**Project Title:** Comparative Analysis of Language Model Performance on Sentiment Analysis Using the Rotten Tomatoes Dataset

**Team Members:**

* Luis Becerra: Flan/Gemini algorithm owner and custom prompt engineering
* Alec Rogers: Programming lead for the embedding baseline and a custom sentiment analysis and
* Mini Sengupta: Bert and Llama algorithm owner

1. **Problem Statement:** Sentiment analysis, the task of determining the emotional tone behind text, is a crucial application of Natural Language Processing (NLP). With the rapid advancement of Large Language Models (LLMs) and the closed nature of their design but open usage model, there are many assessments (e.g. bias [1], x-metrics [2]) to characterize their performance. This project aims to conduct a comparative analysis of various open-source and commercially available LLMs on the Rotten Tomatoes movie review dataset, a widely recognized resource for sentiment classification. By comparing their accuracy, efficiency, and other relevant metrics, we hope to gain insights into current LLM-driven sentiment analysis.
2. **Project Goals:**

* Gain proficiency in the concept of embeddings as applied to dimensionality reduction and establishing semantic relationships in ML applications in the NLP domain
* Using the Rotten Tomatoes dataset, our goal is to compare the performance of selected LLMs (BERT, LLaMA 2, Flan) on sentiment classification using the Rotten Tomatoes dataset.
* Evaluate the accuracy, precision, recall and F1-score for each of the models under study. We will investigate other relevant metrics and add if relevant.
* Analyze the impact of different fine-tuning strategies on the APIs and prompt engineering techniques on model performance.
* Document the findings and provide recommendations for selecting appropriate LLMs for sentiment analysis tasks.

1. **Methodology**

* We plan to use Python (with pretty much all the libraries noted in the Class’ Canvas Python resource announcement) for programming
* We will use GitHub as the repository for the code
* We will start with building the baseline embedding model for the dataset to explore the relationships in the dataset
* Given that the dataset is labeled, we will process it through open-source LMs to derive sentiment analysis and score its performance against the labels
* We will then add some pre-conditioning statements (e.g. assume that this user is a fan of the science fiction movie genre) in the prompts sent to the LMs and investigate if the behavior of model changes as reflected in the scores
* We will also (time permitted) build a custom sentiment analysis engine for this dataset

1. **References:**

|  |  |
| --- | --- |
| [1] | J. Echterhoff, Y. Liu, A. Alessa, J. McAuley and Z. He, "Cognitive Bias in Decision-Making with LLMs," 2024. [Online]. Available: https://arxiv.org/abs/2403.00811v3. |
| [2] | M. Kahng, I. Tenney, M. Pushkarna, M. X. Liu, J. Wexler, E. Reif, K. Kallarackal, M. Chang, M. Terry and L. Dixon, "LLM Comparator: Visual Analytics for Side-by-Side Evaluation of Large Language Models," 2024. [Online]. Available: https://arxiv.org/html/2402.10524v1. |
| [3] | J. Alammar and M. Grootendorst, Hands-On Large Language Models: Language Understanding and Generation, O'Reilly Media, 2024. |