Building an LLM identifiability challenge

Guy Shpigler and **Nir Mezamer** and **Yuval Mizrahi** and **Yuval Mor**
School of Computer Science, Tel-Aviv University
School of Electrical Engineering, Tel-Aviv University
{shpigler,nirmezamer,yuvalmizrahi,yuvalmor3}@mail.tau.ac.il

Abstract

Natural Language Processing (NLP) has made remarkable progress in recent years across various language-related tasks, and the use of large language models (LLMs) for text generation, summarization, and writing is becoming increasingly widespread. However, these advancements have also introduced a new challenge, as distinction between human-generated and machine-generated text becomes less clear. With LLMs becoming more sophisticated, they can produce text that closely resembles human-written content in terms of style, syntax, and coherence. This blurring of lines poses significant implications for various applications where textual authenticity is crucial, such as in journalism, social media, and online encyclopedia (Wikipedia) where they all serve as primary sources of knowledge for people worldwide. Our research project is designed to help answering this fundamental question:
Given a text, was it generated by a human or by a large language model (LLM)?
To address this question, we first design a pipeline to gather diverse human-generated texts in specific domains (Wikipedia, Reddit posts and comments, newspaper articles). By selecting a wide array of sources, we ensure that our dataset captures a broad spectrum of writing styles, topics, and formats. This diversity is crucial for creating a robust challenge. Based on these human-texts, we then design a pipeline to collect LLM-based texts mimicking the same diversity and domains as the natural ones. This involves using the large language model Gemini[1] to generate texts that replicate the styles and contexts of the human texts, and can be modified to use other LLMs. Our goal is to create a dataset where the machine-generated texts are as indistinguishable as possible from the human-generated ones at first glance. Subsequently, the dataset will form the foundational training infrastructure for machine learning models.

[1] https://ai.google.dev/

1 Introduction

To address the classification question, we aim to build a robust infrastructure for creating a Neural Network model that, given a text input, can determine whether it was generated by a human or by an LLM. The model's performance will largely depend on the quality of the dataset we build, which will be used for the model's learning process. The main challenge of our project was to create two text corpora: one of texts generated by humans and the other of texts generated by LLMs. Both corpora need to be representative, diverse, and balanced in size. We started by building the corpus of human-generated texts. To meet the requirements mentioned above, we chose to extract texts from various sources such as Wikipedia, Reddit (a social network) and newspapers. To create this text corpus, we wrote python scripts that use open APIs such as Wikipedia API[2], Reddit API[3], and Newspaper3k API[4]. From these, we extracted Wikipedia pages, posts and comments from Reddit, and published newspaper articles from the BBC, which together form the corpus of human-generated texts. The topics in each domain were selected using a random mechanism to ensure as much diversity as possible. The pipepline for collecting human-generated texts can be easily extended to other domains and sources. Simultaneously, to create the dataset of texts generated by LLMs, we used the Gemini API. We hand-crafted several prompts for each domain to achieve the most appropriate one to use when generating texts. Using the labeled data, we trained a classifier model to distinguish between human-generated and LLM-generated texts. We used a pre-trained BERT model and fine-tuned it on our dataset.

[2] https://pypi.org/project/Wikipedia-API/
[3] https://praw.readthedocs.io/en/latest/
[4] https://newspaper.readthedocs.io/en/latest/

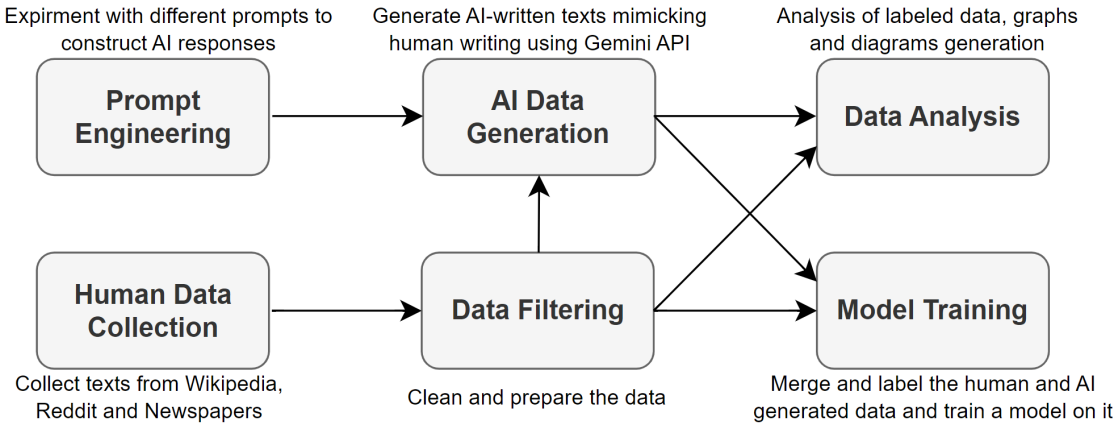

Figure 1: Project overview diagram

2 Related Work

Our project focuses on assessing the ability to distinguish between texts generated by humans and those generated by machines. Specifically, we aim to create a pipeline to gather dataset that serves as a robust infrastructure for training models to excel in this task. In order to perform this task to the best of our ability, we first researched academic papers that were published, addressing the same topic and gained insights regarding the most suitable working method for our project. In a recent study by Zubair Qazi and Papalexakis [2024] published in May 2024 by Qazi, Shiao and Papalexakis, they introduced GRiD (GPT Reddit Dataset), a new collection of texts generated by the Generative Pretrained Transformer (GPT). This dataset is designed to evaluate detection models' performance in identifying responses generated by ChatGPT. Despite the valuable insights gained from this study, since the dataset was entirely based on a single domain - Reddit, the question arises as to how texts from different domains and with different characteristics will be classified. We will examine this question in this project by constructing a diverse dataset as detailed above. In the following sections, we present our methodology, experimental setup, and results. For implementing the task of distinguishing between human-generated and machine-generated texts, a model was implemented relying on Hao Wang and Li [2024], which describes how to use a pre-trained BERT model and fine-tune it on a dataset.

3 Methodology

3.1 Objective

The main objective of the research is to explore the ability of different language models to distinguish between texts generated by humans and those generated by LLM (Large Language Models).

3.2 Data Collection

As mentioned above, the main task was to collect and generate the appropriate data for training the model. At this stage, we needed to collect authentic texts created by humans, texts with the same context generated by LLM, and process the generated data. During the research, we focused on three main sources of text:
Reddit, Wikipedia and Newspaper websites.

3.2.1 Reddit

Reddit is a social news aggregation, web content rating, and discussion website. Registered members submit content to the site such as links, text posts, and images, which are then voted up or down by other members. Posts are organized by subject into user-created boards called "subreddits", which cover a variety of topics including news, science, movies, video games, music, books, fitness, food, and image-sharing.
Our final dataset contains a collection of posts, where from each post we extracted one comment created by a human and one generated by an LLM. We gathered the human-written comments using the Reddit API (PRAW), and the AI-generated comments were created using the Gemini API. The goal during the collection process was to create a large,

reliable, and diverse dataset. To meet those criteria, we initially collected posts with comments from several subreddits, which were chosen randomly from the most popular subreddits list, in order to ensure as much diversity as possible. A total of 10,000 posts were collected. As mentioned above, for each post we extracted from Reddit, we chose one human-generated comment and one AI-generated comment. To create an AI generated comment for each post, we crafted a prompt which requests the model to create a suitable comment for the post. The prompt included the following fields: post's title, post's content, post's subreddit, and 4 of the post's comments. The comment which was used as the human-generated comment for the dataset was not included in the prompt. In this way, we ensure that our dataset is reliable and challenging for learning, as the comment generated by the LLM is strongly based on texts created by humans. Using 4 random comments for the prompt didn't yield good results, since most of the comments did not appear to match human text. For example, comments containing a URL link, such as a link to an image, "[REMOVED]" comments indicating that a user's comment was deleted, unusual characters like emojis, very short comments like "great" or "ok", etc. Additionally, some comments contained offensive or racist information and tone, which can not be used as part of the prompt (blocked by the Gemini API). To overcome the mentioned problem, a cleaning and preprocess procedure of the dataset was added. In the first stage, all non-text characters were removed from the comment. In the second stage, the list of comments in each post was sorted in such a way that the top comments in each post would be the best according to the parameters mentioned above. Of course, there is no direct way to do this, so we relied on external sources and created a weighting function that gives high scores to authentic comments and low scores to comments that are not authentic or offensive. Only around 70% of the posts were used in the final dataset, as the rest were filtered out due to the reasons mentioned above. This process proved successful, where the final dataset we created was in the format of a post with two comments: one is taken from the top 5 human comments, and the other is the comment generated by Gemini.

3.2.2 Wikipedia

Wikipedia is a free online encyclopedia, created and edited by volunteers around the world and hosted by the Wikimedia Foundation. The website has a free API, which was used to extract articles from Wikipedia. Only articles which has an attribute of "summary" were extracted - a total of 4000 articles. Half of the articles comprised the human generated part of the dataset. In addition, only the titles of the other half were inserted into an hand-crafted prompt, which in turn was fed to the Gemini model, to generate 2000 AI generated wikipedia style summaries. Many expirments were made in order to craft the most suitable prompt, and also to determine the best approach. At first the prompt didn't include a specific title, and the request was to generate a random article summary in a Wikipedia style. That attempt didn't work because the model repeated itself quite a lot. The second attempt was to give the model the title and body of the article (without the summary), and ask it to generate the summary. That attempt worked too well, generating a summary which is very similiar to the real summary - probably because the model was trained on that data. The final attempt was to give the model the title of the article, and it worked quite well. The code can be easily scaled up to generate more summaries, both human and AI generated - limited by the number of articles in the English Wikipedia, which is around 6 million.

3.2.3 Newspaper

First, we used the Newspaper3k API to obtain articles published recently on the BBC website. An attempt was made to obtain articles from various different news sites (CNN, Fox News, NYT...) but most sites blocked the API calls after a small amount of tries, requiring a payment for extensive API usage. We then crafted another prompt and sent it to Gemini, including the original article (title and content). The prompt included a request to recreate the article, keeping the generated content closely aligned with the given text, while emphasizing to the model to "write originally and use its own words". As opposed to Wikipedia, the model was not trained on the BBC articles, so the generated articles were more diverse and less similar to the original articles. During testing, we noticed that some of the articles extracted by the API were extremely short and unsuitable for inclusion in the dataset. By setting a minimum length requirement, we filtered out these shorter articles, which yielded better results. A total of 300 articles were collected, half of which were used to create the AI generated part of the dataset, and the other half were used as

the human generated part.

4 Document Body

References

Jiasnwei Li Hao Wang and Zhengyu Li. Ai-generated text detection and classification based on bert deep learning algorithm. *arXiv preprint arXiv:2405.16422v1*, May 2024. URL https://arxiv.org/abs/2405.16422v1.

William Shiao Zubair Qazi and Evangelos E. Papalexakis. Gpt-generated text detection: Benchmark dataset and tensor-based detection method. *arXiv preprint arXiv:2403.07321*, March 2024. URL https://arxiv.org/abs/2403.07321.

A Example Appendix

This is a section in the appendix.