Making LLMs Work for Enterprise Data Tasks

Çağatay Demiralp, Fabian Wenz, Peter Baile Chen, Moe Kayali (UW), Nesime Tatbul, Mike Stonebraker

Background

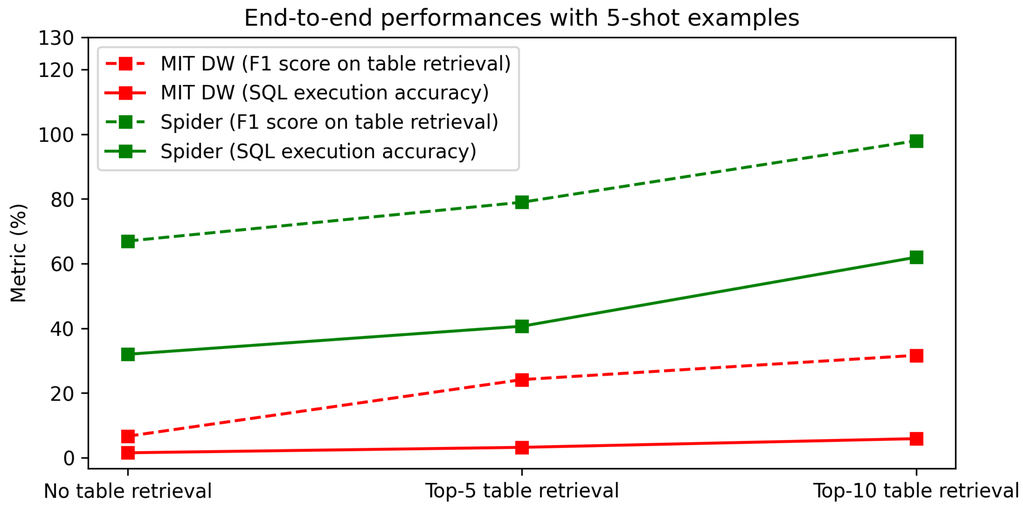
Large language models (LLMs) have shown strong performances on natural language (NL) comprehension tasks, from summarization to question-answering. To expand the scope of what LLMs can achieve, there are recent efforts that try to apply LLMs to data-related tasks, such as question-answering over tables, data cleaning, and data integration. Among these, the task of automatically translating natural language questions into SQL statements (text-to-SQL) for query purposes is a popular and challenging task. Given the complexity of databases, text-to-SQL can greatly empower non-technical users to perform data tasks without relying on experts, boosts SQL user productivity, and improves the usability of the data systems at large: jump start & use any tool.

Problem

Existing open-source datasets on text-to-SQL, Spider[1] and Bird[2], are criticized for their over-simplicity in terms of table structures and queries, which leads to unrealistic assumptions of solutions based on these datasets. An enterprise database, on the other hand, can have

- **Thousands of tables** instead of tens of tables in one database, which makes it impossible to fit all tables in one context and requires an intelligent retrieval system[3]
- **Domain-specific knowledge** (e.g., MIT buildings are numbered) and **non-obvious schema names** (FCLT, which stands for facility) which require external knowledge and additional schema understanding beyond simple semantic matching
- **Complicated queries** that involve joining 5 or more tables, compared to at most 4 joins in open-source datasets, which requires a more sophisticated SQL-generation strategy

While LLMs can achieve 60-70% accuracies on open-source datasets, their performances on enterprise datasets have not yet been explored and the task is far from being solved.



Experimental setup

- Data: 50 queries from the MIT Datawarehouse (DW) that involves 99 tables; 50 queries sampled from Spider that involves 11 tables
- Retrieval-augmented generation (RAG): retrieve top-k most relevant tables in terms of the cosine similarities of the question and table embeddings
- Model: GPT-3.5-turbo-16k with 5-shot examples and up to top-10 tables from RAG
- Metrics: F1 score on tables used in the gold and predicted SQL statements; exact-match execution accuracy between the outputs of the gold and predicted SQL statements

Results

- **Low performances on enterprise data compared to open-source data**: Under the setting of providing 5-shot examples and the top-10 most relevant table in the prompt, execution accuracy is 5.9%, and table retrieval F1 is 32.5% on the MIT DW and 62% and 63% on the Spider dataset. **Execution accuracy and retrieval F1 on MIT DW is 52% and 62% lower than the Spider dataset.**
- **Retrieval-augmented generation (RAG) helps with performances**: Compared to the setting without retrieval, providing the top-10 most relevant tables increases the performances of both metrics on the MIT DW by 325% (execution accuracy) and 284% (table F1). Additionally, the Spider data experienced similar behavior, where the accuracy increased by 97% and the F1 score by 41%.

Conclusion & Next steps

- Our results demonstrate that LLMs cannot generalize to enterprise datasets due to scale, domain-specific knowledge, and query complexity.
- Our project aims to open-source a benchmark dataset from various enterprise sources and explore solutions to improve LLMs' performances on these more complicated datasets.

References

[1] Yu, Tao, et al. "Spider: A large-scale human-labeled dataset for complex and cross-domain semantic parsing and text-to-sql task." arXiv preprint arXiv:1809.08887 (2018).

[2] Li, Jinyang, et al. "Can llm already serve as a database interface? a big bench for large-scale database grounded text-to-sqls." Advances in Neural Information Processing Systems 36 (2024).

[3] Chen, Peter Baile, Yi Zhang, and Dan Roth. "Is Table Retrieval a Solved Problem? Join-Aware Multi-Table Retrieval." ACL 2024.