JSC270 A4 - Report

Group 18: Harnehmat Kaur, Henry Qin, Raon Kim

Link to Github Repository: https://github.com/nehmat-kaur/JSC270_A4_Group18

Breakdown of work

Harnemat Kaur

Part 1:

- E, F, I, K

Part 2:

- Sentiment Analysis
- Fitting the Model
- Fine-tuning the model

Henry Qin

Part 1:

- C, D, H

Part 2:

- Deciding and making the list of keywords for the industry dataset
- Fine-tuning the model
- Making the presentation slides

Raon Kim

Part 1:

- A, B, G, J

Part 2:

- Deciding and making the list of keywords
- Exploratory Data Analysis
- Making the presentation slides

Work done jointly:

Part 2:

- Problem description and motivation
- Finding, cleaning and modifying the data
- Making new labels for the dataset
- Writing the report
- Working on the presentation

Part I: Sentiment Analysis with a Twitter Dataset

A) Sentiment 0: 37.41%, Sentiment 1: 18.74%, Sentiment 2: 43.84%

B) (In Colab)

C) (In Colab)

D) We might want to keep some punctuation when they convey a certain tone or add effect to the sentence. For example, exclamation marks can convey an extreme complaining or sarcastic tone; and tweets containing the question mark might be either neutral (a genuine question) or a rhetorical question (those usually convey negative tones in the context of Covid).

   Additionally, # is the special character leading a Twitter hashtag, which sometimes might contain a positive, neutral or negative tone itself. And that is not necessarily related to the tone of the tweet itself, which might affect our judgment of the tone.

E) We used the PorterStemmer, in Colab.

F) (In Colab)

G) The length of our vocabulary is 74225, we did not set a max_feature value in the CountVectorizer for more accurate results in the later steps.

H) Test Accuracy: 0.6706161137440758
   Train Accuracy: 0.820742108716254

```
Sentiment 0
word: thi, count: 3225.0
word: food, count: 3638.0
word: price, count: 4347.0
word: covid19, count: 4610.0
word: coronaviru, count: 6737.0

Sentiment 1
word: price, count: 1365.0
word: supermarket, count: 1442.0
word: store, count: 1589.0
word: covid19, count: 2568.0
word: coronaviru, count: 3812.0

Sentiment 2
word: price, count: 3339.0
word: thi, count: 3781.0
word: store, count: 3917.0
word: covid19, count: 5681.0
word: coronaviru, count: 7511.0
```

I) We think it's not appropriate to fit an ROC curve in this scenario, because ROC curves are used to count/measure True Positive and False Positive values (which are in binary classification models). Since this scenario isn't a binary classification (there are 3 classes), ROC won't give a meaningful result.

J) Test Accuracy: 0.6169036334913112
Train Accuracy: 0.7137754234199207

The accuracies are slightly lower than those of count vector's. We thought this might be because Naive Bayes assumes independence, which might not be true for all cases, and maybe TF-IDF features are less independent which might have caused the model to be less accurate.

```
Sentiment 0:
coronaviru: 1139.9784423301903
price: 792.7742494390648
food: 701.3965552925403
covid19: 940.1614987111449
thi: 727.9053733628093

Sentiment 1:
coronaviru: 1139.9784423301903
covid19: 940.1614987111449
store: 789.9888913548571
supermarket: 748.2646621155793
groceri: 700.0258242193139

Sentiment 2:
coronaviru: 1139.9784423301903
covid19: 940.1614987111449
store: 789.9888913548571
thi: 727.9053733628093
groceri: 700.0258242193139
```

K) We use the WordNetLemmatizer to lemmatize tokens.

Test Accuracy with lemmatization: 0.6274354923644023
Train Accuracy with lemmatization: 0.7354992345637014

The test accuracy is higher for lemmatization as compared to stemming by approximately 0.01. The train accuracy is also higher for lemmatization by roughly 0.02. These differences are not very huge, so we can say that the increase in accuracy is not very significant.

L) (BONUS QUESTION)
Naive Bayes is generative, because we're relying on P(Y,X), i.e., the joint probability distribution and Bayes Rule instead of modeling P(Y|X) directly.

Part II: “ChatGPT Tweets across Industries”

Problem Description and Motivation

Through the analysis in this report, we aim to find an answer to the research question, “**Can we predict the industry context of a tweet based on its sentiment towards ChatGPT?**”. ChatGPT has induced some rather mixed opinions ever since its launch- but how are these polarizing perspectives related to the various backgrounds of people? Can we predict the industry from these perspectives? We chose seven disjoint industries that we are currently or would like to be involved in the future and decided to study how industries have an impact on the opinions of the usage of ChatGPT.

With our dataset of 30,000 tweets we can train a Naive Bayes model using the sentiment scores and the corresponding industries, then we can predict industry context based on sentiment towards ChatGPT. There were several prior researchers that explored sentiment analysis across different domains and social media platforms, but we found an article by Khalid Ansari (Ansari, 2019, Medium), which used the same dataset as us and was the most similar to our analysis on ChatGPT related tweets. In the article he applied sentiment analysis on topics such as AI, ChatGPT, Elon Musk and many more, to understand how tweets on different topics have different opinions towards ChatGPT. However, in our analysis we will rather use the sentiment scores to predict the industry that the tweet’s topic is associated with, providing an understanding of industry-specific sentiments, and ultimately contributing to the broader discourse on AI adoption in various sectors.

Data Description

The original dataset used for this project was extracted from Kaggle as a result of the technical restraints being experienced with the Twitter API access. If we were to use Tweepy, we would've extracted the tweets by keywords ("ChatGPT", "Chat GPT"), hashtags "#ChatGPT" and @mentions of ChatGPT. It is called "500K ChatGPT-related Tweets Jan-Mar 2023" (Ansari, 2019, Kaggle), and it contains 500,000 tweets talking about ChatGPT, from January 2023 to March 2023. Each observation contains 6 features. Due to computational restraints (RAM crashes, high runtime), the modified dataset we decided to use for our project is a selected subset of the first 30,000 tweets mentioning ChatGPT across various industries. To this, we added two additional features, "industry" and "sentiment_score". The industries include Technology, Education, Finance, Media & Entertainment, Retail & E-commerce, Healthcare, Hospitality & Tourism, and Other.

The industries are labeled by using a list of keywords, and the sentiment_score is a number between -1 and 1 (-1 means negative opinion and 1 means positive opinion), and each tweet was labeled with a score which was assigned by TextBlob, a python library for sentiment scoring, with like counts and retweet counts taken into an account.

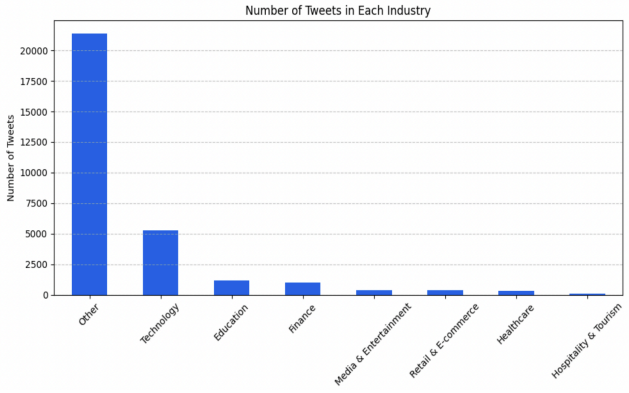
Similar to the research project cited above, there are many research projects that analyze the content of tweets and carry out natural language processing techniques on them for either categorisation or sentiment analysis. This project is not doing something novel in terms of the analysis, but it combines different analysis techniques to explore a unique question which we believe is worth exploring. Our strength is that we have a

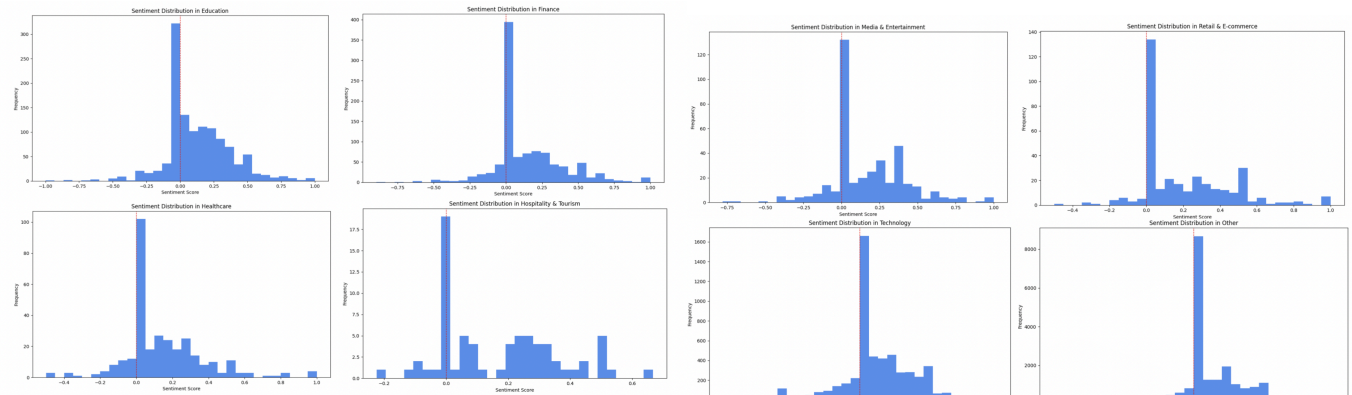
large volume of tweets to filter from and that we have metrics other than the content itself, such as retweets, likes etc. on which we can perform an engagement analysis on. But our weakness is that those tweets were mostly from over a year ago and might not represent the current general public as well.

Exploratory Data Analysis

Our exploratory data analysis involved tokenizing, cleaning (removing URLs, punctuation, and stopwords, and converting text to lowercase), lemmatizing, and adding new features such as industry and sentiment_score. We observed the number of tweets related to each labeled industry and the distribution of sentiment scores across industries. The analysis revealed that most tweets were classified as 'Others', followed by 'Technology', then 'Education', indicating an imbalance in tweet distribution. This could be due to our classification method or the nature of ChatGPT discussions on Twitter. Regarding sentiment, most tweets had near-neutral scores, with a predominance of positive sentiment across all industries. Each industry showed a unique sentiment distribution, suggesting potential for predicting industry association based on sentiment.

```
Other                    21370
Technology                5288
Education                 1196
Finance                   1013
Media & Entertainment      394
Retail & E-commerce        348
Healthcare                 319
Hospitality & Tourism       72
Name: industry, dtype: int64
```





Machine Learning Model

The Multinomial Naive Bayes (MNB) classifier is utilized to predict the industry context of tweets based on their sentiment scores towards ChatGPT. Sentiment scores extracted from the tweets are normalized using MinMaxScaler to ensure consistency in feature magnitudes. The MNB classifier is suitable for this research problem due to its simplicity, efficiency, and effectiveness in text classification tasks.

The model we use is a supervised learning model as it requires labeled training data, in our case, tweets labeled with their respective industry contexts, to learn the relationship between sentiment scores and industry labels. Our model's strength is that it is computationally efficient, making it suitable for processing a large dataset of tweets. Its simplicity facilitates easy implementation and interpretation. It also effectively handles text classification tasks and is robust to irrelevant features. However, the MNB classifier assumes feature independence, which may not hold true in all cases, potentially affecting its performance.

The Multinomial Naive Bayes classifier achieved moderate performance with a test accuracy of approximately 71.03%, indicating its ability to accurately predict the industry context of tweets based on sentiment scores. However, the precision score of 50.46% suggests that there is room for improvement, while the recall score of 71.03% highlights the model's effectiveness in capturing tweets relevant to a given industry. The F1-score of 59.00% reflects a balanced performance between precision and recall, indicating overall reasonable model performance.

Conclusion

In our analysis aimed at predicting the industry context of tweets based on sentiment towards ChatGPT, our model achieved moderate success. We accurately categorized about 71% of the tweets in the test set, although precision was lower, correctly identifying only around half of the tweets belonging to specific industries.

The main challenge we faced was class imbalance, leading the model to frequently predict 'Other' or 'Technology' due to the dominance of these categories in the dataset. We tried using Latent Dirichlet Allocation (LDA) for our model first, but it was unsuitable for our task due to the need for pre-defined labels for clusters and computational inefficiencies. Naive Bayes' assumption of feature independence aligned well with our text data, making it a more suitable choice.

Given more time, we would explore using LDA on a smaller dataset without predefined labels to potentially identify clusters that could be categorized as industries. Additionally, we would consider employing other supervised clustering algorithms that could efficiently handle our text data. Experimenting with additional features beyond sentiment scores, such as like_count or retweet_count, could also enhance our model's predictive performance and provide deeper insights into tweet categorization.

Works Cited

- Ansari, Khalid. *"Cracking the ChatGPT Code: A Deep Dive into 500,000 Tweets Using Advanced NLP Techniques."* Medium. April 19, 2023. https://medium.com/@ka2612/the-chatgpt-phenomenon-unraveling-insights-from-500-000-tweets-using-nlp-8ec0ad8ffd37.

- Ansari, Khalid. "500k ChatGPT-related Tweets Jan-Mar 2023." Kaggle. April 1, 2023. https://www.kaggle.com/datasets/khalidryder777/500k-chatgpt-tweets-jan-mar-2023.