Group 3 Project 4 Plan

Jack Rubin, **Anthony Piacentini**, **Theodore Steiger**

Abstract

This document outlines our project plan for evaluating the shared task focused on question answering over tabular data.

1 Introduction

This project aims to address the gaps in pre-existing large language models (LLMs). We hope to identify the strengths and weaknesses in the current offerings, specifically when LLMs are used for answering over tabular data.

We would like to evaluate and compare the performance of different LLMs in the hopes of figuring out how they can be improved in order to deliver more desirable results.

We look to expand upon the pre-existing evaluation methods, including those mentioned in (Osés Grijalba et al., 2024). This will enable us to gain a more effective understanding to improve the question and answering of tabular data capabilities.

We hope to gain an understanding of the differences between different kinds of large language models. More specifically, these differences will not just include open vs. closed source, but also free vs. paid access, transformer based vs. non transformer based, chain of thought reasoning vs. non chain of thought, and finally context window size differences.

2 Prior Work

This project is directly inspired by the work done by (Osés Grijalba et al., 2024), which introduces DataBench, a benchmark that can be used to evaluate LLMs when answering questions over tabula data. The main contribution is that DataBench features a larger and more diverse dataset than previous work.

The paper also examines different prompting techniques, that being zero shot in-context learning and code-based prompting. In our project, we will look for a way to combine these two techniques to yield better results and reduce overall coding related errors.

3 Data and Software

This project was originally assigned as a competition on codabench.com. This provides us with the necessary evaluation program and datasets.

Starting with the evaluation program, which comes from the git repository titled ”databench_eval.” In this repository, it gives detailed information on how to download and the basic use cases of their program, along with multiple examples of more complex uses to show how to best use the software. Another piece that is given to us from prior works is the competition_dataset.zip. This folder contains a csv file which stores the questions to be answered along with the ID of the corresponding dataset. This folder also has 24 subfolders each named with the unique identifier and the title of the company that produced the data set. This folder has already been downloaded and input into our shared git repository.

The final piece of data given to us is the repository databench available on Huggingface. This repository has the original 65 data sets used in the (Osés Grijalba et al., 2024) paper. This repository also contains documentation on the importing and usage of these additional 65 data sets.

3.1 List of Data and Software

- databench_eval package
- Huggingface Dataset
- competition_databench.zip
- test_qa.csv

4 Task Plan

4.1 Performance Task

In order to improve the performance of these LLMs, the primary objective, it will be important to evaluate all of the models to see more generally what weaknesses these LLMs have when answering Boolean, categorical, numerical and list-based questions.

Along with this, we hope to figure out effective prompt engineering strategies that lead to better results from these models. If possible, we also hope to do some fine-tuning of these models in order to further improve these results. In order to do the best possible job of fine-tuning and prompt engineering so that these models can be effective as possible, we will use our analysis of the differences between models to figure out what characteristics of a model lead to success.

4.2 Evaluation Task

In order to improve evaluation techniques, some strategies may include addressing some of the formatting inconsistencies between different LLM outputs that may have previously led to evaluations that did not capture the full spectrum of the LLMs' capabilities.

It will also be important to experiment with better validation techniques in terms of accessing correctness. More specifically, we may look to develop validation techniques beyond that of simple accuracy scores. Some initial ideas that we have for this include testing the LLMs on their tolerance to noise in the data to see how that could play a role in their answering ability. Another idea is to see whether the model can follow a reasonable multi-step reasoning validation.

When handling errors, we hope to create effective error classification systems that will be able to evaluate more than simply if an error occurred, but more importantly, how and why the error occurred and why it may not occur when using other LLMs. To do this, many of the evaluation techniques mentioned above will be very effective, for example, analysis of the multi-step reasoning path that the model took in order to get to the wrong answer.

4.3 Experimentation with Different Language Models Task

In order to evaluate the differences (strengths and weaknesses) of different types of LLM models as they apply to our specific tasks, we can use many of the evaluation techniques mentioned above. However, this being said, we also intend to apply more targeted metrics when comparing models with different features in order to really highlight their differences. Some examples of this include a performance-to-cost ratio when comparing free (or relatively cheaper models) to more expensive or query intensive strategies. Another similar metric that would highlight the difference between models with largely different context windows would be to test for information retention. Going further from this method of evaluation, we intend to work towards testing context-aware querying to test the performance of the model on multi-turn questions where the previous context is very important.

References

Yevgeni Berkovitch, Oren Glickman, Amit Somech, and Tomer Wolfson. 2024. Generating tables from the parametric knowledge of language models. *Preprint*, arXiv:2406.10922.

Kushal Raj Bhandari, Sixue Xing, Soham Dan, and Jianxi Gao. 2024. On the robustness of language models for tabular question answering. *Preprint*, arXiv:2406.12719.

Sebastian Bordt, Harsha Nori, and Rich Caruana. 2024. Elephants never forget: Testing language models for memorization of tabular data. *Preprint*, arXiv:2403.06644.

Xi Fang, Weijie Xu, Fiona Anting Tan, Jiani Zhang, Ziqing Hu, Yanjun Qi, Scott Nickleach, Diego Socolinsky, Srinivasan Sengamedu, and Christos Faloutsos. 2024. Large language models(llms) on tabular data: Prediction, generation, and understanding – a survey. *Preprint*, arXiv:2402.17944.

Jorge Osés Grijalba, L. Alfonso Ureña-López, Eugenio Martínez Cámara, and Jose Camacho-Collados. 2024. Question answering over tabular data with DataBench: A large-scale empirical evaluation of LLMs. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 13471–13488, Torino, Italia. ELRA and ICCL.

Vaishali Pal, Andrew Yates, Evangelos Kanoulas, and Maarten de Rijke. 2023. Multitabqa: Generating tabular answers for multi-table question answering. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, page 6322–6334. Association for Computational Linguistics.

Hengyuan Zhang, Peng Chang, and Zongcheng Ji. 2023. Bridging the gap: Deciphering tabular data using large language model. *Preprint*, arXiv:2308.11891.