CS 490A Final Project Report
Douglas Silverman, Britney Muth, Holly Wagner

Comparing Classification Methods for Deriving Sentiment from COVID-19 Pandemic Related Tweets

Abstract

The goal of this project is to determine the best classification method to analyze thousands of tweets regarding the COVID-19 pandemic and to determine if the sentiment of the tweet is positive, negative or neutral. This is of particular interest given the current unstable climate. We explore and compare various methods of classification such as VADER, a non-machine learning approach, as well as train ML-classifiers such as logistic regression, naive bayes and a random forest. Analysis and comparison of each of these models confirmed our hypothesis that VADER would have the best performance on the tweets, however logistic regression was also found to be relatively accurate.

Introduction

Sentiment analysis, sometimes referred to as opinion mining, has the practical yet ambitious goal of computationally quantifying subjective information. It is often used in business as a marketing strategy and in politics to predict election outcomes. When implemented successfully, a sentiment model will capture the public opinion of the subject it is trained on.

In March 2020, COVID-19 was declared a pandemic by the World Health Organization[1]. The twitter platform generates a wealth of information through its users, making it an ideal source for gathering overall public opinion on several matters, including the

[1]Avera writers. “How Does a Pandemic End?”

pandemic. Twitter's hashtag feature can further facilitate the evaluation of people's perspectives relating to a certain topic.

The ability to gauge the stance of the public on certain issues relating to the pandemic has several potential applications. For instance, resource strain, political unrest and noncompliance with social distancing regulations like curfews are all tied to how people respond in times of crisis. Tuning an accurate model to assess the sentiment of individuals on such topics would be a useful first step in achieving this task.

Related work

C.J. Hutto and Eric Gilbert from Georgia Tech outline their approach to sentiment analysis in their paper, *VADER: A Parsimonious Rule-based Model for Sentiment Analysis of Social Media Text*[2]. The team from the study applied their own human developed dictionary to social media texts. This dictionary consists of an empirically validated set of lexical features paired with their associated sentiment intensity measures. Furthermore, this dictionary is specifically attuned to sentiment in microblog-like contexts. This is a very similar task to our own, except we strictly look at COVID-19 related tweets. Their VADER model received an impressive F1 classification accuracy of 0.96.

The paper, *An Ensemble Classification System for Twitter Sentiment Analysis*[3], written by Ankit and Nabizath Saleena explores an ensemble classifier which combines the outputs of many classifiers to generate a final prediction. The principle idea behind ensemble classification is that it works to minimize the variance resulting in a more robust model. The team performed sentiment analysis on tweets and similar texts to observe the performance of their proposed ensemble against other classifiers. Contrary to our multiclass approach, binary classification for positive and negative sentiment only

[2] Hutto, C. J., and Eric Gilbert. VADER: A Parsimonious Rule-Based Model for Sentiment Analysis of Social Media Text
[3] Ankit, Nabizath. "An Ensemble Classification System for Twitter Sentiment Analysis."

was used. The random forest model they trained used 150 estimators along with a max depth of 30.

Data

The dataset that we are using was obtained from Kaggle. This dataset contains ~45k entries that are already labelled. It is also already partitioned into training and test data (~4k entries). The dataset contains important information such as location, date tweeted, the label (sentiment of tweet) and of course the raw text of each tweet. This dataset is easy to obtain, it is already labelled making it ideal for testing our classifier and large enough to be confident in the results.

The dataset however does not come cleaned and so we implemented a clean_data function and a cleaned_csv function for this purpose located in ConvertData.py. The challenge with cleaning the data was reading non ‘utf-8’ characters from the csv. Because all the data comes from twitter, there are a lot of emojis and other characters that are hard to deal with. For our data we only wanted to look at the words, so we had a string of “allowed characters” that we care about. This means we parsed every tweet and deleted every character that was not in [a-z][0-9] and [-@$#%]. This is so we include money amounts, @ing users, hashtags, and percents. This way we can read the original csv files using latin-1 and then we can exclude any characters that will hinder the tokenization. We then saved the changes into a new csv file.

The data from the training set contains tweets from March 16, 2020 to April 14, 2020, while the data from the test set contains tweets from March 2, 2020 to March 16, 2020. The tweets are global and overall appear to contain more positive tweets than negative.

Positive	Negative	Neutral
Train Data		
18046	15398	7713
Test Data		
1546	1633	619

Method

Our method was to compare the performance metrics of four different models on our dataset, three classifiers and one rule-based model: Logistic Regression, Naive Bayes, Random Forest and VADER, respectively. Our Logistic Regression, Naive Bayes, and Random Forest were all implemented using Term Frequency - Inverse Document Frequency (tf-idf) as a preprocessing step as it decreases the weights of high frequency function words while increasing the weights of topic words. The point of tf-idf is to lower the value of words that are seen in many documents. For example, the word COVID came up in most of our documents. Out of context, COVID would probably have a negative sentiment. However, since our dataset contains on COVID-19 tweets, this word loses its value. In our implementation of tf-idf, we chose to include stop-words because some of the words might deliver sentiment. For example, referring to a certain politician as him or he might be negative or positive. We chose ‘latin-1’ encoding to catch any non-ascii characters that remained in our cleaned tweet:

```
vectorizer = TfidfVectorizer(encoding= 'latin-1')
```

Our Logistic Regression classifier uses ‘lbfgs’ as its solver, a max_iter of 1000.

```
clf = LogisticRegression(solver= 'lbfgs', multi_class= 'multinomial',
max_iter= 1000)
```

Naive bayes multinomial classifier from Sklearn used an alpha smoothing parameter α = 0.28 which we found to give the highest accuracy score.

```
clf = MultinomialNB(alpha = 0.28)
```

The random forest from Sklearn uses a decision tree as its base classifier and consists of 250 estimators. We used entropy to measure the quality of the splits as opposed to the gini index because it resulted in a slightly better accuracy.

```
clf = RandomForestClassifier(n_estimators=250, criterion = 'entropy',
random_state=42)
```

VADER was the only lexicon and rule based model that we used since it is attuned to social media text.

Results

Below are the results of some metrics we used to rate the various models:

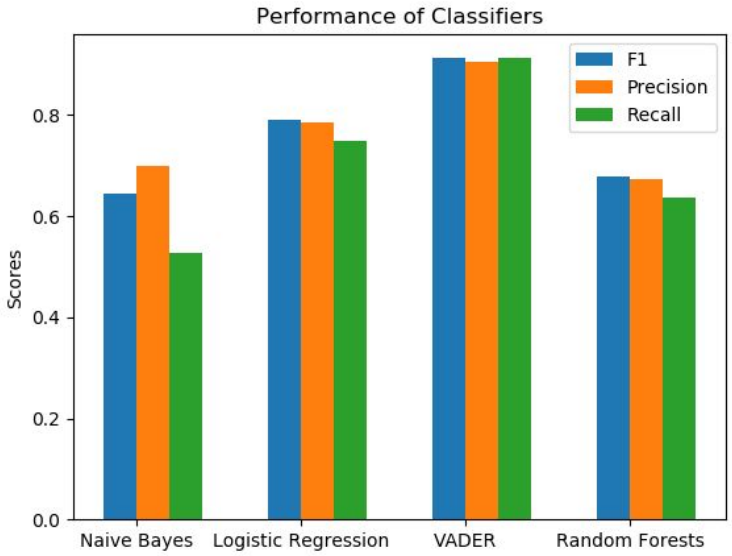
	Accuracy (F1 score)	Precision	Recall
Naive Bayes	0.6451	0.6997	0.5278
Logistic Regression	0.7915	0.7842	0.7498
VADER	0.9134	0.9048	0.9123
Random Forests	0.6793	0.6737	0.6344

VADER appeared to significantly outperform our classification models in all regards. Logistic Regression was the second best overall. While Random Forests had a higher accuracy and recall score than Naive Bayes, Naive Bayes had a higher precision score.

	Positive	Negative	Neutral
Actual	1546	1633	619
Naive Bayes	2032	1672	94
Logistic Regression	1678	1644	476

VADER	1539	1613	646
Random Forests	1918	1426	454

Our Naive Bayes model had the most difficulty predicting Neutral tweets compared to the other models. This could be due to the fact that the training data is unbalanced and contains significantly less neutral tweets (18.7%) compared to the number of positive (43.8%) and negative tweets (37.5%). Logistic Regression and Random Forests were able to predict Neutral tweets better than Naive Bayes. VADER performed the best which might have been expected. According to our source on VADER, on non-specific tweets, VADER had an accuracy of 0.96 which is remarkable for any model.



We can look more at the highest valued words for positive, negative and neutral tweets to see what makes people feel the most positive or most negative about. The table below shows the top 20 highest scoring words for each sentiment.

	Top TF-IDF score for each Sentiment					
Rank	Positive		Negative		Neutral	
1	coronavirus	547.80	coronavirus	489.16	coronavirus	311.54
2	19	425.83	19	384.52	19	224.24
3	covid	415.94	covid	375.48	covid	219.47
4	covid19:	382.08	food	353.036	covid19	214.58
5	store:	355.48	prices	335.71	store	180.84
6	grocery:	339.64	covid19	306.08	supermarket	174.02
7	supermarket	323.30	supermarket	282.48	grocery	171.15
8	food	301.58	people	275.70	prices	152.44
9	prices	289.69	store	250.65	consumer	139.13
10	amp	265.31	panic:	249.94	shopping	123.09
11	people	257.83	grocery	245.40	online	115.03
12	consumer	234.89	amp	202.60	food	97.490
13	hand	233.81	consumer	181.07	toiletpaper	96.701
14	shopping	231.40	buying	175.38	pandemic	91.506
15	online	228.68	crisis	174.38	people	89.874
16	sanitizer	226.67	demand	170.96	amp	68.765
17	like	215.23	pandemic	155.90	toilet	66.241
18	help	214.58	oil	153.520	paper	64.226
19	workers	209.79	shopping	152.35	new	62.690
20	pandemic	187.05	need	146.59	retail	61.690

From the chart above, we can observe that neutrally labeled words tend to have lower tf-idf scores than positive and negative. It should also be noted that there is a lot of overlap among labels such as words like “covid” and “coronavirus” as well as tokens relating to shopping. It makes sense that the words “crisis” and “demand” appear in the top 20 for negative tweets because early in the pandemic there was a strain on

resources brought about by the widespread panic of the unprecedented event. Most people seem to be concerned about getting your next meal or being able to provide for their family. This can be seen with words such as “consumer”, “shopping”, “grocery” and “prices”, which are shared throughout the labels. This means regardless of the tweets label, these terms indicate what was on people’s mind in March through April of 2020.

Discussion and Future Work

Better scoring/predictions could possibly be achieved for Naive Bayes, Logistic Regression, and Random Forests by using different variations of the tf-idf such as the inclusion of stop words, stating minimum and maximum values for the number of documents that a term must be found in to be included, setting a maximum frequency for terms, and normalizing or smoothing the tf-idf algorithm itself. We could also attempt to find a more balanced dataset that contains more neutral tweets.sup

Alternatively, more preprocessing could be done to handle class imbalance, such as the lack of neutrally labeled tweets, through techniques like adjusting the class weights for each model. The addition of a weight for each class would prioritize the minority classes such that the classifier can learn equally from all three classes.

Moving forward, we could implement other strategies instead of bag-of-words, such as word embeddings and language models like BERT, along with its several variations to find the most accurate model with this dataset.

We could also move into further analysis with this dataset by dividing the tweets by location and comparing the overall sentiment within that location to the number of cases within that area as well as noting the policies that were enacted as a response to the pandemic. It would be interesting to see if the sentiment within an area correlates with the number of cases there. In addition to this, we could also compare these findings with more recent tweets to see if the overall sentiment correlates with the increase or decrease in the number of cases in the given location. We could calculate how significant, influential, and reliable public sentiment via Twitter could be to the prediction

of the number of COVID cases in a pandemic where we all must rely on one another to keep each other safe.

References

Ankit, Nabizath. “An Ensemble Classification System for Twitter Sentiment Analysis.” *Procedia Computer Science*, Elsevier, 8 June 2018, www.sciencedirect.com/science/article/pii/S187705091830841X.

Avera writers. “How Does a Pandemic End?” *Avera*, 25 Aug. 2020, www.avera.org/balance/infectious-disease/how-does-a-pandemic-end/.

Hutto, C. J., and Eric Gilbert. *VADER: A Parsimonious Rule-Based Model for Sentiment Analysis of Social Media Text*, Georgia Institute of Technology, comp.social.gatech.edu/papers/icwsm14.vader.hutto.pdf.