Evaluating Ambiguous Questions in Text2SQL

Simone Papicchio[1,2][0009−0005−5361−0042], Luca Cagliero[1][0000−0002−7185−5247], and Paolo Papotti[2][0000−0003−0651−4128]

[1] Politecnico di Torino, Turin, Italy
{simone.papicchio, luca.cagliero}@polito.it
[2] EURECOM, Biot, France
{paolo.papotti, simone.papicchio}@eurecom.fr

Abstract. Recent advancements in Tabular Representation Learning (TRL) and Large Language Models (LLMs) have achieved promising results in the Text2SQL task, which involves converting natural language questions about relational tables into executable SQL queries. However, when questions are ambiguously defined to the table schema, existing models often fail to produce correct outputs. Assessing the robustness of such data ambiguity is labor-intensive, as it requires identifying ambiguous patterns across many queries with varying levels of complexity. To address this challenge, we introduce the *Data-Ambiguity Tester*, a dedicated pipeline designed for ambiguous Text2SQL generation. This approach first generates a diverse set of unambiguous questions alongside their corresponding SQL queries. It then methodically injects ambiguous patterns from a human-annotated set of relational tables into these questions, simulating realistic schema ambiguities. Finally, the pipeline employs customized metrics to evaluate Text2SQL model performance under ambiguity. Our experimental results provide valuable insights into the strengths and limitations of current Text2SQL models.

Keywords: Text2SQL · Ambiguity · Automatic Generation.

1 Introduction

Text2SQL, also defined as Semantic Parsing, uses a relational table to translate natural language (NL) questions into SQL declarations. Text2SQL supports end users who are not proficient in SQL code writing and speeds up user-database interactions [2]. State-of-the-art Text2SQL approaches include Tabular Representation Learning (TRL) models fine-tuned for this task (e.g., [14, 15]) and general-purpose Large Language Models (LLMs)[5, 3]. However, both TRL models and LLMs are challenged by the inherent ambiguity between text (NL questions) and relational data (table schema and instance) [9, 12, 1].

This study focuses on *Column Ambiguity* since it represents the majority of ambiguous cases in practical scenario [12]. In *Column Ambiguity*, the NL question ambiguously references multiple table attributes (referred to as *Ambiguous attributes*) using a free-text label (referred to as *Ambiguous label*) that applies to at least two attributes [11]. As a toy example, let us consider the table

Abalone(AbaloneID, Sex, Length, Diameter). A NL question such as "Show me the *size* of the Abalone fish with Id 1" is ambiguous because the Ambiguous label *size* can be arbitrarily mapped to either *Length*, or *Diameter* (the Ambiguous attributes).

We propose DAMBER (*Data-AMBiguity testER*) a new pipeline for ambiguous test generation and evaluation. DAMBER relies on QATCH [7] to initially generate a large set of questions and SQL queries given a relational table. Then, the NL questions are injected with the Ambiguous labels extracted from an annotated human-curated set of tables [11]. The queries associated with the Ambiguous attributes represent the SQL queries related to the injected NL question. In our example, the NL question is paired with the three SQL queries:

```
- SELECT Length FROM Abalone WHERE AbaloneID = 1
- SELECT Diameter, FROM Abalone WHERE AbaloneID = 1
- SELECT Length, Diameter, FROM Abalone WHERE AbaloneID = 1
```

Once generated, each ambiguous question is given as input into the Text2SQL model and its single-query output is compared to each target query using five metrics [7]. The final results are based on the query interpretation (across the alternatives in the ground truth) with the highest mean across all metrics. For example, suppose the predicted query correctly projects *Length* but not *Diameter*; it receives a mean score of one for the first query and (almost) zero for the others. In that case, the final metric values are the ones with the highest mean.

Table 1. Results for all models with ambiguous questions; average on 13 tables.

Model	Cell precision	Cell recall	Tuple cardinality	Tuple constraint	Tuple order
GPT 3.5 (LLM)	**0.76**	**0.78**	**0.80**	**0.63**	0.83
LLama-code (LLM)	0.52	0.54	0.58	0.39	**0.86**
ResdSQL (TRL)	0.37	0.38	0.42	0.31	0.46
UnifiedSKG (TRL)	0.36	0.37	0.39	0.31	0.65
GAP (TRL)	0.24	0.24	0.26	0.21	0.27

We test five representative models: three TRL models (ResdSQL [4], GAP [10], and UnifiedSKG [13]) and two LLMs (GPT 3.5 turbo-0613 [6] and LLama-Code [8]). The findings in Table 1 demonstrate that LLMs significantly outperform TRL models when addressing ambiguous questions. LLMs are capable of linking ambiguous labels to the appropriate database attributes. In contrast, TRL models face challenges, frequently generating SQL errors in similar situations. However, GPT 3.5 struggles with consistent performance, showing an average over all metrics from 0.98 on simple tables to 0.60 on challenging ones.

In future work, we aim to broaden the generation beyond *Column Ambiguity*. Additionally, we will investigate how data ambiguity impacts other downstream tasks, including Tabular Question Answering and Tabular Computational Fact-Checking.

References

1. Bhaskar, A., Tomar, T., Sathe, A., Sarawagi, S.: Benchmarking and improving text-to-sql generation under ambiguity. In: Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing. pp. 7053–7074 (2023)
2. Floratou, A., Psallidas, F., Zhao, F., Deep, S., Hagleither, G., Tan, W., Cahoon, J., Alotaibi, R., Henkel, J., Singla, A., van Grootel, A., Chow, B., Deng, K., Lin, K., Campos, M., Emani, V., Pandit, V., Shnayder, V., Wang, W., Curino, C.: Nl2sql is a solved problem... not! In: 14th Annual Conference on Innovative Data Systems Research (CIDR'24). (2024)
3. Hong, Z., Yuan, Z., Zhang, Q., Chen, H., Dong, J., Huang, F., Huang, X.: Next-generation database interfaces: A survey of llm-based text-to-sql (2024), https://arxiv.org/abs/2406.08426
4. Li, H., Zhang, J., Li, C., Chen, H.: Resdsql: Decoupling schema linking and skeleton parsing for text-to-sql. In: AAAI (2023)
5. Liu, A., Hu, X., Wen, L., Yu, P.S.: A comprehensive evaluation of chatgpt's zero-shot text-to-sql capability. arXiv preprint arXiv:2303.13547 (2023)
6. OpenAI: Chatgpt 3.5 (2023), https://openai.com/blog/chatgpt
7. Papicchio, S., Papotti, P., Cagliero, L.: Qatch: Benchmarking sql-centric tasks with table representation learning models on your data. In: Thirty-seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track (2023)
8. Roziere, B., Gehring, J., Gloeckle, F., Sootla, S., Gat, I., Tan, X.E., Adi, Y., Liu, J., Remez, T., Rapin, J., et al.: Code llama: Open foundation models for code. arXiv preprint arXiv:2308.12950 (2023)
9. Saparina, I., Lapata, M.: Ambrosia: A benchmark for parsing ambiguous questions into database queries. In: Neural Information Processing Systems (2024)
10. Shi, P., Ng, P., Wang, Z., Zhu, H., Li, A.H., Wang, J., dos Santos, C.N., Xiang, B.: Learning contextual representations for semantic parsing with generation-augmented pre-training. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 35, pp. 13806–13814 (2021)
11. Veltri, E., Badaro, G., Saeed, M., Papotti, P.: Data ambiguity profiling for the generation of training examples. In: 2023 IEEE 39th International Conference on Data Engineering (ICDE). pp. 450–463 (2023). https://doi.org/10.1109/ICDE55515.2023.00041
12. Wang, B., Gao, Y., Li, Z., Lou, J.G.: Know what i don't know: Handling ambiguous and unknown questions for text-to-sql. In: Annual Meeting of the Association for Computational Linguistics (2023), https://api.semanticscholar.org/CorpusID:259859006
13. Xie, T., Wu, C.H., Shi, P., Zhong, R., Scholak, T., Yasunaga, M., Wu, C.S., Zhong, M., Yin, P., Wang, S.I., et al.: Unifiedskg: Unifying and multi-tasking structured knowledge grounding with text-to-text language models. arXiv preprint arXiv:2201.05966 (2022)
14. Yin, P., Neubig, G., Yih, W.t., Riedel, S.: TaBERT: Pretraining for joint understanding of textual and tabular data. In: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. pp. 8413–8426. Association for Computational Linguistics, Online (Jul 2020). https://doi.org/10.18653/v1/2020.acl-main.745, https://aclanthology.org/2020.acl-main.745
15. Yu, T., Wu, C.S., Lin, X.V., bailin wang, Tan, Y.C., Yang, X., Radev, D., richard socher, Xiong, C.: GraPPa: Grammar-augmented pre-training for table seman-

tic parsing. In: International Conference on Learning Representations (2021), https://openreview.net/forum?id=kyaIeYj4zZ