Matching Table Metadata to Knowledge Graphs: A Data Augmentation Perspective

Duo Yang[0009−0008−5942−3397], Ioannis Dasoulas[0000−0002−8803−1244], and Anastasia Dimou[0000−0003−2138−7972]

KU Leuven, Sint-Katelijne-Waver, Belgium
{duo.yang,ioannis.dasoulas,anastasia.dimou}@kuleuven.be

Abstract. Data augmentation is essential for matching table metadata, such as column names, to a knowledge graph without accessing the table's data or content. Previous works used large language models (LLMs) to enrich each column name into a single-sentence description but did not consider the entire table header. In this work, we propose a two-stage LLM-based process for column description generation, leveraging all available table metadata. The results highlight the importance of table headers for a broader context in data augmentation, with an 11–30% improvement of Hit@k in table metadata matching across two datasets.

Keywords: large language models · semantic linking · table metadata

1 Introduction

The rise of large language models (LLMs) has changed how people use them — treating them as knowledge sources that can be easily accessed through chat or natural language inputs (Petroni et al., 2019). LLMs are highly effective data generators for enriching column names and can contribute to the scenarios where limited table content is available for annotation systems to perform semantic linking between column names and a glossary (Vandemoortele et al., 2024). Previous work generated the column description directly without considering a broader context for the given column, such as the table header (Vandemoortele et al., 2024). Thus, we investigate how to leverage all available table metadata in data augmentation and conduct experiments with datasets[1] from the SemTab challenge (Lobo et al., 2023) to obtain preliminary insights.

2 Column Description Generation & Glossary Matching

Glossary matching is to select k properties from the glossary (source) that best describe the semantics of a given metadata M (target), such as a column name. Each glossary property includes a short description but the column does not. Therefore, it is crucial to generate column descriptions using all available table metadata. Our architecture includes two parts: 1) **a two-stage column**

[1] https://zenodo.org/records/14207376

description generation process. First, the table name and the entire table header (all the column names) are input into LLMs to generate a detailed table description. Second, the same inputs with generated table description, are used to generate a column description for the given column name; 2) **an embedding ranker for glossary matching**. The column description and all the glossary property descriptions are encoded into embeddings using the same LLM. By ranking the cosine similarities between the column embedding and property embeddings, the top-k properties are retrieved as predicted results.

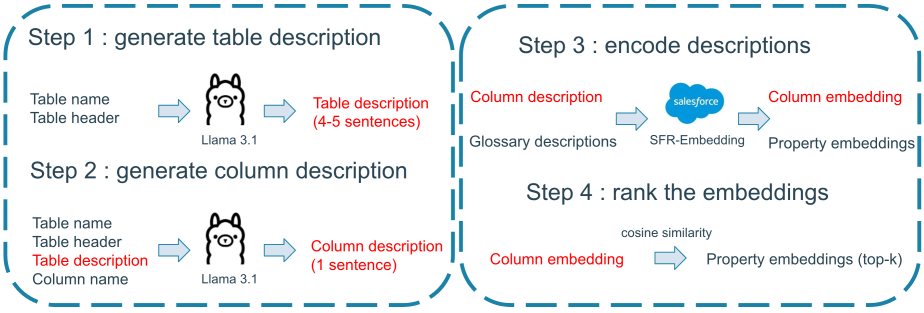

Fig. 1. Architecture: column description generation (left) & embedding ranking (right).

Table 1. Performance comparison of our approach on the Metadata2KG datasets.

Methods	Re-ranking	Dataset 1		Dataset 2	
		Hit@1	Hit@5	Hit@1	Hit@5
SOTA	no	**33%**	**52%**	**41%**	**72%**
(Vandemoortele et al., 2024)	yes	75%	92%	83%	98%
Ours	no	**47%**	**63%**	**71%**	**94%**

Table 1 represents the main results of glossary matching. Our approach demonstrates significant potential, achieving an 11–30% improvement in Hit@k across both datasets under the same conditions without re-ranking. Notably, in dataset 2, it reaches 94% in Hit@5, approaching state-of-the-art performance with re-ranking. This highlights that leveraging all available metadata is essential for generating better column descriptions compared to relying solely on the table name and the given column name, as the quality of generated column descriptions influences the performance of the glossary matching task. The Hit@k can be further improved by applying a LLM-based re-ranker (Sun et al., 2023; Vandemoortele et al., 2024), as our current approach embeds these descriptions directly using LLMs without more fine-grained re-ranking to refine the results.

Bibliography

Lobo, E. A., Hassanzadeh, O., Pham, N., Mihindukulasooriya, N., Subramanian, D., and Samulowitz, H. (2023). Matching table metadata with business glossaries using large language models. In *Proceedings of the 18th International Workshop on Ontology Matching co-located with the 22nd International Semantic Web Conference Athens, Greece, November 7, 2023*, volume 3591 of *CEUR Workshop Proceedings*, pages 25–36. CEUR-WS.org.

Petroni, F., Rocktäschel, T., Riedel, S., Lewis, P., Bakhtin, A., Wu, Y., and Miller, A. (2019). Language models as knowledge bases? In Inui, K., Jiang, J., Ng, V., and Wan, X., editors, *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2463–2473, Hong Kong, China. Association for Computational Linguistics.

Sun, W., Yan, L., Ma, X., Wang, S., Ren, P., Chen, Z., Yin, D., and Ren, Z. (2023). Is chatgpt good at search? investigating large language models as reranking agents. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Vandemoortele, N., Steenwinckel, B., Hoecke, S., and Ongenae, F. (2024). Scalable table-to-knowledge graph matching from metadata using llms.