SDS384

Scientific Machine Learning

Professor Arya Farahi

University of Texas, Spring 2023

Final Project

Title: Sentiment Analysis

Date: April 26th 2023

By:
Taylor Cox - tec989
Zexi Zhou - zz7277
Liaoyi Xu - lx2755

Table of Contents

Introduction

The World Health Organization declared Coronavirus, or COVID-19, a global pandemic on March 11, 2020. WHO's announcement was quickly followed by the Trump administration issuing a nationwide emergency and extending social distancing measures through April 2020. As US states subsequently implemented lockdown measures, many people living in the US took to Twitter to voice their feelings about the pandemic and pandemic protocols. Twitter became a space for people to express frustrations and joys they encountered during quarantine and also to communicate information and ideas relating to COVID-19.

Sentiment analysis studies often use social media as a source of data to explore the public's reaction to policy. Many studies use Twitter in particular, since Twitter data is widely available, can be easily categorized using hashtags, and includes information about a user's location and the date a Tweet is published. Tweet text is also especially useful since it is often short, and Twitter itself provides a common forum for emotionally charged expression.

This study seeks to use sentiment analysis of Twitter data relating to the COVID-19 pandemic to explore public perception of COVID-19 and the possible predictors of public perception. Prior studies have begun this exploration, and this study will contribute to existing literature by including a Bidirectional Encoder Representations from Transformers (BERT) model for text classification and evaluating its performance against a series of other machine learning models.

COVID-19 is a global issue with long-lasting and far-reaching implications. Understanding how the public perceived the pandemic and related policies can help policymakers better understand people's mental health during quarantine and how people felt about the measures that were put in place to combat the pandemic.

Literature Review

Many prior studies utilize Twitter as a source of data to analyze public perception of various policies.[1] Several studies identifiy Twitter as a space that enables policymakers to explore popular discourse about the COVID-19 pandemic in particular. Christian E. Lopez, Malolan Vasu, and Caleb Gallemore in *Understanding the Perception of COVID-19 Policies by Mining a Multilanguage Twitter Dataset* use Natural Language Processing, Text Mining, and Network Analysis to identify common responses concerning COVID-19 on Twitter.[2] They propose using similar methods to explore how misinformation was transmitted on Twitter in January 2020, at the onset of the pandemic, and track how public perception continued to change over time.

[1] Giachanou, Anastasia, and Fabio Crestani. "Like it or not: A survey of twitter sentiment analysis methods." *ACM Computing Surveys (CSUR)* 49, no. 2 (2016): 1-41.

[2] Lopez, Christian E, Malolan Vasu, and Caleb Gallemore. "Understanding the Perception of COVID-19 Policies by Mining a Multilanguage Twitter Dataset." https://doi.org/10.48550/arXiv.2003.10359 (2020).

Several other studies have applied BERT-based models to classify text relating to public perception, such as tweets.[3] Because this kind of data is often noisy due to the presence of unnecessary words, or even emoticons, BERT is especially useful in sentiment analysis contexts.[4] Rukhma Qasim, Waqas Haider Bangyal, Mohammed A. Alqarni, and Abdulwahab Ali Almazroi in *A Fine-Tuned BERT-Based Transfer Learning Approach for Text Classification* use BERT-based models for text classification on various datasets, including a COVID-19 fake news dataset, a COVID-19 English tweet dataset, and an extremist/ non-extremist dataset containing news blogs, posts, and tweets related to coronavirus and hate speech.[5] They find that BERT-base and BERT-large models applied to the COVID-19 English tweet dataset resulted in a 98.44% accuracy score, which was only exceeded by XLM-RoBERTa and Transfer learning models.

U. N. Wisesty, R. Rismala, W. Munggana and A. Purwarianti in *Comparative Study of Covid-19 Tweets Sentiment Classification Methods* conduct a study similar to ours. They analyze the same dataset used in our study using three schemes: the vector space model (Bag of Words and TF-IDF) with Support Vector Machine, word embedding (word2vec and Glove) with Long Short-Term Memory, and BERT.[6] They find that BERT outperforms the other two schemes and achieves the highest F-1 score. Additionally, they find that their algorithm performs better at classifying texts into three classes, negative, neutral, and positive, as opposed to five classes, which includes extremely negative and extremely positive.

Dataset

The dataset for this study is called “Coronavirus tweets NLP - Text Classification” and comes from Kaggle.[7] The tweets have been pulled from Twitter and manually categorized into five sentiment categories: extremely negative, negative, neutral, positive, and extremely

[3] Al-Garadi, Mohammed Ali, Yuan-Chi Yang, Sahithi Lakamana, and Abeed Sarker. "A text classification approach for the automatic detection of twitter posts containing self-reported covid-19 symptoms." (2020); for studies exploring how BERT can be applied to non-English language text classification, see Chavan, Tanmay, Shantanu Patankar, Aditya Kane, Omkar Gokhale, and Raviraj Joshi. "A Twitter BERT Approach for Offensive Language Detection in Marathi." *arXiv preprint arXiv:2212.10039* (2022) and Alammary, Ali Saleh. "BERT models for Arabic text classification: a systematic review." *Applied Sciences* 12, no. 11 (2022): 5720.

[4] Devlin, Jacob, Chang Ming-Wei, Kenton Lee, and Kristina Toutanova. “BERT: Pre-Training of Deep Bidirectional Transformers for Language Understanding.” *arXiv.org* (2019).

[5] Qasim R, Bangyal WH, Alqarni MA, Ali Almazroi A. A Fine-Tuned BERT-Based Transfer Learning Approach for Text Classification. J Healthc Eng. 2022 Jan 7;2022:3498123. doi: 10.1155/2022/3498123. PMID: 35013691; PMCID: PMC8742153.

[6] U. N. Wisesty, R. Rismala, W. Munggana and A. Purwarianti, "Comparative Study of Covid-19 Tweets Sentiment Classification Methods," *2021 9th International Conference on Information and Communication Technology (ICoICT)*, Yogyakarta, Indonesia, 2021, pp. 588-593, doi: 10.1109/ICoICT52021.2021.9527533.

[7] For another study using this dataset which improves upon the initial study conducted by U. N. Wisesty, R. Rismala, W. Munggana and A. Purwarianti in *Comparative Study of Covid-19 Tweets Sentiment Classification Methods*, see S. S. Ayon, S. Ishrat, S. A. Mallick, P. Chandra Das and F. B. Ashraf, "Sentiment Analysis on COVID-19 Tweets," *2022 25th International Conference on Computer and Information Technology (ICCIT)*, Cox's Bazar, Bangladesh, 2022, pp. 551-556, doi: 10.1109/ICCIT57492.2022.10055015.

positive. Each *n* observation corresponds to a tweet and includes the date, location, user ID, and tweet text. The dataset has already been divided into a training and test set.

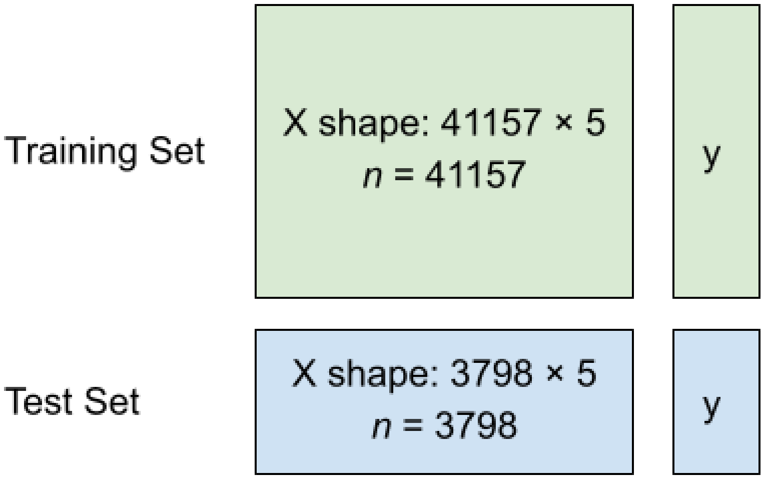


Figure 1: Dataset shape

Many of the observations included tweet text that used emoticons. These “emojis” had to be eliminated from the tweets for easier text classification, as did special characters. The date, location, user ID were also excluded from our dataset, since only the tweet text was necessary to train our model.

Exploratory Analysis and Generating Hypothesis

Although we did not use tweet date in our model to predict sentiment analysis, visualizing the number of tweets by date is helpful to understand the dataset and shows how activity on Twitter spiked in response to policies implemented to combat COVID-19.

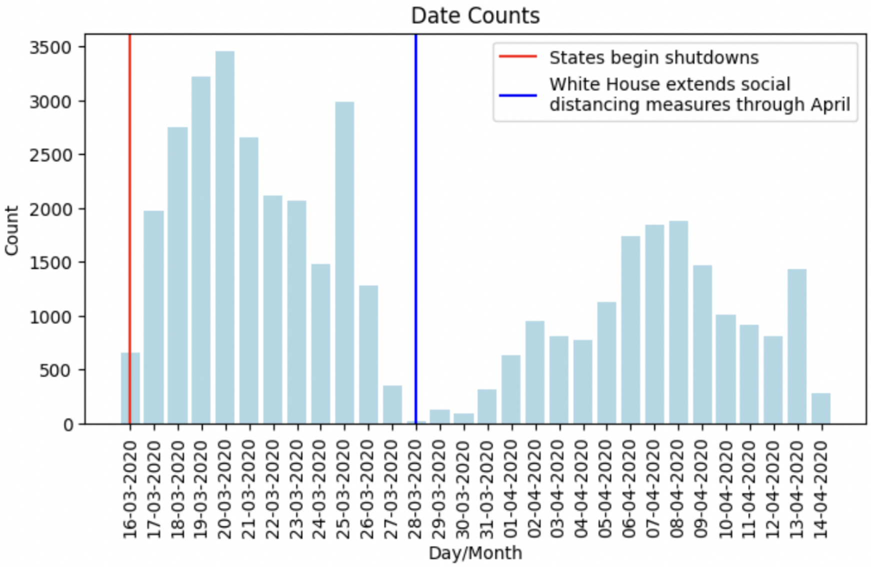


Figure 2: Number of Tweets by Date

Figure 2 shows that Twitter activity spiked after states began implementing lockdown measures in mid-March. A smaller spike occurred later in March when the White House announced plans to extend social distancing and other COVID-19 policies through April.

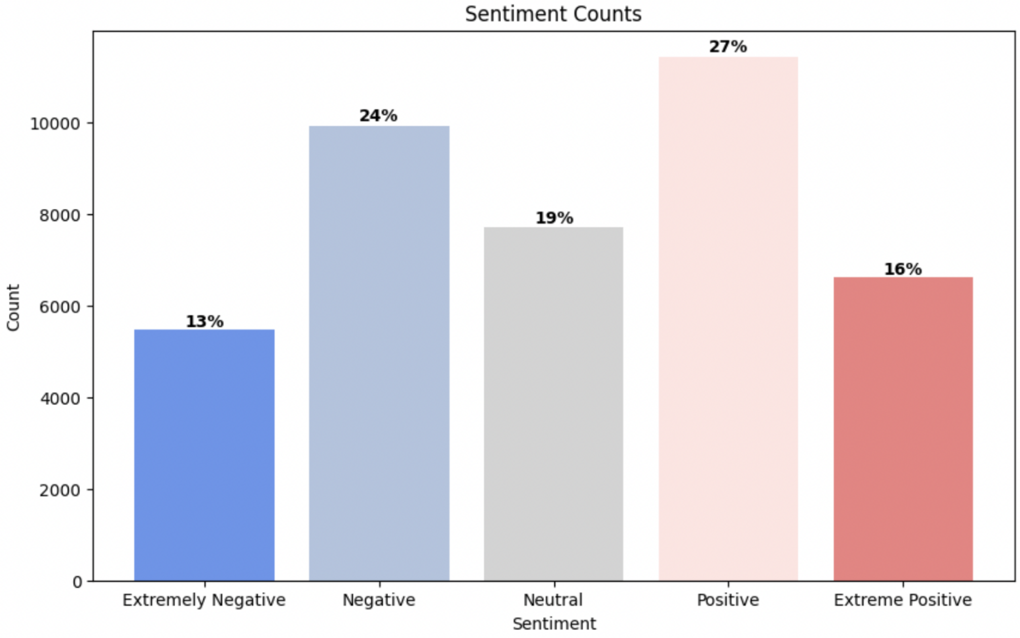


Figure 3: Number of Tweets by Sentiment

The number of tweets is fairly evenly distributed by sentiment, with “negative” and “positive” coded tweets accounting for slightly over half of the dataset. Tweets coded as “extremely negative” or “extremely positive” are rarer.

The length of the tweets by sentiment follow normal distributions, except for tweets coded as “neutral.” Tweets coded as “extremely negative” or “extremely positive” tend to be longer than tweets coded as “negative” or “positive.” The distribution of the length of tweets coded as “neutral” has a right-skew, as these tweets tend to be shorter. These trends are logical, since we might expect emotionally-charged “extremely negative” or “extremely positive” tweets to be longer than “neutral” tweets, which may instead be intended to convey information.

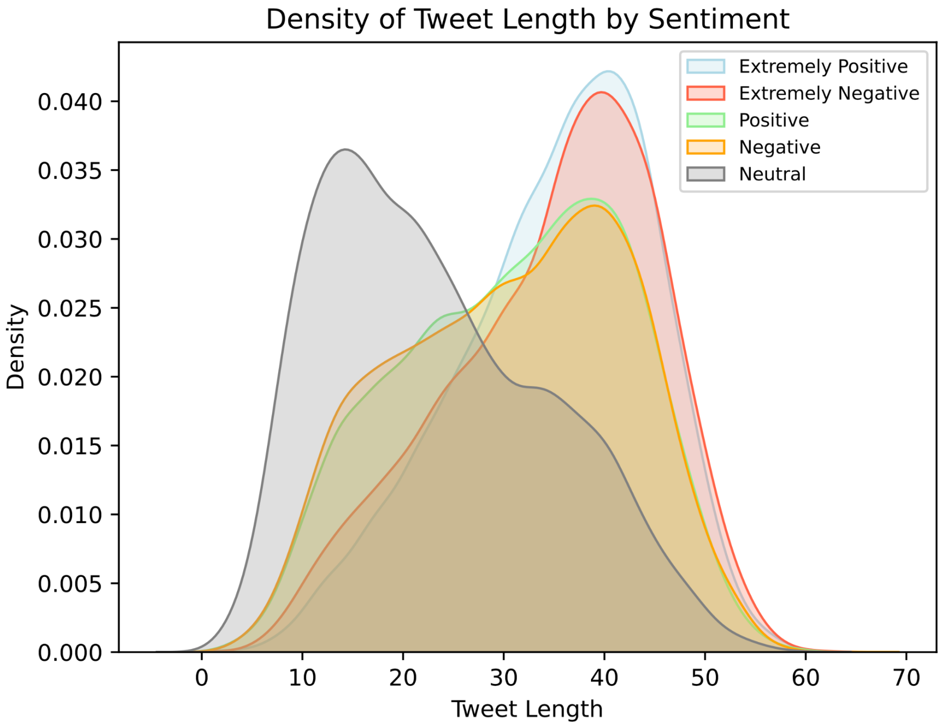


Figure 4: Density of Tweet Length by Sentiment

Modeling and Validation

Model Comparison

We first conducted data cleaning on the original tweets to remove any potential hindrances to the model's performance, such as emojis, punctuations, and special characters like & and $. Subsequently, we utilized various vectorizer functions from the scikit-learn library, including TfidfVectorizer, HashingVectorizer, and CountVectorizer, to process the cleaned tweet text in the training set.

After vectorizing the tweet text, we employed multiple machine learning classifiers, including Logistic Regression, K-Nearest Neighbors, Random Forest, Decision Tree, Naive Bayes, LightGBM, Gradient Boosting, XGBoost, CatBoost, and AdaBoost. We then fed the cleaned training set to each of these classifiers. Following the training process with default parameters, we utilized the models to generate predictions on the testing set.

To evaluate the performance of the various models, we employed five key metrics: AUC, precision, accuracy, recall, and F1-score. Among all the tested classifiers, Logistic Regression demonstrated the best performance across all five metrics, as illustrated in Figure 5.

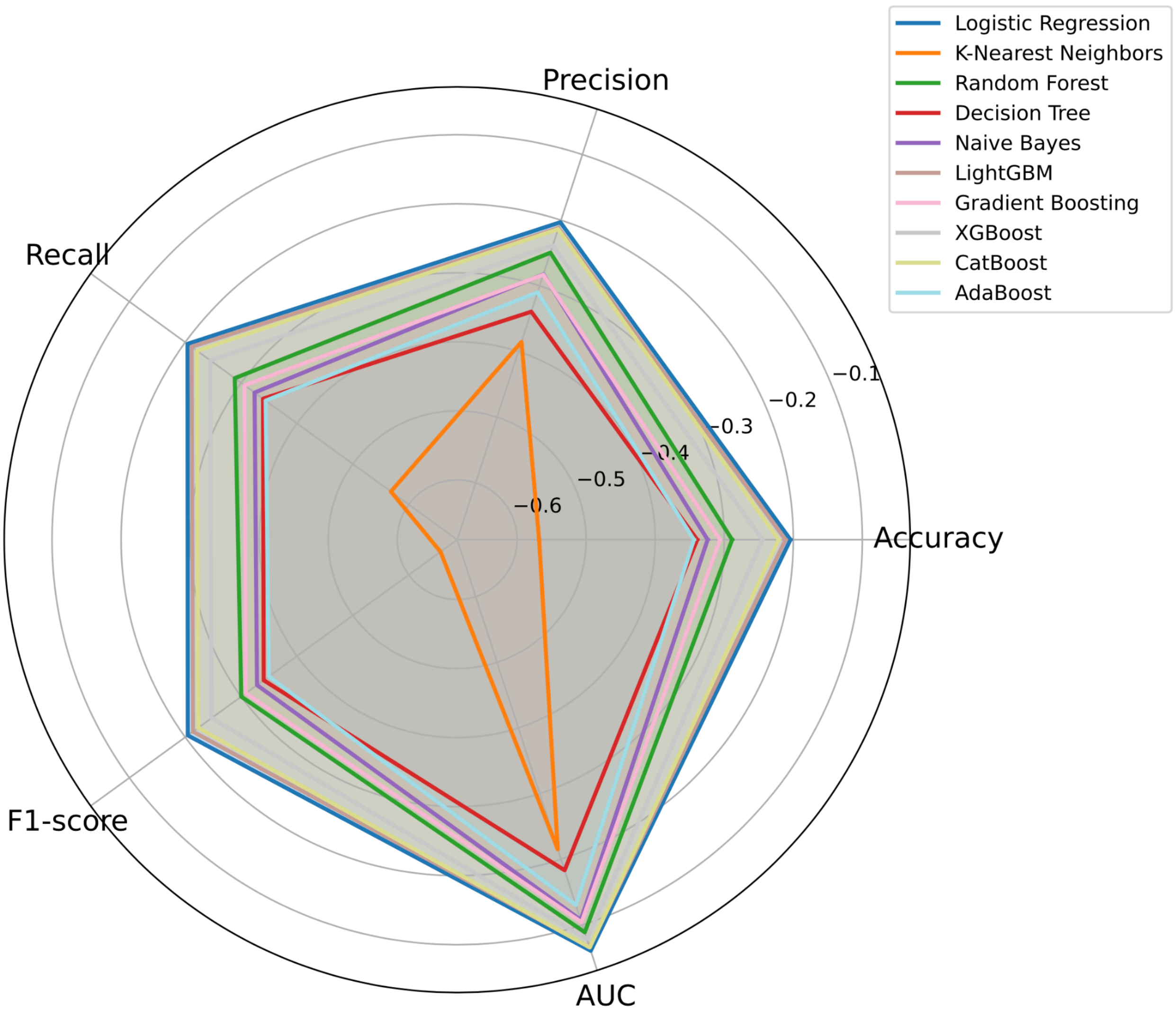


Figure 5: Comparison of various machine learning classifiers across five metrics (with log10-transformed axes, i.e., Log10 (Metrics score))

Hyperparameters Tuning

As Logistic Regression outperformed all other machine learning classifiers tested, we sought to optimize it further by tuning its hyperparameters in order to achieve the highest possible F1-score. Specifically, we conducted a grid search on three parameters: C (with values of 0.1, 1, and 10), penalty (either 'l1' or 'l2'), and max_iter (with values of 100, 500, and 1000), as demonstrated in Figure 6A.

Ultimately, we discovered that the combination of CountVectorizer and hyperparameters {C: 1, penalty: 'l1', max_iter: 1000} yielded the highest F1-weighted score, approximately 0.67. This result represented an improvement of around 0.5 from the default parameters, as depicted in Figure 6B.

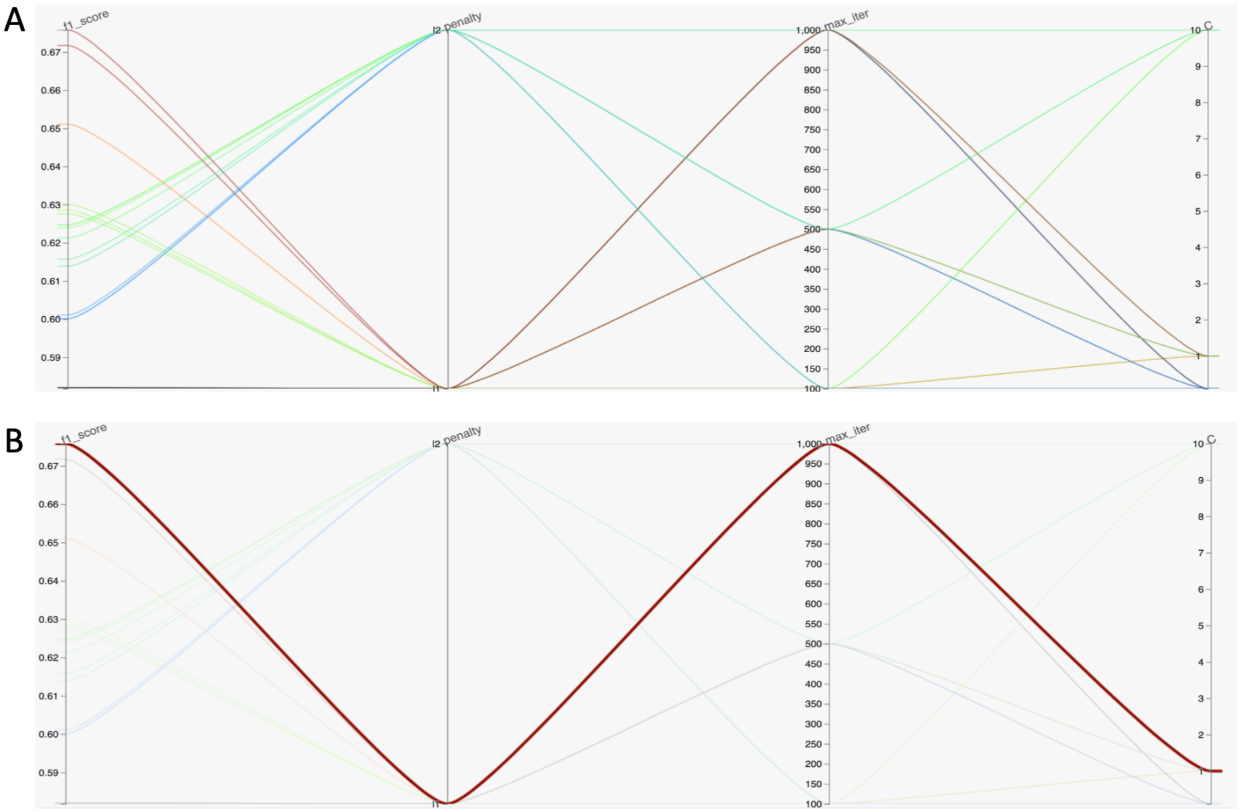


Figure 6: Hyperparameters Tuning. Figure 3A displays all the hyperparameter combinations we tested for Logistic Regression. Figure 3B highlights the combination of hyperparameters that yielded the highest F1-weighted score.

BERT

Machine learning models, as we've employed thus far, rely solely on the frequency of each word for training purposes. However, the order of words also plays a crucial role in understanding sentiment. To address this issue, we incorporated a pre-trained BERT model from Hugging Face, which is designed to capture the contextual information of words and their sequence within a given text, thus providing a more comprehensive understanding of sentiment.

Import required packages:

```
from datasets import Dataset
from transformers import AutoTokenizer, AutoModelForSequenceClassification,
TrainingArguments, Trainer, EarlyStoppingCallback
import tensorflow as tf
import torch
from torch.nn.functional import softmax
```

We first convert pandas dataframe to Hugging Face dataset:

```
hg_train_data = Dataset.from_pandas(df_train)
hg_test_data = Dataset.from_pandas(df_test)
```

Then we load tokenizer from pretrained model and tokenizer our dataset:

```
tokenizer = AutoTokenizer.from_pretrained("bert-base-cased")
tokenizer
```

```
BertTokenizerFast(name_or_path='bert-base-cased', vocab_size=28996, model_max_length=512,
is_fast=True, padding_side='right', truncation_side='right', special_tokens={'unk_token': '[UNK]',
'sep_token': '[SEP]', 'pad_token': '[PAD]', 'cls_token': '[CLS]', 'mask_token': '[MASK]'})
```

```
# Funtion to tokenize data
def tokenize_dataset(data):
    return tokenizer(data["text"],
                     max_length=64,
                     truncation=True,
                     padding="max_length")

# Tokenize the dataset
dataset_train = hg_train_data.map(tokenize_dataset)
dataset_test = hg_test_data.map(tokenize_dataset)
```

Load pretrained BERT model:

```
model =
AutoModelForSequenceClassification.from_pretrained("bert-base-cased",
num_labels=5)
```

Set up training arguments:

```
training_args = TrainingArguments(
    output_dir="./results/",
    logging_dir='./results/logs',
    logging_strategy='epoch',
    logging_steps=100,
    num_train_epochs=10,
    per_device_train_batch_size=64,
    per_device_eval_batch_size=2,
    learning_rate=5e-6,
    seed=42,
    save_strategy='epoch',
    save_steps=100,
    evaluation_strategy='epoch',
    eval_steps=100,
    load_best_model_at_end=True
)
```

Define a metrics allow to evaluate model’s performance so that can early stop at best model:

```
def compute_metrics(eval_pred):
    logits, labels = eval_pred
    predictions = np.argmax(logits, axis=1)
    f1 = f1_score(labels, predictions, average='weighted')
    return {"f1": f1}
```

Train the model:

```
trainer = Trainer(
    model=model,
    args=training_args,
    train_dataset=dataset_train,
    eval_dataset=dataset_test,
    compute_metrics=compute_metrics,
    callbacks=[EarlyStoppingCallback(early_stopping_patience=10)]
)

trainer.train()
```

During the training process of the BERT model, we closely monitored its F1-weighted score, as shown in Figure 7. Interestingly, Figure 7 reveals that the pre-trained BERT model outperformed the fine-tuned Logistic Regression model even after just a single epoch of training. This demonstrates the effectiveness of the BERT model in capturing the contextual information and word order for sentiment analysis.

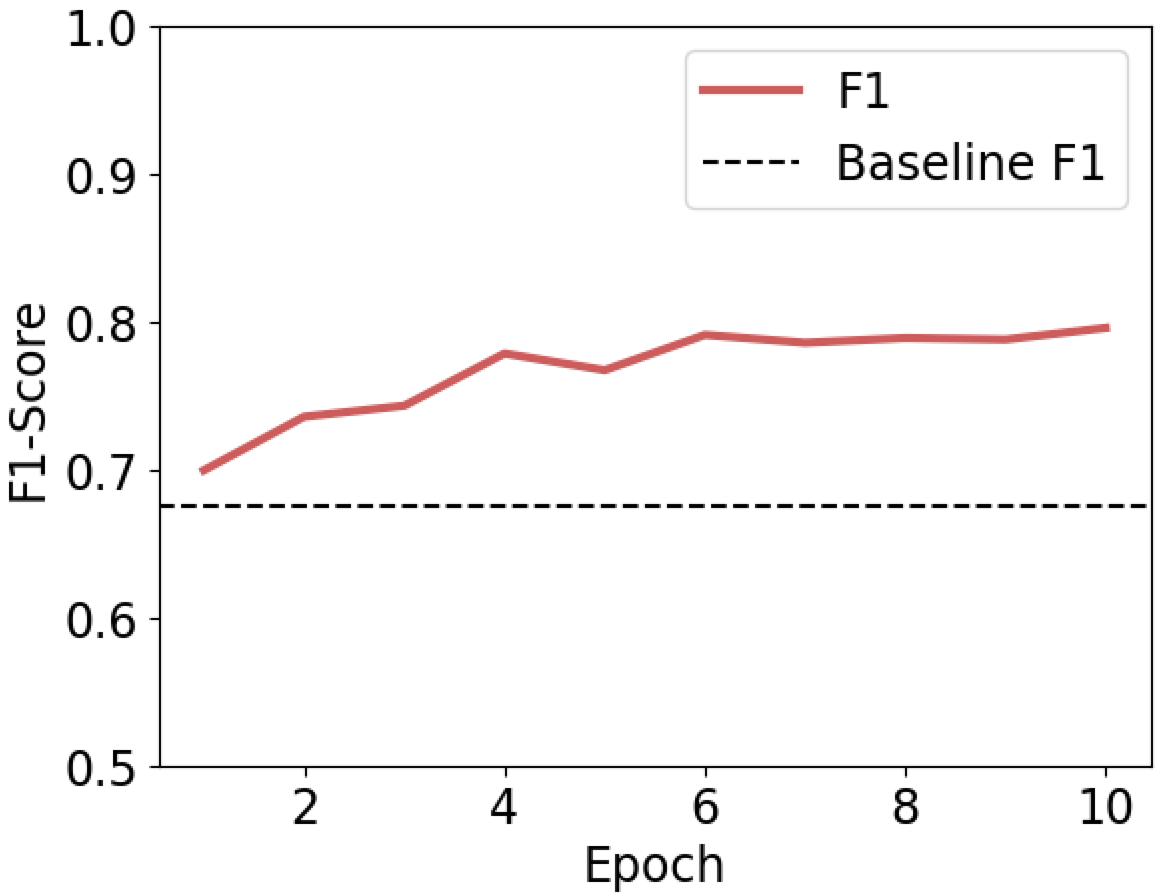


Figure 7: BERT Performance on Sentiment Classification During Training. The red line represents the F1-score of the BERT model, while the black dashed line indicates the F1-score of the Logistic Regression model. The graph illustrates the superior performance of the BERT model in comparison to Logistic Regression for sentiment analysis.

After model training finished, we saved the model and tokenizer:

```
# Save tokenizer
tokenizer.save_pretrained('/content/drive/MyDrive/sentiment_transfer_learning_transformer')

# Save model
trainer.save_model('/content/drive/MyDrive/sentiment_transfer_learning_transformer')
```

Then we applied trained BERT model to make predictions on testing data:

```
df_test['PredSent'] = df_test['OriginalTweet'].apply(lambda x: analyze_sent(x)[0])
```

In the end, we compared the performance of the BERT model to that of the fine-tuned Logistic Regression model. As illustrated in Figure 8, the F1-score achieved by the BERT model significantly surpasses the F1-score obtained from the Logistic Regression model, emphasizing the superiority of the BERT model for sentiment analysis tasks.

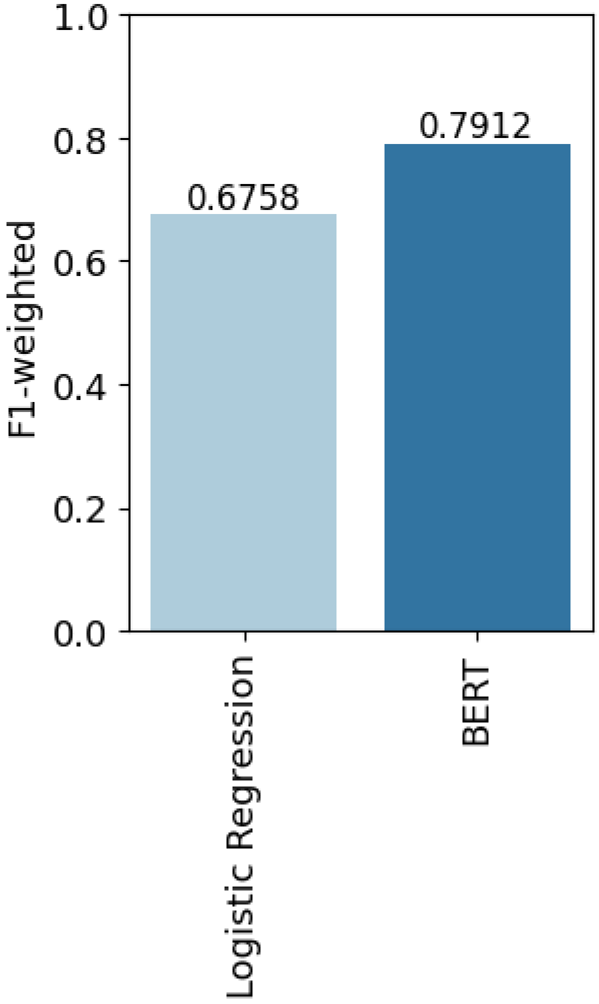


Figure 8: Model comparison of fine-tuned logistic regression and BERT

We further analyzed the BERT model's performance by plotting a confusion matrix (Figure 9). According to Figure 9, the model performs well overall, but it struggles to distinguish between Negative/Positive and Extremely Negative/Positive sentences. This is understandable, as even humans can have difficulty differentiating between these two categories of sentiment.

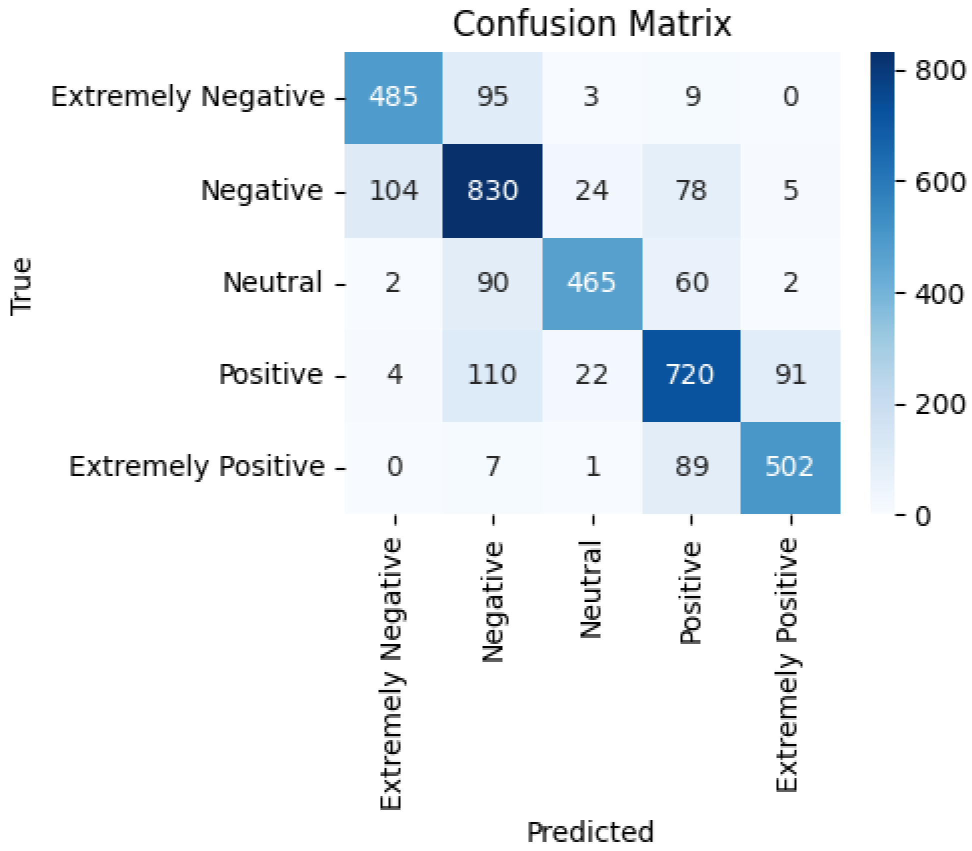


Figure 9: confusion matrix for BERT.

Model Interpretation with SHAP

Based on the fine-tuned BERT model, we interpreted and model and understood how it arrived at its predictions using SHapley Additive exPlanations (SHAP). We randomly selected 10 tweets on March 16 2020 (i.e., the states began shutdowns) and on April 14 2020 (i.e., one month later), and calculated the SHAP values for each original tweet text. Once the SHAP values were computed, we then visualized the feature attributions towards each sentiment category using the average over 10 tweets at each of the two time points. Figure 10 shows the top words impacting the category “Negative” and “Positive” for the 10 tweets on March 16 2022. Some negative tweets keywords include “rebel”, “panic”, “empty”, “shortage”, “paper”, etc. Some positive tweets keywords include “hand”, “dedicated”, “thank”, “adequate”, “ready”, “safe”, etc. Figure 11 shows the top words impacting the category “Negative” and “Positive” for the 10 tweets on April 14 2022. Some negative tweets keywords include “lower”, “crazy”, “falling”, “struggling”, “laming”, etc. Some positive tweets keywords include “good”, “free”, “worthy”, “hand”, “help”, etc.

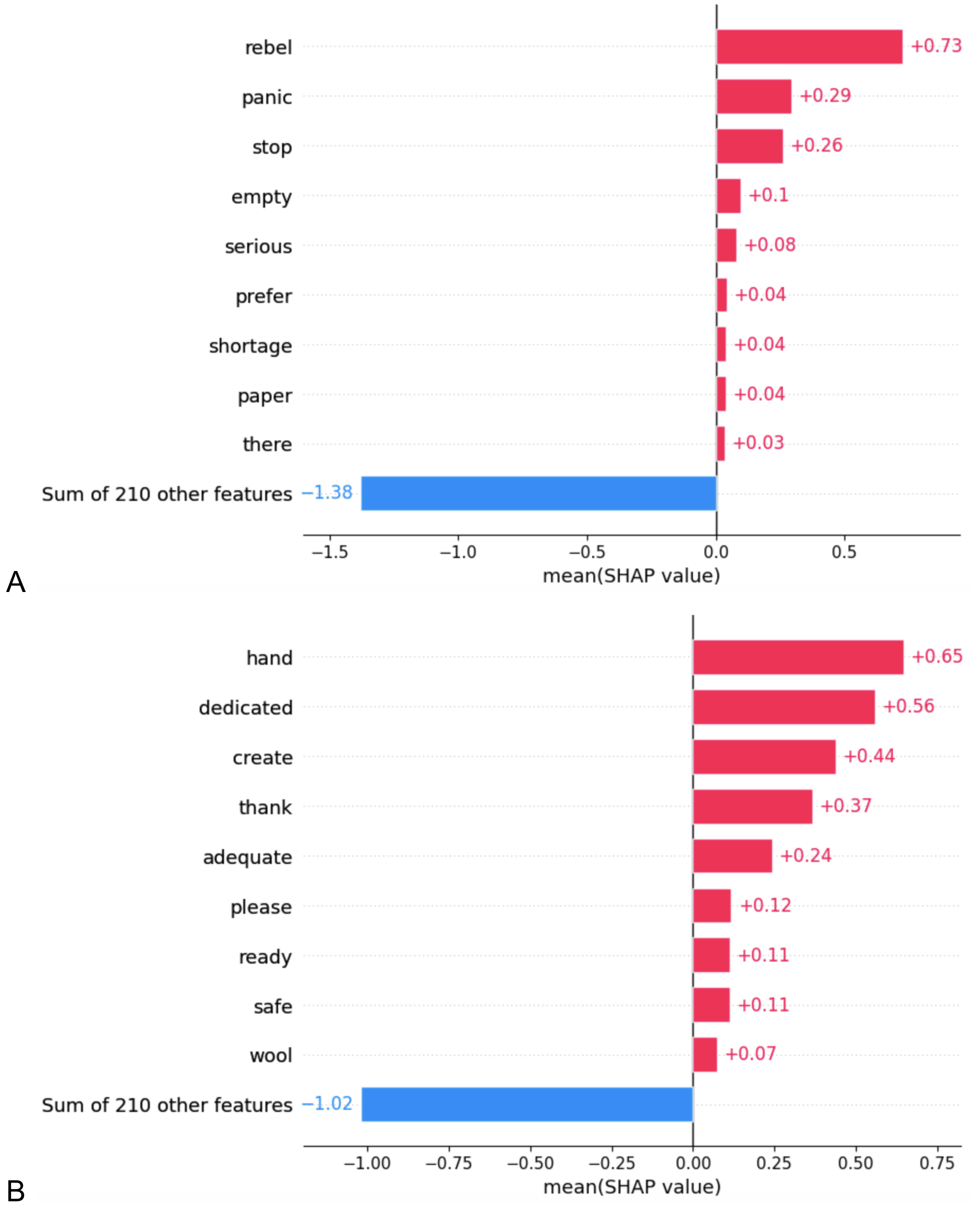


Figure 10: Top words impacting the category “Negative” (Figure 10A) and “Positive” (Figure 10B) on 2020-03-16

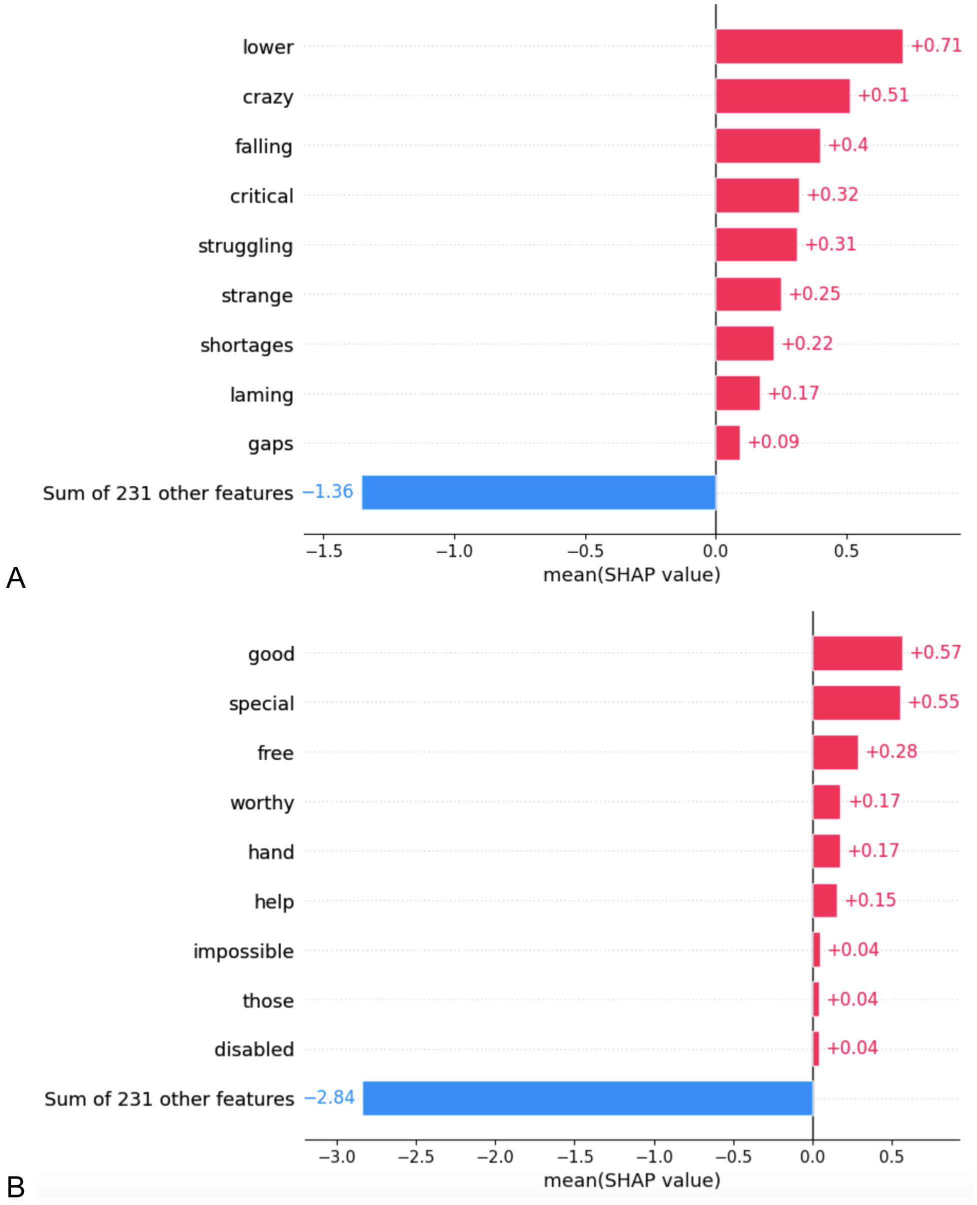


Figure 11: Top words impacting the category “Negative” (Figure 11A) and “Positive” (Figure 11B) on 2020-04-14

Broadly speaking, some keywords emerged in mid-March seem to be around the panic and resource shortages that may be caused by the shutdown policy, whereas some new keywords showed up in mid-April seem to be around the severity of infection and feelings after a long-time shutdown. As shown in Figure 12, the relative proportion of the five sentiment categories of tweets seem to remain stable within one month. That is, there does not seem to be an observable change in the overall public emotional attitudes toward Covid-19. Yet, according to the SHAP results, the specific topic and content of public concern, as well as why they felt positive or negative toward the pandemic, may have already changed over the period of one month.

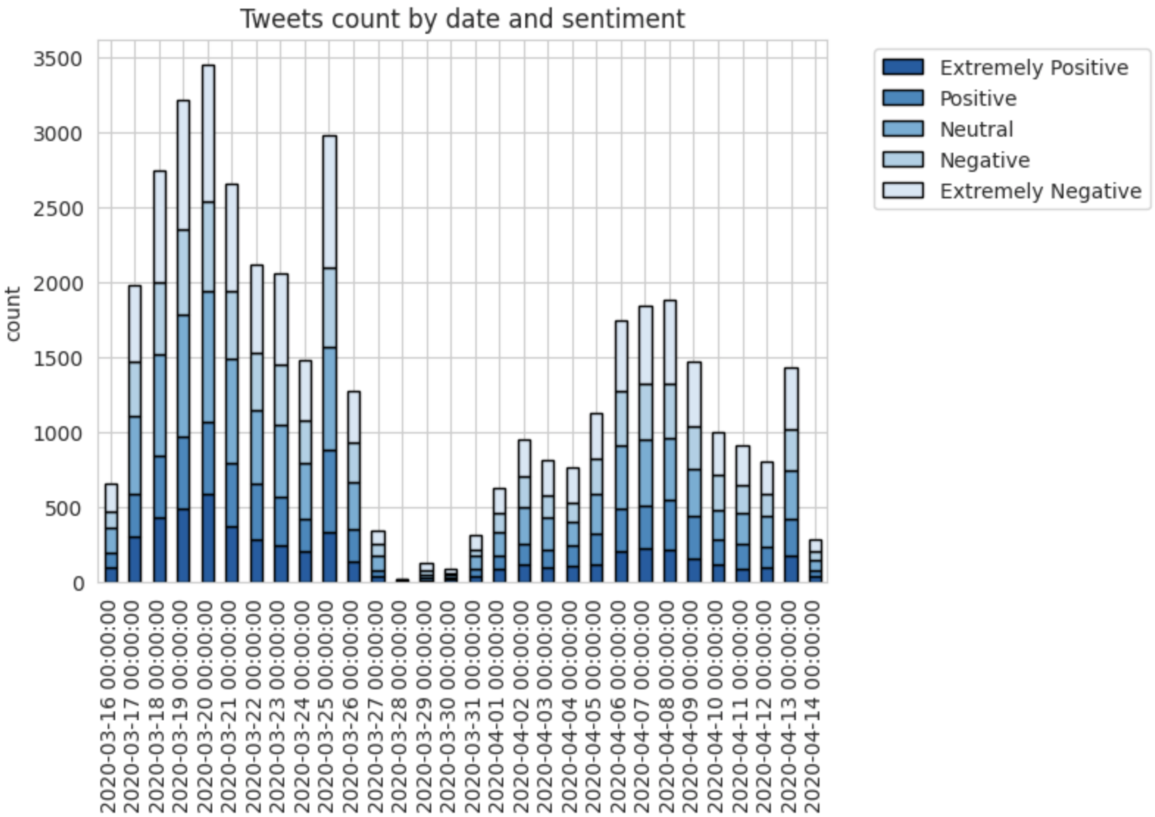


Figure 12: Tweets count by date and sentiment

In addition, we used SHAP to understand the model validation besides the metrics. Specifically, the model occasionally misclassifies positive/negative sentences as negative/positive. Upon closer examination, we found that some of the sentences were indeed incorrectly predicted by the model, while others appeared to have incorrect annotations. For example, below is a tweet that has the annotation as “Positive” but predicted as “Extremely Negative” by our model:

“i dont rly understand the stock market nor do i ideologically believe it matters at this point i understand itll affect other things but what matters rn is peoples healththeir ability to access healthcare food safety etc but i think this is bad coronavirus.”

Figure 13A shows how our model made its prediction. The words that contributed the most in order were “bad”, “but”, “ideologically”, “matters”, etc. There were also words that went against the prediction such as “safety” and “ability” in the sentence. Such positive words were followed by “but” and “bad”, which to some extent support that the BERT model has the ability to utilize the information of word order to help identify the sentiment of a sentence more accurately.

Similarly, below is a tweet that has the annotation as “Extremely Negative” but predicted as “Positive” by our model:

“we may have no food because we have no space to hoard tinned tuna but we have 48 extra long rolls of recycled toilet paper downunder loorollshortage covid19 tbh we were not buying out of panic its our.”

As shown in Figure 13B, the most contributing words are “not”, “but”, “out of”, etc. There was the word “but” after negative expressions such as “no food” and “no space”, and the word “not” before negative expressions such as “out of panic”, which may be why our model predicted this sentence as positive although there were many negative words in the sentences. Again, it supports the efficacy of the BERT model to use the contextual information in text.

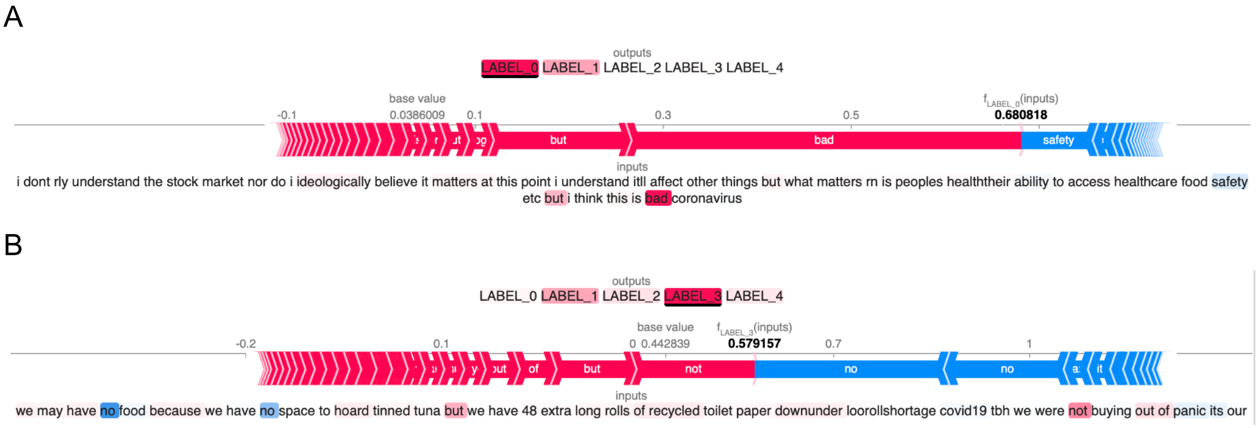


Table 13: Top words impacting the model prediction toward specific categories, with red words increase the likelihood of the sentence to be classified as this category, and blue words decrease the likelihood of the sentence to be classified as this category

Public Sentiment Towards #ChatGPT

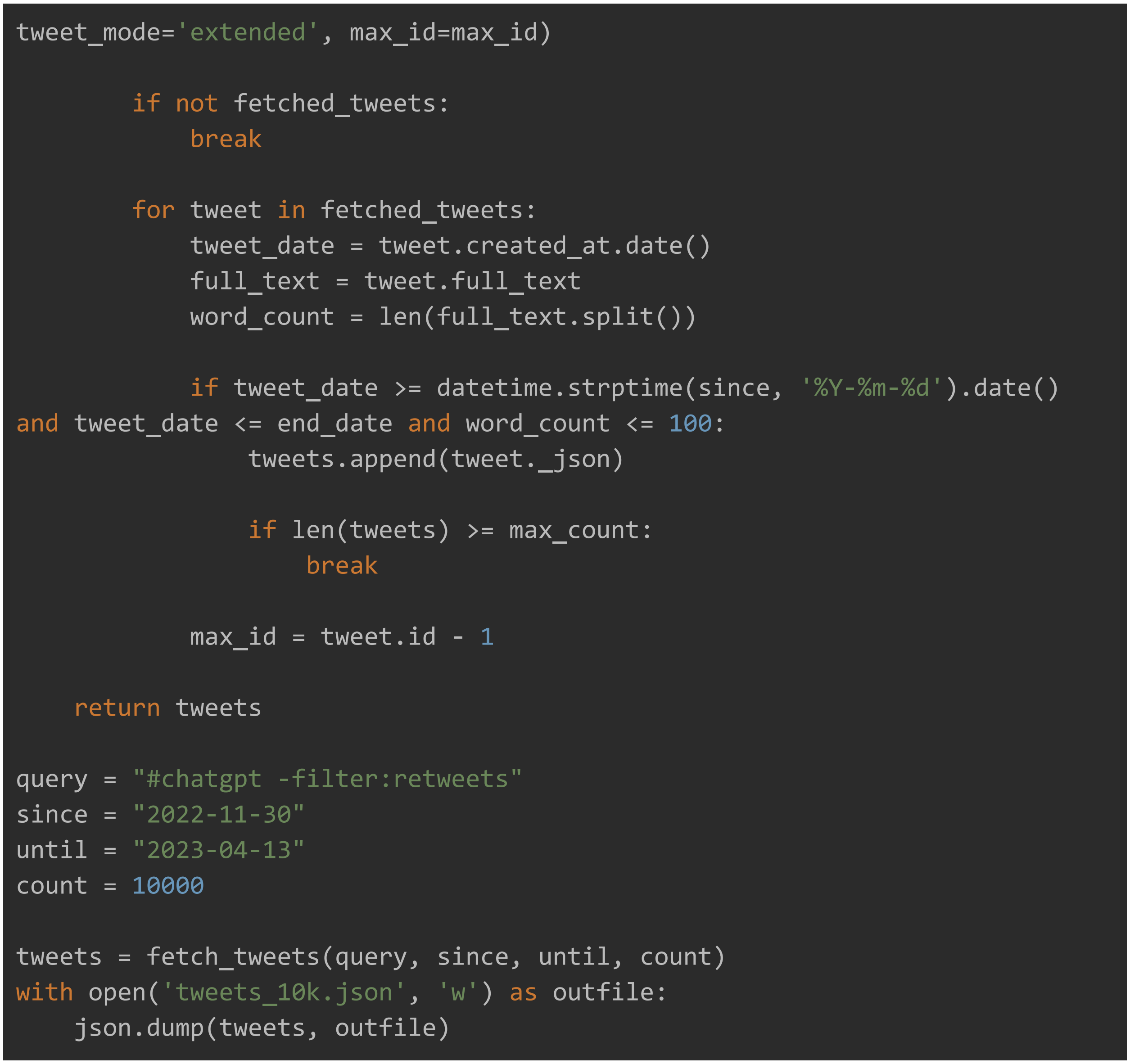
ChatGPT has had a significant impact on our daily lives, enabling people to work more efficiently. However, some individuals argue that ChatGPT could lead to an abundance of misinformation, ultimately harming society. Consequently, we became interested in understanding public sentiment towards ChatGPT. To this end, we aimed to apply our fine-tuned BERT model, initially trained on tweet sentiment data, to tweets specifically related to ChatGPT.

We began by applying for a free API on Twitter and used the API to fetch 10,000 tweets containing the hashtag #ChatGPT:

```
auth = tweepy.OAuthHandler(API_KEY, API_SECRET_KEY)
auth.set_access_token(ACCESS_TOKEN, ACCESS_SECRET_TOKEN)
api = tweepy.API(auth)

def fetch_tweets(query, since, until, max_count):
    tweets = []
    max_id = None
    end_date = datetime.strptime(until, '%Y-%m-%d').date()

    while len(tweets) < max_count:
        fetched_tweets = api.search_tweets(q=query, count=100, lang='en',
```



```
tweet_mode='extended', max_id=max_id)

        if not fetched_tweets:
            break

        for tweet in fetched_tweets:
            tweet_date = tweet.created_at.date()
            full_text = tweet.full_text
            word_count = len(full_text.split())

            if tweet_date >= datetime.strptime(since, '%Y-%m-%d').date()
and tweet_date <= end_date and word_count <= 100:
                tweets.append(tweet._json)

                if len(tweets) >= max_count:
                    break

            max_id = tweet.id - 1

    return tweets

query = "#chatgpt -filter:retweets"
since = "2022-11-30"
until = "2023-04-13"
count = 10000

tweets = fetch_tweets(query, since, until, count)
with open('tweets_10k.json', 'w') as outfile:
    json.dump(tweets, outfile)
```

We performed data cleaning on the fetched tweets to ensure optimal analysis. After cleaning, we applied our trained BERT model to classify the sentiment of these tweets related to ChatGPT. Our analysis revealed that only about 16% of the tweets exhibited negative sentiment towards ChatGPT, indicating an overall positive sentiment (Figure 14).

To gain further insight, we analyzed sentiment across different regions worldwide, allowing us to observe specific countries or regions' attitudes towards ChatGPT (Figure 15).

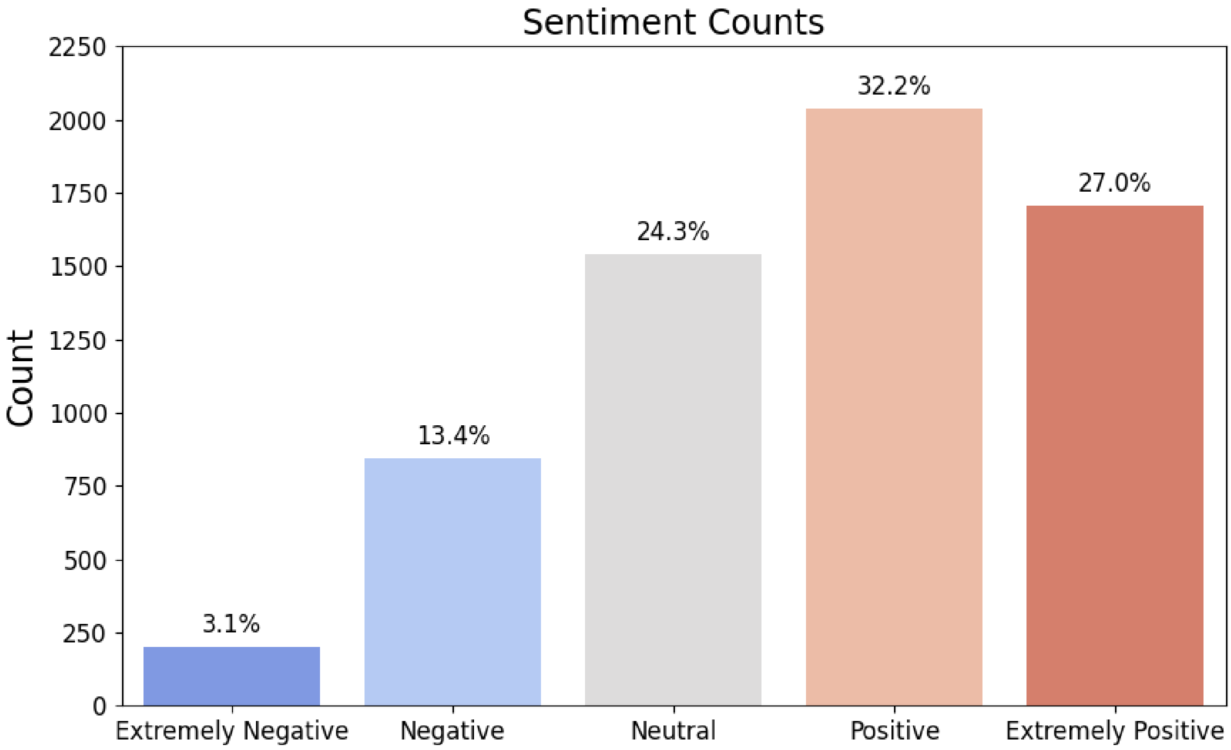


Figure 14: public sentiment towards #ChatGPT

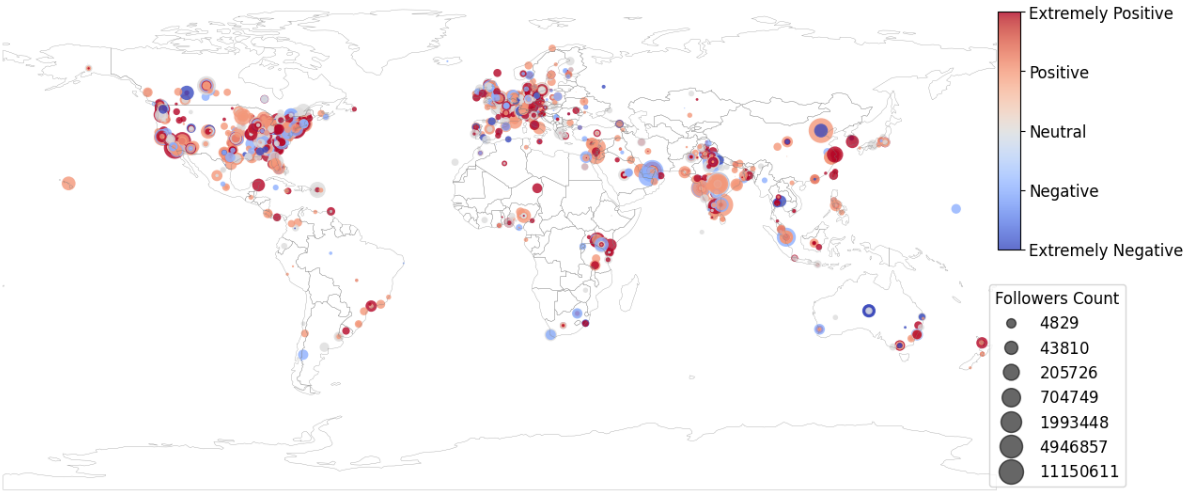


Figure 15: public sentiment towards #ChatGPT (worldwide)

Discussion

In this study, we completed a text classification task to identify the sentiment of Covid-19 related tweets by training and comparing multiple machine learning and deep learning algorithms (i.e., Logistic Regression, K-nearest Neighbors, Random Forest, Decision Tree, Naive Bayes, LightGBM, Gradient Boosting, XGBoost, CatBoost, AdaBoost). In classical machine learning classifiers that heavily rely on the word frequency, logistic regression performed the best over other models with a weighted F1 score of 0.67. We also examined a pre-trained BERT model from Hugging Face, which performed better than the logistic regression model after fine-tuning and achieved a weighted F1 score of 0.79. These results were in line with the previous studies[8] [9] [10] showing that BERT outperformed other classical machine learning schemes such as Support Vector Machine (SVM) and word embedding (word2vec and Glove) with Long Short-Term Memory. BERT is designed to understand the content of text taking into account the contextual information and the word sequences, which therefore may provide a more accurate and comprehensive understanding of the sentiment components of human language. Indeed, the capability and efficiency of BERT to utilize the text context and word order information were supported by our analysis toward specific tweets using SHAP.

Moreover, we used SHAP to interpret the BERT model and identified the keywords that contributed the most to the sentiment predictions. At two different time points of the pandemic (i.e., right after and one month after the shutdown policy), different sets of keywords emerged related to positive and negative tweets, which may suggest shifts in the specific topics of public concern along with the progression of the pandemic. This practice may serve as an approach to understand the change of public perceptions toward a specific event or topic over time and across locations by identifying feature importance.

Finally, we applied our fine-tuned BERT model to tweets related to another topic, ChatGPT. We found that the public showed an overall relatively positive attitude toward ChatGPT. We further mapped the predicted tweets sentiment on the world map to visualize the regional distribution of positive and negative tweets. This practice demonstrates a flexible way to use the social media data to explore public perceptions and its potential socioeconomic and psychological correlates.

Some strengths of this study include: (a) we used a relatively large dataset involving user-generated social media texts, and (b) we generalized and applied our model to tweets about other topics of public concern such as ChatGPT. This study also has some limitations. First, we just have limited self-reported location information in the Covid-19 tweets dataset. Therefore, we were not able to explore the data with regard to users' locations and address potential location-related research questions, which may be interesting to investigate. We were also not able to guarantee the accuracy of the location information drawing on the users' self-report. Second, we just have tweets over one month. Therefore, we can only explore the public perceptions for a relatively short period of time. The Covid-19 pandemic has lasted for a longer time and can have long-lasting effects. It can be interesting to explore how public

[8] Devlin, Jacob, Chang Ming-Wei, Kenton Lee, and Kristina Toutanova. "BERT: Pre-Training of Deep Bidirectional Transformers for Language Understanding." *arXiv.org* (2019).

[9] Qasim R, Bangyal WH, Alqarni MA, Ali Almazroi A. A Fine-Tuned BERT-Based Transfer Learning Approach for Text Classification. J Healthc Eng. 2022 Jan 7;2022:3498123. doi: 10.1155/2022/3498123. PMID: 35013691; PMCID: PMC8742153.

[10] U. N. Wisesty, R. Rismala, W. Munggana and A. Purwarianti, "Comparative Study of Covid-19 Tweets Sentiment Classification Methods," *2021 9th International Conference on Information and Communication Technology (ICoICT)*, Yogyakarta, Indonesia, 2021, pp. 588-593, doi: 10.1109/ICoICT52021.2021.9527533.

concern changes throughout the entire course of the pandemic, even after it ends. Finally, only a small portion of the data was used for the interpretation of the model due to the limited computational resources. Using a larger portion of the data can lead to a better and more comprehensive summary of the model interpretation.

Conclusion

In conclusion, this study compared the performance of different machine learning and deep learning algorithms in a text sentiment classification task using a Covid-19 tweets dataset. Logistic regression outperforms other classical machine learning classifiers such as LightGBM and XGBoost. After fine-tuning, the pre-trained BERT model outperformed logistics regression. We also showed how such models and approaches can be used to understand public perceptions of a specific topic (e.g., trends in public attitudes over time) and their potential correlates (e.g., attitudes differences between users with varied characteristics) in flexible and efficient ways.

Based on the current study, future studies can consider the following next steps. First, more specific and accurate location data, including longitude/latitude coordinates or GPS data, can allow the investigation of more meaningful and interesting research questions. For example, a time series analysis can be used to explore how proximity may exert influence on the extent to which the virus spread impacts public sentimental perceptions. Second, future studies can focus on enhancing the accuracy of observations that may be misclassified along the positive/extremely positive and negative/extremely negative decision threshold. Finally, future studies can consider expanding the time frame of the tweets collection to get more insights on the public perceptions in a longer run.

Acknowledgment

Taylor Cox	100%
Zexi Zhou	100%
Liaoyi Xu	100%