

Reddit Controversial Metric: LLM for Context-Aware Social Media Analysis (Proposal)

Luis Dale Gascon

Computer Science

Towson University

Towson, Maryland

lgascon1@students.towson.edu

Brendan Lauterborn

Computer Science

Towson University

Towson, Maryland

blaute3@students.towson.edu

Hermela Abraham

Computer Science

Towson University

Towson, Maryland

habraha3@students.towson.edu

Abstract—Social media platforms such as Reddit and 4Chan have a great deal of niche communities. These smaller communities often have opinions that contradict other communities and new members may not know what a communities' respective norms are. By fine-tuning an LLM on Reddit data, we plan to have a model that inputs a draft comment/post and predict an upvote/downvote score, alongside the model's reason for the score.

Index Terms—Reddit, Controversy, multi-modal, Parameter Efficient Fine-Tuning

I. INTRODUCTION

Due to the size of the platform, Reddit contains an immense number of users who share their thoughts in many different communities. Having so many users allows for many different viewpoints. Thus, it leads to large datasets that offer valuable insight. In this project, we aim to predict and analyze controversy within reddit communities based on a user's comment by using Natural Language Processing and multi-modal Large Language Models. We will analyze the text through the LLM, while the images will be analyzed using LLaVA. We hope to gain valuable insights on how text and images contribute to controversy. We also want to predict which communities and topics will have more controversy.

II. DATASET

We have a dataset of Reddit comments with their respective subreddits. The entire dataset has a size of 292gb, a size a little too much for consumer grade hardware, so we've opted to get $\frac{1}{32}$ of the sample. We have to ensure that our sample is representative of the dataset's diversity to avoid bias through stratified sampling. The dataset is hosted by HuggingFace. [Dataset link](#)

The first column is named author and represents the author's name that made this comment. The second column is named body and it represents the text from the author. The next column is called controversy and is a binary value. The controversy attribute represents whether the comment is controversial or not but we will likely drop it since about 99% of the data has the value of 0. A 0 being not controversial and a 1 being controversial. created_utc is the next attribute within the dataset. This attribute represents the Unix timestamp when the comment was made by the author on Reddit. Link_id is also another column within

the dataset. Each comment is tied to a specific post. This attribute represents the unique id of that post.

Following this attribute, we have another attribute called score. score represents the net score of upvotes to downvotes that a comment has. We also have the attribute called subreddit. This refers to the name of the subreddit that this comment was made under. Subreddit_id represents the internal unique id for that subreddit. Each subreddit will have its own id. Finally, we have the last attribute which is called id. This is the unique id of this exact comment made by the author.

III. RELATED WORKS

We want to efficiently train the model on consumer hardware so we explored alternative methods of fine tuning pre-trained models. An alternative to vanilla fine-tuning that we've looked at is Low-Rank adaptation or LoRA, a type of Parameter-Efficient Fine Tuning. LoRA freezes the original weights and generates adapters that can be mixed and matches tasks as described by E. J. Hu *et al.* [1]. The freezing of model weights removes that hardware barrier to entry since we're only working with the low-rank matrices.

On a similar literature review, F. Zhao *et al.* [2] introduce RedOne, a domain specific LLM that is designed to break the performance bottleneck of single-task baselines and establish a foundation for social networking services. According to their research, the LLM was established through a three-stage optimization by using large-scale datasets. The results of the study showed that RedOne was able to reduce the exposure rate of harmful content by 11.23% and improve the click page rate in post-view search by 14.95%. The research aimed to inspire future research in developing specialized LLMs and advancing practical applications in social media.

Another revolutionary attempt to bring instruction tuning into multimodal domain is witnessed with the introduction of LLaVa (Large Language and Vision Assistant). H. Liu, C. Li, Q. Wu, and Y. J. Lee [3] highlights the methodology and empirical impact this model has by presenting benchmarks that evaluate instruction-following across different tasks. The researchers agree that although this is an initial step, it's a very substantial one as it helps move towards building a more capable multimodal assistants and while the

paper acknowledges the further need for deeper evaluation benchmarks, it clearly demonstrates that instruction tuning extends far beyond text and plays a crucial role in vision-language models.

IV. METHODS

We plan to evaluate and select a pre-trained LLM that allows us run and perform supervised fine-tuning at under 48GB of VRAM. Since we plan to also work with other forms of media such as images and videos, we'll need a way for our trained LLM to understand those. We plan to take advantage of multi-modal models. The multi-modal Large Language Model of our choosing will be Llava (Large Language and Vision Assistant). This model will be used to describe in text image and video content.

As stated in the introduction of our dataset, we'll perform stratified sampling of our dataset to avoid any biases.

We're fortunate to have access to a computer with 2 RTX 4090 GPUs. Each GPU packs 24GB of VRAM.

To perform fine-tuning, we plan to take advantage of QLoRA through Unsloth to efficiently train our model. To have our data set up for supervised fine-tuning, instruction is set somewhere along the lines of "Predict the upvote/downvote score for this post", input being the body of the content and its subreddit and the output as the upvote/downvote score and the model's best guess for a reason.

After training our model, we plan to export it in GGUF format through llama.cpp, which should allow us to use the model with Ollama and its API. Through Ollama's API, users will be able to prototype the model's capabilities through Streamlit.

A. Toolset

The entire project will rely on Python and its extensive libraries. For fine-tuning, we will leverage Unsloth by M. H. Daniel Han and U. team [4], which enables significantly faster training, up to 30x faster compared to conventional methods. To store and analyze our dataset as a dataframe, we will use Polars. It will also be used to perform stratified sampling. For deployment, we will utilize Ollama to run both the LLaVA model and our fine-tuned model.

V. EXPECTED OUTCOMES

The goal of this research is to detect how controversy occurs across Reddit's community and to further understand the contextual patterns that may indicate controversy. Because the dataset contains a mixture of common and popular subreddits, we anticipate predicting which communities tend to have more controversial discussions and also which topics gain most traction. By using Natural Language Processing (NLP), we anticipate discovering semantic markers associated with controversy.

The model is also expected to predict the relationship score between upvotes and downvotes and analyze whether controversial comments tend to have more polarized scores. For instance, it would be a possible indication that communities

value strong debates if the controversial comments have a higher score in some specific subreddits.

The "created_utc" timestamp is expected to analyze periods of controversies during significant events. The "author" field is expected to analyze if some users consistently post controversial content across the platform.

VI. MIDTERM PROGRESS

A. Pivot

This project has pivoted from predicting the upvote score of a post via LLM to predicting the intrinsic engagement and quality of a comment, relative to its originating subreddit by a fine-tuned RoBERTa model. This change was necessary due to the absence of a direct link between the comment data and the original posts, which removed the context required for accurate post-level prediction. The new goal is to classify a comment's relative popularity (Y) based purely on its linguistic features and community context (X). Specifically, we shift from a regression task (predicting the exact score) to multi-class classification, where the target variable are bins of upvotes (e.g., 'Controversial', 'Baseline', 'High Quality', 'Viral').

B. Loading the dataset

Due to the massive size of the dataset, our team does not have enough storage capacity to keep the entire dataset locally. Fortunately, HuggingFace's dataset library allows us to stream portions of the data. We selected 100,000 random rows from the dataset and excluded columns that were not relevant to our analysis: subreddit_id, id, created_utc, and controversy. Since the dataset is unsorted, these 100,000 samples are expected to be randomly distributed.

C. Exploratory Data Analysis

Examining the distribution of score values, we find that most posts in the dataset have only a single upvote, with no additional votes. Since this value dominates the data, a model trained without adjustment would likely predict a score of 1 for most inputs. To address this bias, we use the Python package `denseweight`, which assigns higher weights to more interesting data points. Incorporating these weights into the loss function allows us to use weighted MSE, helping the model focus on the more informative samples.

A subreddit's size is a factor in the number of upvotes a post could get, so

