Performing Sentiment Analysis on Tweets: An Analysis of Multiple Methods

Luca Mouchel, Colin Smyth, Frederik de Vries
CS-433 - Machine Learning - EPFL

***Abstract*—Although traditional machine learning methods have performed well over the years in several natural language processing tasks by analyzing sentiments from textual data, the advent of Large Language Models (LLMs) like BERT, GPT, and their variants has revolutionized this domain. In this project, we use these new methods to analyze the sentiment of a wide range of tweets. The project explores how these models, pre-trained on vast corpora, can be adapted to the specific linguistic styles and emotional expressions in tweets. Models were chosen based on performance using a small training and validation sample. Analysis was extended in two ways to include inference using Microsoft's Phi-2 LLM, and by using an ensemble of two models to reinforce results that lacked certainty from the initial approach. Our study shows that the highest performing model is the one using Ensemble Methods, which leverages two fine-tuned models and weighs their probabilites to make a prediction.**

I. Introduction

Large language models took the research community by storm in 2017 [17] with a new architecture: transformers. Transformers are designed to use a self-attention mechanism to draw global dependencies between input and output, which was very different from traditional architectures at the time, namely recurrent and convolutional neural networks. The self-attention mechanism, allows it to weigh the significance of different parts of the input data, providing a more effective way of capturing contextual information in text. The paper demonstrated the transformer's superior performance in translation tasks, setting new benchmarks in accuracy and efficiency. This led to many new models that have excelled at different tasks, namely BERT [3], GPT [1], LLaMa [16], and many more. These recent innovations have led to a revamp of classical NLP tasks such as sentiment analysis on Twitter. While many classifiers were developed using neural networks in the early 2010s, research has now reviewed these pipelines and implemented them using these transformer-based models, which outperform previous techniques. In fact, [9] finds that their proposed technique outperforms many word embedding techniques for sentiment analysis, achieving the highest results with close to 20% increase from the worst performing technique which was TF-IDF with SVM [9].

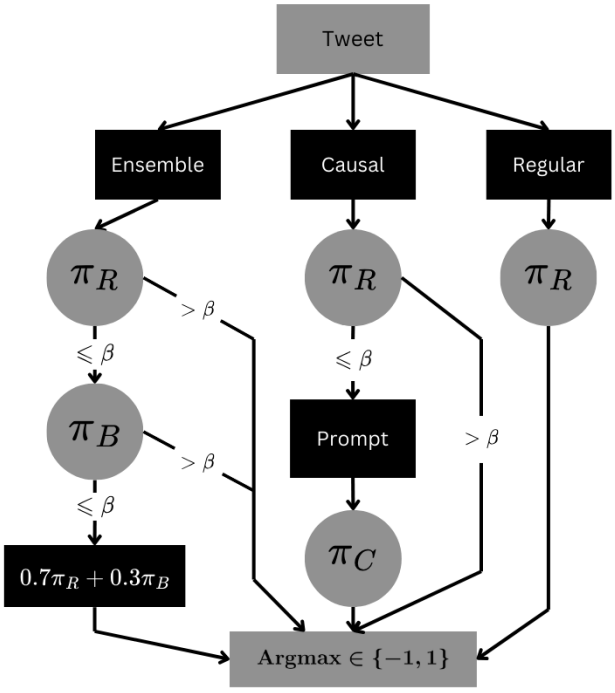

Fig. 1. Our three proposed classification frameworks. We leverage different optimization methods using Ensemble Methods, Prompting using causal models, and no optimization at all. We finetune models with the best performing one being R and the second best B. Using the ensemble method, we weigh the respective probabilities (π_R, π_B) if both models are not confident. Confidence is defined as $\Pr(\text{argmax } \pi.) > \beta$, with $\beta = 0.85$. Using prompting, we feed the causal model a prompt if our fine-tuned model is not confident and ask it to classify the tweet. π_C represents the probabilities of the causal model. More details in Section III.

Other papers have introduced models that are specifically trained on large datasets of tweets, enhancing their ability to interpret internet slang, emojis, and hashtag-

driven content, which are unique to social media discourse. Others have explored the integration of multimodal data, such as combining textual content with visual or auditory information present in tweets, to enrich sentiment analysis.

II. Related Works

Sentiment analysis on text has been an important task in many fields in recent years, including market research [15], feedback analysis [11, 12] and social media monitoring [4].

In particular, analysis of data from what was formerly known as Twitter can be used to understand the opinion of its users on social issues, brands, and or international events. In their paper, Sharma and Ghose used sentiment analysis on Twitter data to predict public opinions of two candidates running for the general election in India [14]. Two approaches to the analysis were taken in this paper: a lexicographic approach and a word-emotion dictionary-based approach. Both approaches give good results, with the latter giving a more detailed look into the sentiments expressed in the data.

Sentiment analysis branches out from the subject of text classification. In the last number of years, we have seen the development of transformer-based solutions to text analysis. The BERT model [3], and its numerous offshoots (DistilBERT, RoBERTa, etc.) have proven to be very useful in text classification tasks. These models take text and transform it into contextually relevant low-dimensional vectors. The model is then typically fine-tuned to learn how to categorize these vectors into the correct sentiment category. In our work, we make significant use of these transformer-based models for inference.

Finally, there are GPT-based solutions. The use of BERT has some drawbacks, namely that it is trained on masked data, and that it is from 2021 [6]. It is an LLM but it can be used as a classifier using a specific classification bit. GPTs are trained differently, as they do not mask random words from a sentence, but instead are trained to predict the next word in a sentence. This way of training introduces the concept of causality into the model. They, therefore, no longer need the classification bit. As they are trained to predict the next word, they can still be used as a classifier, given the right prompt.

III. Model and Methods

A. Our Data

In this project, we dive into tweet sentiment analysis (positive or negative sentiments) using a large corpus of tweets that have already been processed. See Table I for a set of examples from both classes.

Negative	rip meridith , we will miss u . dr . meridith marks's obituary by ottawa citizen : <url> <user> <user>
Positive	grateful today for a dream fulfilled ! ! my heart is so full - first 3 completed tracks have arrived back from new york ! #yeslord !

TABLE I
An example of a negative and positive tweets

The tweets in this corpus all previously contained either a positive or a negative smiley (☺ or ☹). Our goal is to predict the smiley that each tweet once contained. Naturally, sentiment analysis classification was chosen as a reliable method for this. Each tweet had already been pre-processed so that each token had been separated by a single whitespace.

We add an extra layer of normalization when processing tweets. We use the same normalization strategy as Dat Quoc Nguyen et al. [10], which essentially tokenizes and replaces strings that match a certain regex (e.g., replacing "ai n't" with "ain't").

B. Training and Hardware Configuration

Our dataset contains 2.5M tweets but because of our limited computing power, we train on a balanced randomized subset of our dataset. Table II summarizes our training configuration.

Train	Validation	Test
293K	80K	10K

TABLE II
Distribution of our train/test split

Moreover, we ran our training on 2 GPUs[1], which allowed for a speedup in the training process by several orders of magnitude compared to running the training without dedicated GPUs.

C. Model Selection

To choose an existing model to use as a base for our model, we conducted a testing process on several pre-trained models from the Hugging Face platform. The models chosen are shown in the ***Fine-tuned*** section in Table III.

[1]2 NVIDIA GeForce GTX TITAN X with 12Gb memory each

As fine-tuning on our full subset as shown in Table II would be too costly and time-consuming, we chose to train our models on an even smaller subset for model selection. We then took the best-performing models to train them with the full subset as shown in the table. The model selection subset consisted of 11000 positive and 11000 negative samples which was then randomly split into a train-test set with 80% as train data and the remaining 20% as testing. Each model was trained on the same data and was trained for 4 total epochs.

Model Name	Accuracy	F1 Score
Zero-Shot Classifier		
RoBERTa 124M Tweets	0.7102	0.6934
BART-MNLI	0.5	0.33
Fine-tuned		
FinancialBERT [5]	0.821	0.817
DistilBERT [13]	0.8267	0.8254
BERT [3]	0.8277	0.8264
BERTweet	**0.8764**	**0.874**
RoBERTa 58M Tweets	0.8612	0.8582
RoBERTa 124M Tweets [8]	**0.8694**	**0.8687**
Mixed		
RoBERTa 124M Tweets & Phi-2 [7, 8]	0.859	0.858

TABLE III
TESTING SCORES FOR MODELS

As expected, we saw significantly higher accuracy when using models previously trained on tweets, however other models performed relatively well. Notably, there was a significant improvement on the latest RoBERTa model from the zero-shot model to our fine-tuned model. We chose to continue working with this model, along with the BERTweet model.

D. Performance Optimization

Using a causal model: In an attempt to increase performance even further, we took one of the two chosen models and tried to enhance it using the Phi-2 LLM from Microsoft [7]. We chose this LLM as it is extremely recent, and also has a comparatively small size (2.7B parameters) which fits our computing capacity. This model still took a long time to perform inference (~ 15 seconds per classification), and as the performance of RoBERTa was already good, we decided to use a mixed approach.

We directly accepted the predictions that RoBERTa made with more than 85% confidence, and entered the remaining cases as input into Phi-2 using the following prompt:

> You are a sentiment classifier. What is the sentiment of {tweet}? Reply with one word: positive or negative.

However, as can be seen in Table III, we found that this resulted in a performance that was worse than the original model without Phi-2, so we did not explore this idea any further.

Ensemble Methods: Table III shows the best performing models are RoBERTa 124M Tweets (π_R), which is a fine-tuned model trained 124M tweets for sentiment analysis (Link). The other best-performing model is BERTweet (π_B), but Table IV shows π_B performed slightly worse on the test set. Because both models performed relatively well (>0.89 accuracy), we attempted to optimize the prediction on the test set by using both models. At prediction time, we iterate over batches and make the prediction with the best performing model (π_R), and use π_B as a support if our model is not confident. If the maximum probability of the prediction is less than 85%, that is, if $\Pr(\text{argmax}_y\ \pi_R(y)) < 0.85$. If this is the case, we then feed the input to π_B and if the corresponding probability is over 0.85, we use its prediction. Otherwise, we weigh both probabilities and predict the arg max of the resulting sequence. The prediction becomes

$$\text{Sentiment}(y) = \text{argmax}\big[0.7\pi_R(y) + 0.3\pi_B(y)\big]$$

where y is the input to the model (an individual tweet). This performs much better than simply returning the first fine-tuned model's predictions, with an accuracy of 0.899.

IV. RESULTS

A. Zero-shot prediction

To evaluate the performance of pre-existing sentiment analysis language models, we performed a zero shot prediction using the RoBERTa 124M Tweets (π_R) and BART-MNLI. For this, we used HuggingFace's `pipeline` tool which is easy to use. These models performed particularly poorly, and BART-MNLI is equivalent to random guessing, this is most likely because of the language of tweets and its more intricate vocabulary and slang language. In fact, π_R, which is already trained on 124M tweets performs 20% better than BART in the zero-shot condition. With the poor performance of these models, we did not do any other zero-shot classification as the performance was poor and we instead focused on fine-tuning language models on our tweets.

B. Fine-tuning a pre-trained LLM

Due to BERTweet and RoBERTa 124M being our most successful models in initial testing, separated by a fine margin, further fine-tuning was conducted on both models. The results of which can be seen in Table IV

Model Name	Accuracy	F1 Score
BERTweet	0.890	0.894
RoBERTa 124M Tweets [8]	0.896	0.899
Ensemble		
RoBERTa 124M + BERTweet	**0.899**	**0.901**

TABLE IV
MODEL RESULTS ON AICROWD

As seen from Table IV, using the ensemble method described in Section III-D our prediction accuracy increases by a small margin, by leveraging both fine-tuned models on the same data.

V. LIMITATIONS & FUTURE DIRECTIONS

A. Computational Power

One of the biggest limitations we faced during the training of proper models for our classification problem was computational power. Even with two powerful GPUs at our disposal, we had to significantly downsample the training set. Even with the reduced dataset (Table II), the transfer learning process took around 5 hours. The same limitation came into play when testing the GPT-based models. The reason we picked Microsoft's Phi-2 model compared to other open-source pre-trained LLMs was due to size constraints. The initial idea we had was to adapt these models to our use case, given out resources proved impractical. BERTweet, on which we successfully performed transfer learning, has 135M parameters [10]. We attempted transfer learning on XLM-R [2], another variant of RoBERTa, which performes better on tweet analysis [10], but we ran into issues because of insufficient memory. Retraining large GPT-based models would therefore be beyond our capabilities.

Moreover, because of the limited access to GPUs, we could not conduct additional experiments with the fine-tuned models to improve our performances. In fact, *Ensemble Methdods* usually combine several models to perform inference, but this would entail fine-tuning many different models for classification and using these when performing inference on the test set. As such, we decided to stick with using only two fine-tuned models.

B. Future Directions

Future directions could leverage larger language models, as recent works have shown the larger the models, the better the performance [19]. In fact, with more computing power, running larger language models would most likely improve accuracy by several percentage points. Moreover, by simply using GPT4 with the OpenAI API, our predictions using causal models with prompts would also increase given that GPT4 has 1.7 trillion parameters and Phi-2 has only 2.7B.

VI. ETHICAL CONSIDERATIONS

A. Relevance of Models

The models we are using are trained on tweets from 2021 and earlier. As we are unsure about when the tweets in our dataset were created, we risk using an outdated model, due to the rapid evolution of social media, and such the sentiments that the model knows may have changed. To mitigate this risk, we chose the model trained on the most recent set of tweets from the models that were available to us. The updated RoBERTa model was consistently retrained on new tweets every three months for two years after its creation to keep it relevant [8].

B. Energy Consumption

The energy consumption associated with GPU usage and AI in general is huge [18]. During this project, we remained vigilant of our use of GPUs and ensured that any large training batches were only executed when necessary.

C. Confidentiality and Anonymity

Throughout this project, the identities of the publishers of all tweets, and the identities of those mentioned in the tweets remained unknown to us, thus anonymity was preserved. However, as this is public data there is a possibility that the identity of the publisher could be found through reverse searching on Twitter.

VII. CONCLUSION

In this project, we successfully performed sentiment analysis on a large dataset of tweets. We found that Large Language Models such as BERT and roBERTa performed exceptionally well, and combining these methods with supports from Microsoft's Phi-2 model gave a very satisfactory result. Given greater computing power and additional time, an analysis could have been conducted with a larger GPT-based model such as LLaMa2 or the cutting-edge GPT-4. With the constraints that we were faced with, we are very content with the final accuracy achieved.

References

[1] Tom Brown et al. "Language models are few-shot learners". In: *Advances in neural information processing systems* 33 (2020), pp. 1877–1901. arXiv: 2005.14165.

[2] Alexis Conneau et al. *Unsupervised Cross-lingual Representation Learning at Scale*. 2020. arXiv: 1911.02116.

[3] Jacob Devlin et al. "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding". In: *CoRR* abs/1810.04805 (2018). arXiv: 1810.04805.

[4] Zulfadzli Drus and Haliyana Khalid. "Sentiment Analysis in Social Media and Its Application: Systematic Literature Review". In: *Procedia Computer Science* 161 (2019). The Fifth Information Systems International Conference, 23-24 July 2019, Surabaya, Indonesia, pp. 707–714. ISSN: 1877-0509. DOI: 10.1016/j.procs.2019.11.174.

[5] Ahmed Hazourli. "FinancialBERT - A Pretrained Language Model for Financial Text Mining". In: (Feb. 2022). DOI: 10.13140/RG.2.2.34032.12803.

[6] Martin Jaggi. *Lecture notes in Self-supervised Learning*. 2023. URL: https://github.com/epfml/ML_course/blob/master/lectures/13/lecture13a_self_supervised.pdf.

[7] Mojan Javaheripi and Sébastien Bubeck. *Phi-2: The surprising power of small language models*. Dec. 20223. URL: https://www.microsoft.com/en-us/research/blog/phi-2-the-surprising-power-of-small-language-models/.

[8] Daniel Loureiro et al. "TimeLMs: Diachronic Language Models from Twitter". In: *CoRR* abs/2202.03829 (2022). arXiv: 2202.03829.

[9] Usman Naseem et al. "Transformer based Deep Intelligent Contextual Embedding for Twitter sentiment analysis". In: *Future Generation Computer Systems* 113 (2020), pp. 58–69. ISSN: 0167-739X. DOI: 10.1016/j.future.2020.06.050.

[10] Dat Quoc Nguyen, Thanh Vu, and Anh Tuan Nguyen. "BERTweet: A pre-trained language model for English Tweets". In: *arXiv preprint arXiv:2005.10200* (2020). arXiv: 2005.10200.

[11] Pankaj et al. "Sentiment Analysis on Customer Feedback Data: Amazon Product Reviews". In: *2019 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COMITCon)*. 2019, pp. 320–322. DOI: 10.1109/COMITCon.2019.8862258.

[12] Aksh Patel, Parita Oza, and Smita Agrawal. "Sentiment Analysis of Customer Feedback and Reviews for Airline Services using Language Representation Model". In: *Procedia Computer Science* 218 (2023). International Conference on Machine Learning and Data Engineering, pp. 2459–2467. ISSN: 1877-0509. DOI: 10.1016/j.procs.2023.01.221.

[13] Victor Sanh et al. "DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter". In: *NeurIPS EMC^2 Workshop*. 2019. arXiv: 1910.01108.

[14] Ankita Sharma and Udayan Ghose. "Sentimental Analysis of Twitter Data with respect to General Elections in India". In: *Procedia Computer Science* 173 (2020). International Conference on Smart Sustainable Intelligent Computing and Applications under ICITETM2020, pp. 325–334. ISSN: 1877-0509. DOI: 10.1016/j.procs.2020.06.038.

[15] Hamed Taherdoost and Mitra Madanchian. "Artificial Intelligence and Sentiment Analysis: A Review in Competitive Research". In: *Computers* 12.2 (2023). ISSN: 2073-431X. DOI: 10.3390/computers12020037.

[16] Hugo Touvron et al. "Llama: Open and efficient foundation language models". In: (2023). arXiv: 2302.13971.

[17] Ashish Vaswani et al. "Attention is all you need". In: *Advances in neural information processing systems* 30 (2017). arXiv: 1706.03762.

[18] Alex de Vries. "The growing energy footprint of artificial intelligence". In: *Joule* 7.10 (2023), pp. 2191–2194. ISSN: 2542-4351. DOI: 10.1016/j.joule.2023.09.004.

[19] Ruiqi Zhong et al. *Are Larger Pretrained Language Models Uniformly Better? Comparing Performance at the Instance Level*. 2021. arXiv: 2105.06020.