DIAS: Data-Intensive Applications and Systems Laboratory
School of Computer and Communication Sciences
Ecole Polytechnique Fédérale de Lausanne
Building BC, Station 14
CH-1015 Lausanne
URL: https://www.epfl.ch/labs/dias/

Effective Query Synthesis for Table-Augmented Generation

Keywords: LLM, semantic operator, TAG, query synthesis, RL, CoT

Problem: Semantic operators [1] offer semi-declarative interface to build pipelines over structured and unstructured data. They are not fully declarative so that the pipelines still need to be built by programmers, unlike SQLs that do not specify the execution flows. They are declarative so that bulk semantic processing can be done with a few lines of codes, without manual prompting and iteratively running ad-hoc LLM requests.

Table-augmented generation (TAG) [2] built on semantic operators suggests a new way to tackle the challenges not solved by RAG (from LLM domain) and Text2SQL (from DB domain); RAG simply performs point lookups and tries to answer questions with such fragmented information, while Text2SQL focuses on a limited set of questions that can be translated into SQLs. In contrast, TAG suggests a three-phase approach to resolve their limitations: query synthesis, query execution, and answer generation. Query synthesis generates LLM-enhanced SQLs (pipelines with semantic operators) that contain useful information to answer questions, followed by query execution to actually generate the data, and answer generation takes the data to answer the original question as in RAG.

While such an approach can handle a wider range of queries more effectively, by incorporating both world knowledge and reasoning from LLMs and domain knowledge and exact, efficient computation from DBs, an effective query synthesis method has not been proposed. The authors use hand-written pipelines in their own benchmark built for TAG. Unfortunately, query synthesis is an extremely challenging task, similar to the notorious query optimization in DBs that has not been solved perfectly for several decades. Optimizing ML query pipelines for compound AI systems and agentic LLMs has just started to appear [5, 6, 7].

On the other hand, OpenAI’s latest model, GPT-o1 [3], offers powerful reasoning capability adequate for complex tasks. While the exact mechanism is unveiled, it is known that the model has learned to exploit chain-of-thought (CoT) reasoning using RL, generating multiple reasoning paths, revising them, and selecting the best path to generate the most suitable answer. Since such a combination of LLMs and RL (using LLMs as agents, action generators, and reward functions) is a promising direction also discussed in rStar [4] from MS, we aim to apply such approaches to query synthesis. While we expect that this direction is also valid for traditional query optimization, applying it to TAG and semantic operators fits better as the query execution with heavy LLM calls can mitigate the overhead of LLM calls in query synthesis.

[1] LOTUS: Enabling Semantic Queries with LLMs Over Tables of Unstructured and Structured Data
[2] Text2SQL is Not Enough: Unifying AI and Databases with TAG

[3] GPT-o1 related links
https://openai.com/index/learning-to-reason-with-llms/,
https://medium.com/@tsunhanchiang/openai-o1-the-next-step-of-rl-training-692838a39ad4
https://ai.plainenglish.io/deep-dive-into-gpt-o1-58765058745b
https://sbagency.medium.com/openai-o1-alternatives-reasoning-is-all-you-need-683677e2ecbe
https://platform.openai.com/docs/guides/reasoning/advice-on-prompting
[4] Mutual Reasoning Makes Smaller LLMs Stronger Problem-Solvers
[5] nsDB: Architecting the Next Generation Database by Integrating Neural and Symbolic Systems
[6] Automated Design of Agentic Systems
[7] Hydro: Adaptive Query Processing of ML Queries
[8] Self-Harmonized Chain of Thought

Project: In this project, the student will 1) survey the integration techniques of LLMs and RL for complex reasoning tasks, 2) apply the techniques to query synthesis in TAG, 3) analyze successful and failure cases, and 4) further optimize the performance.

Plan:

1. Survey the integration techniques of LLMs and RL for complex reasoning tasks such as rStar from MS and query optimization in nsDB.
2. Implement the techniques for the query synthesis module in TAG.
3. Analyze the performance and identify the bottlenecks.
4. Optimize the performance.

Supervisor: Prof. Anastasia Ailamaki, anastasia.ailamaki@epfl.ch
Responsible collaborator(s): Dr. Kyoungmin Kim, kyoung-min.kim@epfl.ch

Duration: 5 months