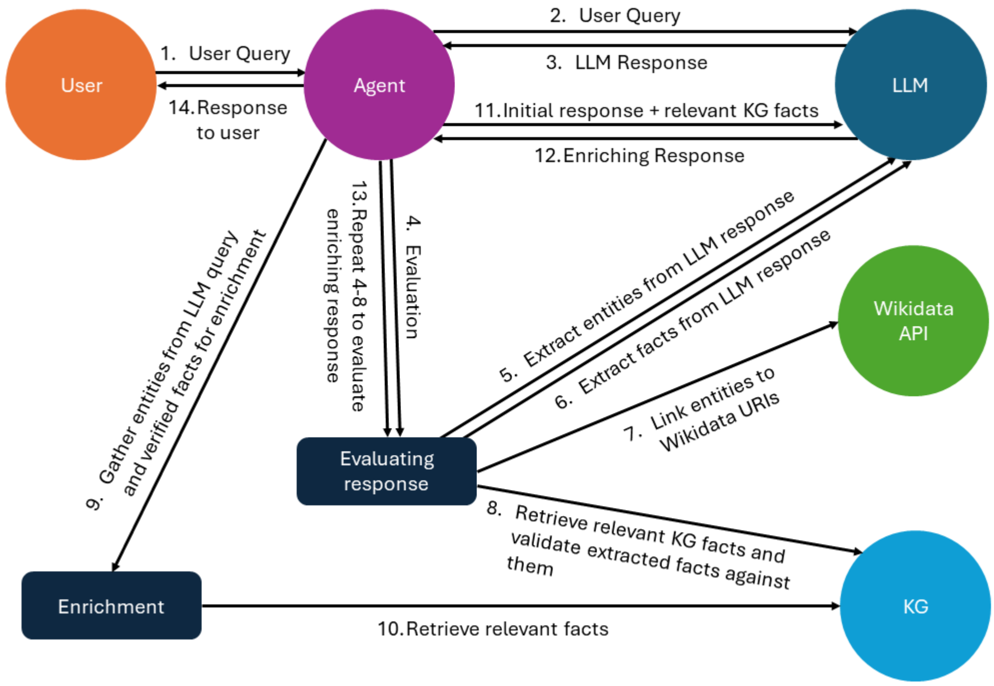

Figure 1: The overall architecture of our Factum Graph Enhancer.

alignment between text and KG facts, and then uses the relevant extracted facts to improve the original text. This process improves the factual accuracy of texts, including human and LLM-generated texts, ensuring they align with verified facts and real-world knowledge. The key contributions are: i. Integration of a general-purpose KG with LLMs for enhanced fact-checking. ii. Use of soft matching mechanisms based on LLM encoder for better text and KG fact alignment. iii. Development of a dynamic and iterative process for continuous text assessment and improvement.

2. Factum Graph Enhancer

Our proposed system, Factum Graph Enhancer (Fig. 1), begins its process after the initial response to a user query is generated by the Gemma LLM [12]. While the generation of the initial response is not part of our system, our diagram starts from this point. The initial response is assessed against a knowledge graph, such as Wikidata, through entity recognition and relation extraction to identify relevant entities and link them to their corresponding URIs. Using embedding similarity, we approximate the factual accuracy of these extracted facts by comparing them with triples retrieved from the KG. Following the assessment, relevant facts from the KG are selected and used to enrich the original response. This enriched response is then returned to the user, ensuring improved factual accuracy and relevance through evaluation and enhancement stages. Fig. 2 demonstrates an example of how an LLM-generated response

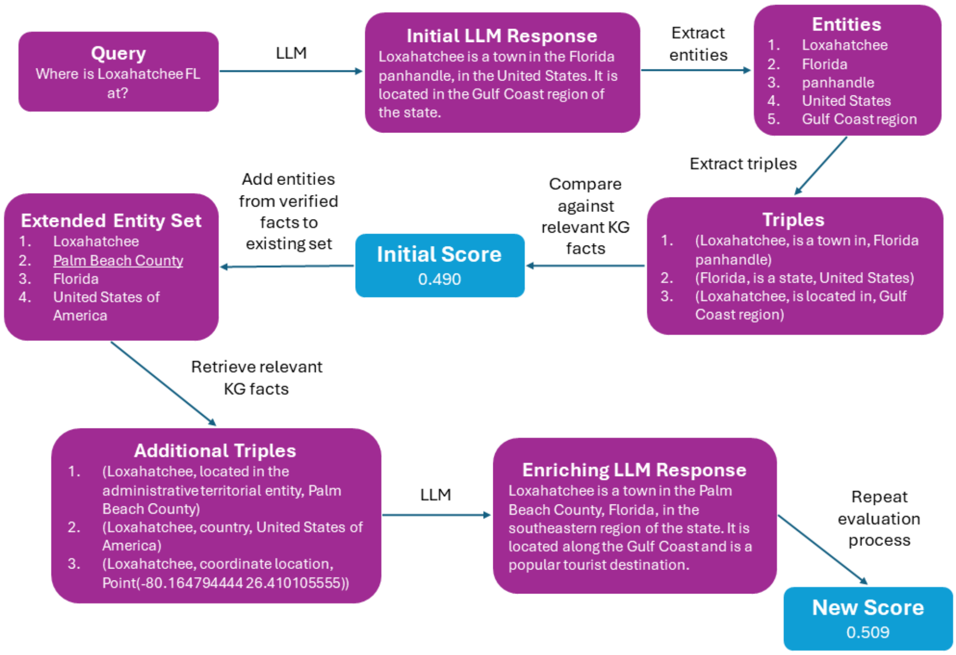

Figure 2: An example of how our Factum Graph Enhancer would enhance an LLM-generated response.

can be enriched.

Evaluating Text Based on the Knowledge Graph: We leverage Gemma LLM decoder [12] to extract entities from the initial response due to the zero-shot learning prompt. We use prompt engineering to extract entities from the text in the Python list format to achieve this. Using the extracted entities, we then leverage Gemma LLM to extract (subject, predicate, object) triples from the original response in markdown table format. We utilize the Wikidata REST API [13] to fetch the URI for each entity that appears as a subject in one of the extracted triples. Next, we use the Wikidata knowledge base to evaluate the degree of truthfulness of each fact extracted from the response by a soft matching mechanism. For each triple $(s_i, p_i, o_i) \in F$ extracted from the text, we perform a SPARQL query to retrieve triples with s_i in the subject position. We call this set $R(s_i)$. To reduce the complexity of selecting the most relevant triple, we first take the cosine similarity between the Sentence-BERT [14] embeddings of p_i and the predicate in each triple retrieved from the SPARQL query and select the three predicates from the retrieved set that produce the highest cosine similarity scores. We filter SPARQL results to the set of triples containing one of the top three predicates, then calculate the cosine similarity between the concatenated p_i and o_i string and the corresponding predicate and object string for each retrieved triple (s_i, p_r, o_r). We take the highest of those cosine similarity values as the evaluation score of the extracted triple and also take the corresponding retrieved fact to be used for correction. To calculate the Factuality Degree (FD) for the entire response, we sum the evaluation scores of the extracted triples and divide them by the number of facts extracted.

$$FD = \frac{\sum_{(s_i,p_i,o_i)\in F} \max_{(s_i,p_r,o_r)\in R(s_i)} \mathtt{sim}(\mathtt{emb}(\mathtt{concat}(p_i,o_i)), \mathtt{emb}(\mathtt{concat}(p_r,o_r)))}{|F|} \tag{1}$$

Enhancing the LLM output using LLM and relevant facts from the KG: We enhance the LLM output by considering entities involved in the selected facts from the previous section, which have the highest similarity, along with other linked entities in the text. We construct a set of linked entities from Wikidata and use SPARQL to retrieve all facts in Wikidata that have these entities as their subjects. This extensive set of retrieved facts forms our fetched KG, which we filter using a two-stage method. First, we consider all predicates in the fetched KG and find the similarity between each predicate and the vector representation of the entire original text, selecting the top n predicates. We then prune the KG to keep only the facts with these selected predicates. In the second stage, we calculate the cosine similarity between the vector representation of each fact (as a textual statement) and the original text, selecting the top m facts, in our experiment $n = 5$ and $m = 5$. Using cosine similarity to compare each retrieved KG fact and the original text ensures that less interesting KG facts would produce lower cosine similarity scores and get filtered out. Finally, we prompt our LLM decoder to use these selected facts to enrich the original text if they are relevant and helpful.

3. Experiments and Conclusion

A GitHub repository for this experiment is available[1]. For our experiments, we used a set of 35 questions from the WikiQA dataset [15]. We generated initial responses using Gemma LLM, assessed their factual accuracy, enriched the responses with relevant facts from a knowledge graph, and then reassessed them. The results showed an improvement in factual accuracy when using our Factum Graph Enhancer framework, with the FD score increasing from 0.260 with pure Gemma to 0.326 with Factum.

For example, in Fig. 2, the initial LLM response to the query *"Where is Loxahatchee FL at?"* includes the country (United States) and the region (Gulf Coast) Loxahatchee is located in. When the relevant KG facts were retrieved during the first evaluation stage, a relationship between Loxahatchee and Palm Beach County was identified. Hence, Palm Beach County was included in the extended entity set, allowing the triple *(Loxahatchee, located in the administrative territorial entity, Palm Beach County)* to be included in the enrichment set. The additional triples and the original response are then fed to the LLM to generate the enriching response. Consequently, the enriching LLM response includes the county Loxahatchee is located in (Palm Beach County), in addition to the existing information from the original response. This process demonstrates the effectiveness of our framework in enhancing the factual accuracy of LLM-generated text.

This enhancement underscores the potential of integrating knowledge graphs with LLMs to create more reliable and accurate AI-generated content. Future work will focus on expanding the dataset to include various questions and domains. Additionally, we plan to investigate the integration of other knowledge bases and vectorization methods, and the application of this framework in real-world scenarios to validate its effectiveness and scalability.

[1]https://github.com/lindakwan/factum-graph-enhancer

References

[1] P. Pezeshkpour, Measuring and modifying factual knowledge in large language models, in: 2023 International Conference on Machine Learning and Applications (ICMLA), IEEE, 2023, pp. 831–838.

[2] X.-Y. Fu, M. T. R. Laskar, C. Chen, S. B. TN, Are large language models reliable judges? a study on the factuality evaluation capabilities of llms, arXiv preprint arXiv:2311.00681 (2023).

[3] N. Lee, W. Ping, P. Xu, M. Patwary, P. N. Fung, M. Shoeybi, B. Catanzaro, Factuality enhanced language models for open-ended text generation, Advances in Neural Information Processing Systems 35 (2022) 34586–34599.

[4] F. F. Bayat, K. Qian, B. Han, Y. Sang, A. Belyi, S. Khorshidi, F. Wu, I. F. Ilyas, Y. Li, FLEEK: Factual Error Detection and Correction with Evidence Retrieved from External Knowledge, arXiv Preprint (2023). URL: https://platform.openai.com/docs/modelshttp://arxiv.org/abs/2310.17119. arXiv:2310.17119.

[5] C. Mavromatis, P. Karypis, G. Karypis, SemPool: Simple, robust, and interpretable KG pooling for enhancing language models, arXiv Preprint (2024). URL: https://arxiv.org/abs/2402.02289v1http://arxiv.org/abs/2402.02289. arXiv:2402.02289.

[6] E. C. Choi, E. Ferrara, FACT-GPT: Fact-Checking Augmentation via Claim Matching with LLMs, in: WebConf, volume 1, 2024. URL: https://arxiv.org/abs/2402.05904v1. doi:XXXXXXX.XXXXXXX. arXiv:2402.05904.

[7] Z. Yuan, A. Vlachos, Zero-Shot Fact-Checking with Semantic Triples and Knowledge Graphs, arXiv preprint (2023). arXiv:2312.11785v1.

[8] L. Pan, M. Saxon, W. Xu, D. Nathani, X. Wang, W. Y. Wang, Automatically Correcting Large Language Models: Surveying the landscape of diverse self-correction strategies, arXiv preprint (2023). arXiv:2308.03188v1.

[9] J. Liu, W. Wang, D. Wang, N. A. Smith, Y. Choi, H. Hajishirzi, P. G. Allen, VERA: A General-Purpose Plausibility Estimation Model for Commonsense Statements, arXiv preprint (2023). URL: https://huggingface.co/liujch1998/vera. arXiv:2305.03695v1.

[10] R. L. Logan, N. F. Liu, M. E. Peters, M. Gardner, S. Singh, Barack's Wife Hillary: Using Knowledge-Graphs for Fact-Aware Language Modeling, in: ACL, Association for Computational Linguistics (ACL), 2019, pp. 5962–5971. URL: https://arxiv.org/abs/1906.07241v2. doi:10.48550/arxiv.1906.07241. arXiv:1906.07241.

[11] D. Vrandečić, M. Krötzsch, Wikidata: a free collaborative knowledgebase, Commun. ACM 57 (2014) 78–85. URL: https://doi.org/10.1145/2629489. doi:10.1145/2629489.

[12] T. Mesnard, C. Hardin, R. Dadashi, et al., Gemma: Open models based on gemini research and technology, 2024. URL: https://arxiv.org/abs/2403.08295. arXiv:2403.08295.

[13] Wikidata:REST API - Wikidata, 2024. URL: https://www.wikidata.org/wiki/Wikidata:REST_API.

[14] N. Reimers, I. Gurevych, Sentence-bert: Sentence embeddings using siamese bert-networks, in: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, 2019. URL: https://arxiv.org/abs/1908.10084.

[15] Y. Yang, W.-t. Yih, C. Meek, WikiQA: A challenge dataset for open-domain question answering, in: L. Màrquez, C. Callison-Burch, J. Su (Eds.), Proceedings of the 2015 Conference

on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Lisbon, Portugal, 2015, pp. 2013–2018. URL: https://aclanthology.org/D15-1237. doi:10.18653/v1/D15-1237.