Study the Generated Texts from Published LLMs Based on Completing Truncated Text Pairs

Group Member: Ruimeng Shao

Introduction:

With the growing popularity of Large Language Models (LLMs), numerous models have been published online. However, questions remain about how each LLM functions and how their core theory or training data may influence their performance. To investigate these differences, we designed an experiment to compare various LLMs. We used the Stanford Sentiment Treebank dataset, which contains truncated text, as input for the LLMs to generate completed text pairs. We then created a new dataset consisting of the original inputs, the generated outputs from each LLM, and corresponding labels. Finally, we developed a classifier to determine whether we could identify which LLM produced a given text pair.

Related Works:

Recent advancements in Large Language Models (LLMs) have led to numerous research studies aimed at exploring their capabilities, weaknesses, and applications in various fields. [1] Törnberg (2023) provides a comprehensive guide on utilizing LLMs for text analysis, highlighting the versatility of LLMs in performing complex natural language processing tasks. His work emphasizes how LLMs can assist researchers in extracting patterns from large volumes of text, automating content classification, and generating meaningful summaries. However, Törnberg also raises concerns about the limitations of LLMs, particularly in interpreting nuanced or context-dependent information, thus suggesting that human expertise is still essential in some text analysis scenarios.

Another area of exploration involves improving the retrieval-augmented LLMs for better information processing. [2] Xia et al. (2024) focus on enhancing the interaction between generated text and external information sources. Their study introduces an innovative method for improving retrieval-augmented LLMs by interleaving reference-claim generation to ground LLM outputs more effectively. This method significantly improves the accuracy and relevance of LLM-generated content by ensuring that responses are not only coherent but also well-supported by external factual information, mitigating a common issue of hallucination in LLMs.

Evaluating LLM-generated content has also been an area of active research. [3] Van Schaik and Pugh (2024) offer a detailed examination of automatic evaluation methods for summaries produced by LLMs. Their work critiques the conventional evaluation metrics and proposes new approaches that more accurately assess summary quality. They emphasize the importance of human-in-the-loop evaluations, combining automatic metrics with human judgment to provide a more holistic view of LLM performance. Their study highlights the complexities in evaluating abstract tasks like summarization, where traditional metrics may fail to capture nuanced qualities such as readability, coherence, or informativeness.

[4] Chen et al. (2024) and [5] Tang et al. (2024) focus on identifying the weaknesses and potential detection of LLM-generated text. Chen et al. introduce a self-challenge framework that probes LLMs with deliberately complex or ambiguous questions to uncover their limitations. This method not only identifies gaps in LLM knowledge but also sheds light on areas where LLMs can be easily misled. Similarly, Tang et al. (2024) investigate techniques to detect text generated by LLMs. Their work explores both linguistic and statistical approaches to distinguish between human-generated and machine-generated content, addressing the growing concern over the use of LLMs in misinformation and automated content generation. Together, these studies contribute to a deeper understanding of LLMs' strengths and vulnerabilities, highlighting the need for ongoing evaluation and improvement as these models continue to evolve.

Causal LLMs: Causal language models, such as GPT and its variants, generate text in an autoregressive manner, meaning they predict the next word in a sequence based solely on the previous words. These models process input in a left-to-right fashion, which makes them well-suited for tasks like text generation, completion, and conversation. Causal LLMs are typically used for generating coherent text over long sequences because they generate words one at a time, relying on past context without seeing future tokens during training.

Seq2Seq LLMs: Sequence-to-sequence (Seq2Seq) models, like T5, BART, and FLAN-T5, are designed for tasks where both input and output are sequences, such as translation, summarization, and question answering. These models typically consist of an encoder-decoder architecture. The encoder processes the input sequence, converting it into a hidden representation, and the decoder generates the output sequence based on that representation. Seq2Seq models are well-suited for tasks requiring structured transformation from input to output, especially when generating responses that are conditioned on the entire input context.

Embedding LLMs: Embedding models, such as BERT, DistilBERT, MiniLM, and Electra, focus on generating contextualized vector representations (embeddings) for each token in the input. These models are typically bidirectional, meaning they consider both the left and right context of each word to generate embeddings that capture the meaning of the token within its broader context. Embedding LLMs are often used for tasks like classification, sentence similarity, and token-level prediction, where understanding the semantic relationships between words is crucial. They do not typically generate new text but are instead used to produce meaningful representations of input text for downstream tasks.

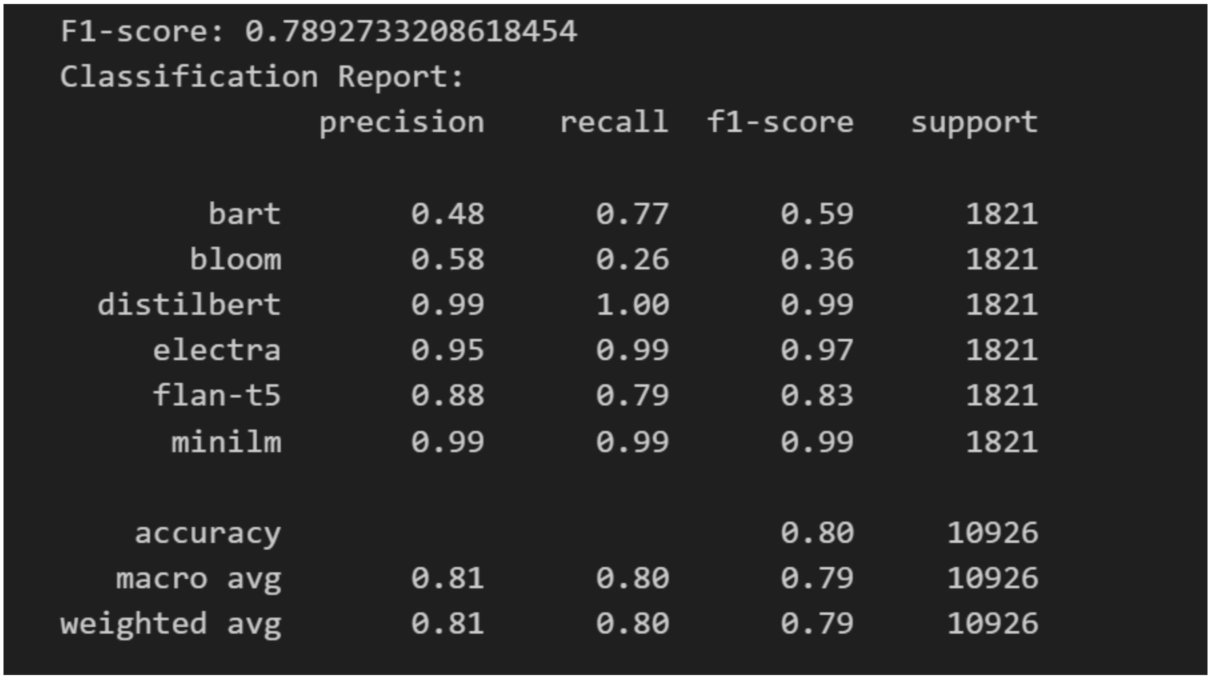
Methods:

BERT (Bidirectional Encoder Representations from Transformers) is a pre-trained language model designed to generate contextual embeddings by considering both the left and right context of each word in a sentence. BERT is pre-trained using two primary tasks: Masked Language

Modeling (MLM), where a portion of the input tokens is masked and the model predicts the missing words, and Next Sentence Prediction (NSP), which helps the model understand relationships between sentence pairs, and in our cases, we utilized pre-trained BERT to extract embeddings from the input text and generated output text.

Dataset & Experiments:

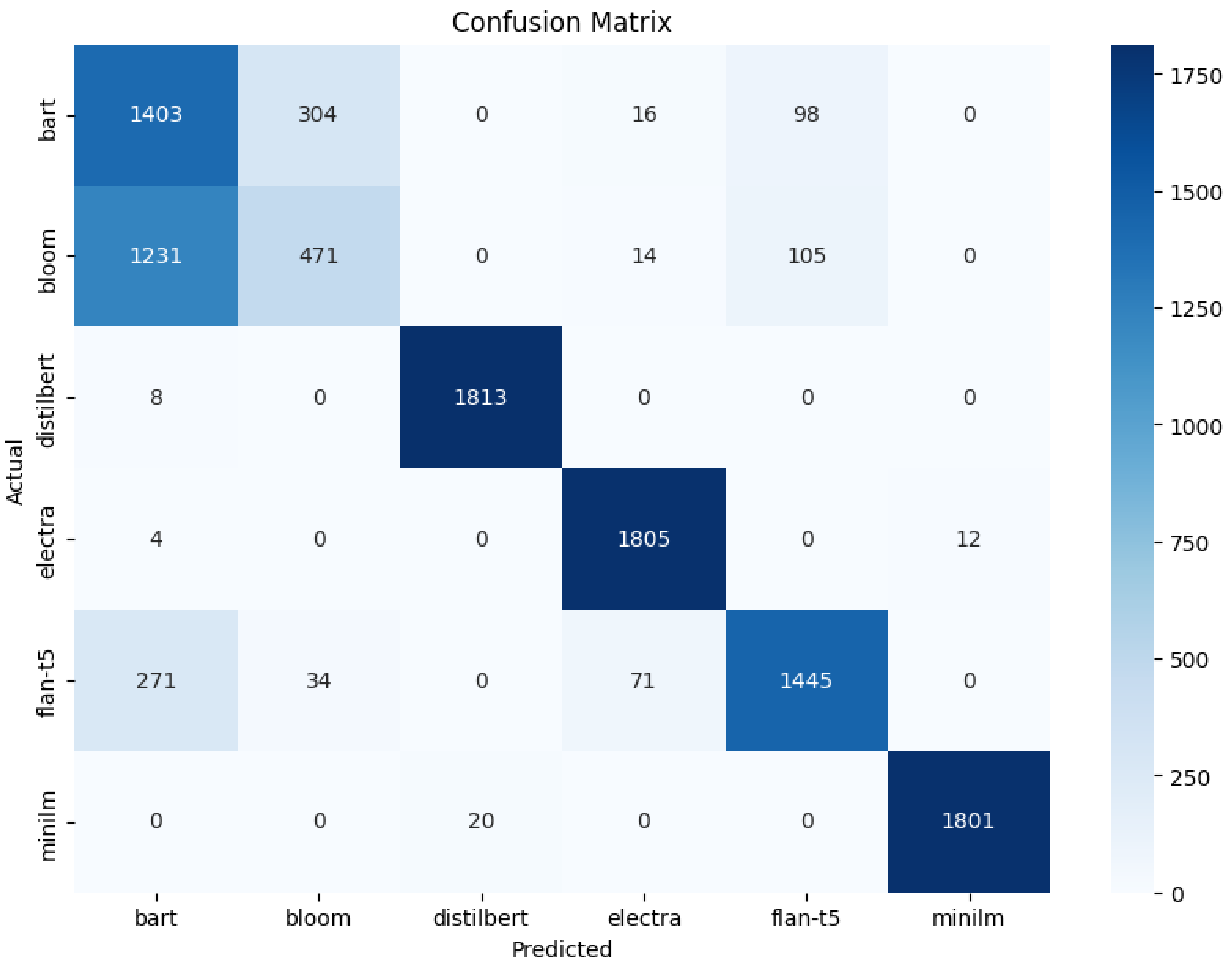
Using the outputs from six LLMs (Flan-T5, Bloom, BART, DistilBERT, MiniLM, and Electra) based on the Stanford Sentiment Treebank dataset, we constructed a new dataset containing the original inputs, the corresponding outputs from each LLM, and labels indicating the LLM used. However, due to the limitation of computing resources, we compressed the training dataset into 10k rows, validation sets to 872 rows, and test sets to 1.82k rows. We then utilized a pre-trained BERT model to extract embeddings from both the original inputs and the LLM-generated outputs. These embeddings were subsequently fed into a logistic regression model for the classification task. To evaluate the performance of our model, we generated a confusion matrix for the six target LLMs and calculated the F1-score as showing below:



```
F1-score: 0.7892733208618454
Classification Report:
              precision    recall  f1-score   support

        bart       0.48      0.77      0.59      1821
       bloom       0.58      0.26      0.36      1821
  distilbert       0.99      1.00      0.99      1821
     electra       0.95      0.99      0.97      1821
     flan-t5       0.88      0.79      0.83      1821
      minilm       0.99      0.99      0.99      1821

    accuracy                           0.80     10926
   macro avg       0.81      0.80      0.79     10926
weighted avg       0.81      0.80      0.79     10926
```



	bart	bloom	distilbert	electra	flan-t5	minilm
bart	1403	304	0	16	98	0
bloom	1231	471	0	14	105	0
distilbert	8	0	1813	0	0	0
electra	4	0	0	1805	0	12
flan-t5	271	34	0	71	1445	0
minilm	0	0	20	0	0	1801

Conclusion & Discussion:

With an F1 score of 0.789 from the classification model, our results demonstrate that we can effectively identify text generated by different Large Language Models (LLMs). While there is room for improvement, the model shows strong performance in distinguishing between outputs from various LLMs, highlighting the potential for automated detection of LLM-generated content. However, the confusion matrix reveals that text generated by LLMs of similar types tends to be misclassified into one another, suggesting that these models produce outputs with comparable patterns that make them harder to differentiate.

For the future works, since we don't have enough time to perform too many experiments, we only test the BERT based embedding extractions and it is showing that our model is performing extremely well on Distilbert which is a LLM developed based on BERT.

Reference:

[1] Törnberg, Petter. "How to use llms for text analysis." *arXiv preprint arXiv:2307.13106* (2023).

[2] Xia, Sirui, et al. "Ground Every Sentence: Improving Retrieval-Augmented LLMs with Interleaved Reference-Claim Generation." *arXiv preprint arXiv:2407.01796* (2024).

[3] van Schaik, Tempest A., and Brittany Pugh. "A Field Guide to Automatic Evaluation of LLM-Generated Summaries." *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 2024.

[4] Chen, Yulong, et al. "See What LLMs Cannot Answer: A Self-Challenge Framework for Uncovering LLM Weaknesses." *arXiv preprint arXiv:2408.08978* (2024).

[5] Tang, Ruixiang, Yu-Neng Chuang, and Xia Hu. "The science of detecting llm-generated text." *Communications of the ACM* 67.4 (2024): 50-59.