Evaluating LLM performance on question-answering based on Wikidata triples

Aksel Mads Madsen

s232448

Jun 24, 2024

Abstract

This report evaluates the performance of three large language models (LLMs), GPT4-o, Claude 3 Sonnet, and Mistral Large, on a question-answering task derived from Wikidata triples. We generate a novel dataset of 1000 question-answer pairs from randomly sampled Wikidata entries, to avoid evaluation dataset contamination. The LLMs' responses are evaluated using embedding distances. We employ a two-way ANOVA and a Tukey Post-Hoc test to assess performance differences and potential biases in the evaluation method. Results indicate statistically significant differences between models, with GPT4-o and Claude 3 Sonnet outperforming Mistral Large, though the effect sizes are small. The report also investigates whether embedding distances are a good way of evaluating LLM answers, finding a moderate correlation ($R^2 = 0.4$) with BLEU scores and no significant bias towards specific models.

1 Introduction

Evaluating LLMs have proven to be a difficult task. The tasks LLMs are used for tend to be exactly the kind of task that is hard to quantify and evaluate, making benchmark design difficult. We evaluate the performance of three LLM using using methods we hope will avoid these problems. The chosen task is answering questions around atomic pieces of knowledge. The main problem here is comparing the answer given by the LLM to the known, correct answer. LLMs differ in style and doing a direct string-match will inevitably favor some LLMs over others, and give an incorrect view of their performance, both relative and absolute. To combat this we use embedding distances as scores.

In addition, LLMs large need for training data necessitate training them on large text corpora, making data contamination common [1]. Data contamination inevitably occurs with any publicly available benchmark, and can easily cast doubts on any such regular benchmarks. The only solutions are private benchmarks or making up whole new tasks with the same difficulty when testing models trained after the benchmarks were released, making comparison difficult. To combat this we generate a fresh dataset, and offer a method for repeatedly doing the same on-demand, while still maintaining a consistent difficulty.

2 Background

2.1.1 Wikidata

Wikidata [2] is a large open knowledge graph, affiliated with the Wikipedia project. It contains a diverse set of facts, organised as “triples”, which are relations between objects. For example, the triple `(Public Parks (Ireland) Act 1869 | legislated by | Parliament of the United Kingdom)` represents a relation between two objects `Public Parks (Ireland) Act 1869` and `Parliament of the United Kingdom`, which has type `legislated by`.

Wikidata contains a large amount of highly specific data about diverse topics. Using randomly sampled data from wikidata is useful for evaluating language models, since the variety of facts make it less biased towards specific topics than other similar sources [3].

2.1.2 Embeddings

Embeddings are an NLP technique for generating vectors that represent the semantic meaning of a piece of text. In an embedding the text under analysis is run though the first layers of a transformer language model, and the internal residual stream of the model is then taken as a vector, which is called the embedding vector, or just embedding. This vector is the internal representation the model uses for that piece of text. Empirically distances between pieces of text correspond to qualitative semantic differences between the meanings of the texts. This technique is primarily used to enable semantic search of existing unstructured textual data.

3 Methods

3.1 Dataset

One particular challenge when evaluating LLMs have proven to be benchmark contamination, the phenomenon that open benchmark dataset tend to end up on the web and later in the training datasets for new LLMs (see [1], Appendix 6). Our evaluation avoids this by generating a new evaluation set of natural language questions and answers from randomly sampled wikidata entries.

We construct a dataset by extracting random triple samples from the wikidata database. The extracted data is clean and filtered, mainly by removing data without labels in english and removing relations representing various identifiers in external databases, since these make up a large proportion of wikidata, but contain very little interesting information. Each triple is transformed to a natural language question-answer pair, using the Claude API. The prompt contains ten examples taken from the [4] dataset. These question-answer pairs are manually checked. The dataset contains 1000 such examples of questions and answers.

3.2 Pilot & sample size estimation

Though the setup allows for generating a practically infinite amount of questions, using an LLM for generating the natural-language questions means we do not want to generate more samples than necessary. Therefore we perform a pilot study by first generating 100 samples and using the observed effect size to estimate how many samples we would need to reveal an effect of the estimated effect size at a significance threshold of .05. The test conducted using using F-test power analysis and solving for the sample size. The test estimated that 1085 samples would be sufficient for a significant result, which is well within our ability to generate.

3.3 Evaluation

Three language models are evaluated for their performance on the task of answering questions from the dataset: `Claude Sonnet 3.5`, `Mistral Large` and `gpt4-o`. Each model is prompted to answer each question in a zero-shot setting, with a prompt requesting brief answers, since all models are prone to give explanations, whereas the answers in wikidata are short, and contain no explanation. To score the answers an embedding is generated for the correct answers, and the answer from each model. The euclidean distance between a model's answer and the correct answer from wikidata is taken as the model's score. Embeddings are supplied from the OpenAI and Mistral embedding API. Two different embeddings suppliers are used because of concerns that the embedding model might score text generated by its corresponding generative model differently than other text.

Since the Mistral embeddings use 1024-dimensional vectors, and the OpenAI embedding model uses 3072 dimensions the raw distances will differ on average, and thus be incomparable. In addition the embeddings could have different structures, that also invalidates direct distance comparisons. To alleviate this we transform the distances using the rank based inverse normal transformation within each group of answers embedded by the same model. This transformation is as valid a representation as the raw distances, since those are only meaningful when compared to one another anyway. The transformed scores can be compared directly between different rater models.

In addition to the embeddings the more traditional metric of BLEU [5] is used for scoring some questions. The purpose of this is to evaluate the method of using embeddings for evaluating answer correctness against alternative approaches.

4 Results & Analysis

Model	Mean distance
GPT4-o	-0.1013
Claude 3 Sonnet	-0.0038
Mistral Large	0.1051

Table 1: Mean normalized distance from correct answer (in standard deviations)

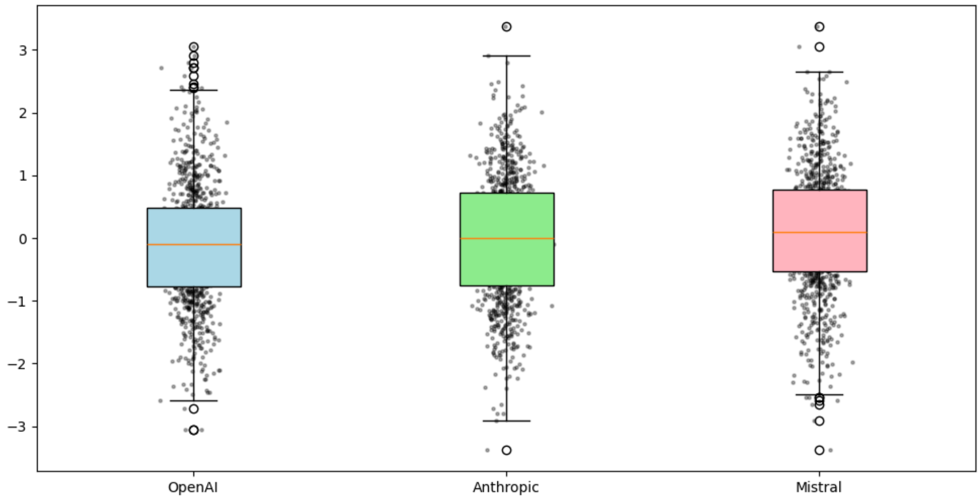

Figure 1: Box-plots of the normalized distances, by model under evaluation

Term	P-value	Effect Size (F^2)
Answerer	$6.6 \cdot 10^{-5}$	$7 \cdot 10^{-3}$
Answerer-Rater interaction	0.44	$5 \cdot 10^{-4}$

Table 2: Results of a two-way ANOVA of the answer embedding distances

Summary statistics are in Figure 1 and Table 1. As is apparent the differences in model performance are very slight.

To use the results for comparing the models against each other, we must be sure that the models generating the answer embeddings do not advantage any one model. To do this we perform a two-way ANOVA test, including a term for the interaction effect between model and embedder specifically, the results of which are in Table 2. The hypothesis that the interaction term is not null has a p-value of 0.5, so we assume there is no such effect moving forward. The term for the model yields a p-value of $6.6 \cdot 10^{-5}$ which is less than 0.05, meaning we reject the null hypothesis that all models have the same mean performance.

A post-hoc test reveals that the worse-performing model is mistral, and that the p-value for a comparison between GPT4-o and Claude 3.5 Sonnet is at 0.09, meaning we do not reject the hypothesis that these two models have identical performance. The effect sizes of the differences between the performance of the models are very small, meaning that the models perform almost as well as each other. That we have a low p-value despite small effect sizes can be attributed to our large sample size, which increases the power to detect small differences between the models. However, qualitative manual inspection of some answers indicate that there are notable differences in the quality of the answers. We theorise that his comes from many questions in the dataset being either too hard or too easy, meaning that no information is gained from them.

4.1 BLEU score analysis

When analyzed with One-Way ANOVA on the Z-values of the BLEU scores, a similar result emerges. In addition we find that the BLEU scores are correlated with the embedding distances

($R^2 = 0.4$) (Figure 2). This is good, since they are both measures of answer quality, so we would hope that they correlate.

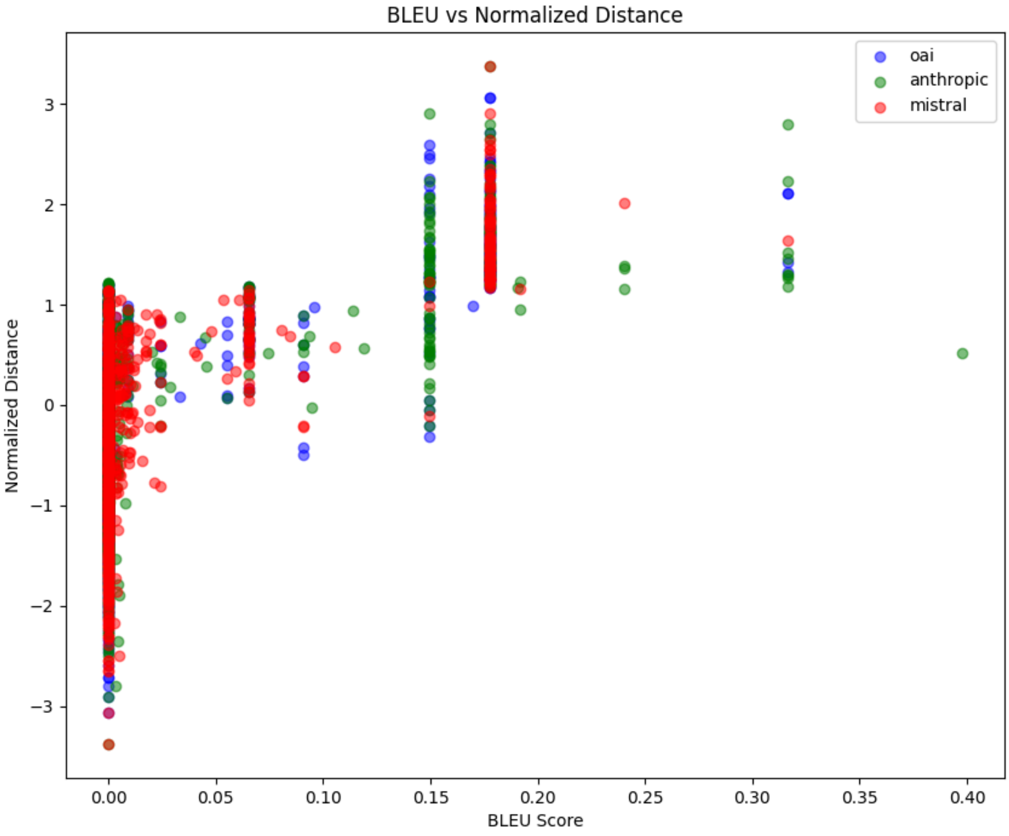

Figure 2: Plot of relationship between BLEU ratings and normalized distance ratings (in standard deviations), colored by LLM

5 Conclusion & Discussion

We conclude that there are statistically significant (though small) differences in performance between the evaluated models and that GPT4-o and Claude 3 Sonnet have better performance than Mistral Large in this task.

In addition to evaluating the models, we also set out to investigate whether embeddings are useful in the role of evaluating performance on question-answering tasks. Since we did not observe a significant interaction between the embedding and generating model, we believe that this is not an obstacle to using this technique, at least in this setting. Furthermore using embeddings gives similar results to BLEU, and will therefore not generate surprising results in existing tests.

5.1 Limitations

Using LLMs for both question generation and rating while evaluating the LLMs themselves makes ample room for biases. We have attempted to eliminate these in the embedding models by comparing multiple LLMs in this role, but only a single LLM was used for generating questions. This could have included bias in the kind of questions in the dataset.

Properly generating questions from triple data turned out to be more difficult than expected, and there are still issues here. In particular some kinds of information are vastly overrepresented in the wikidata graph (chemical structures, iranian villages, drug trials). Because no biasing of the dataset towards more useful facts was done these are also overrepresented in the questions. In addition some triples did not contain enough data to usefully ask a question, usually because one or both of the entities were ambiguous.

Including more data from the surrounding graph, in particular lengthier descriptions of entities, would have allowed for much better questions. In addition the question-generating LLM could have been prompted to filter out triples with insufficient information to generate a good question-answer pair.

As an attempt to find the best LLM for questions answering, the report falls short, since only the result that Mistral is inferior is significant. In general, knowing which model is best is more important than knowing which model is worst, and this study does not tell us this.

Bibliography

[1] H. Touvron *et al.*, “Llama 2: Open Foundation and Fine-Tuned Chat Models.” 2023.

[2] D. Vrandečić and M. Krötzsch, “Wikidata: a free collaborative knowledgebase,” *Communications of the ACM*, vol. 57, no. 10, pp. 78–85, 2014.

[3] bigbench, “Beyond the Imitation Game: Quantifying and extrapolating the capabilities of language models,” *Transactions on Machine Learning Research*, 2023, [Online]. Available: https://openreview.net/forum?id=uyTL5Bvosj

[4] K. Han, T. C. Ferreira, and C. Gardent, “Generating questions from Wikidata triples,” in *13th Edition of its Language Resources and Evaluation Conference*, 2022.

[5] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, “Bleu: a method for automatic evaluation of machine translation,” in *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, 2002, pp. 311–318.

Code is availabe at https://github.com/retonlage/02445-report-llm-evaluation.