CS 230

Sentiment Analysis on COVID-19 Tweets

Michael Lin, George Younger

Abstract

Initially this project was designed to explore possible relationships between sentiments among the general public's tweets and Donald Trump's tweets. However, after exploration determined no such relationship existed, the project pivoted to exploring sentiment analysis using several different models on Coronavirus after no exposure to Coronavirus tweets in training. These predictions are then compared to a ground truth, state-of-the-art sentiment analysis model from a Python library called NLTK to determine how well models adapted to Coronavirus as a negative concept.

1 Introduction

In 2020, the coronavirus has had an incredible impact on the psyche of the public. Given stay-at-home orders all over the United States, many people have taken to social media to air their grievances and talk about the current situation. In such a highly charged time, we believe that there is much to be learned about how people respond to crisis, and sentiment analysis is a perfect way to dig into that psyche.

Initially, we wanted to model how the U.S. government responds to what the public is saying (specifically on Twitter), and we figured President Trump could be a fascinating study of that behavior. In our first iteration of this project, we trained a basic sentiment classifier in order to label coronavirus-related tweets as positive or negative, and we attempted to train a model that would find a correlation between the sentiment of the U.S. population of a given day and the sentiment of President Trump's tweets the next day [8]. However, every model that we tried to train failed to create a good predictor of Trump's tweets' sentiments, so we had to shift our focus for this project in the last week or so.

One of the main issues that we faced was we couldn't find a good sentiment analysis classifier for our initial COVID-19 tweet dataset, so we decided to focus our efforts on that.

What our project ended up turning into is a comparison of 6 different basic models (Naive Bayes, Decision Tree, SVM, logistic regression, LSTM, CNN) on our dataset of coronavirus tweets in the United States (in a three-day span in March), as compared to a state-of-the-art classifier. We figure that this dataset is unique because we expect it to be largely negative (due to the nature of the virus), so we're interested in seeing how different models respond to mostly negative datasets. The goal is to see which of these six models perform the best (relative to the state-of-the-art, which uses Naive Bayes as a base), to see if another model could possibly be a better base to build upon when analyzing a dataset that's pretty negative. We

found a state of the art model online called TextBlob that utilizes a library called NLTK with which we could compare our newly designed model. Given that our new goal was just to build as good a sentiment classifier as we could for coronavirus tweets, we treated this model as the ground truth because it seems widely accepted to be a very accurate sentiment classifier, especially as far as binary sentiment is concerned.

2 Related Work

There is quite a bit of previous work on sentiment analysis regarding tweets. Many of the related works that we found could be categorized into two different types of papers: 1) the ones that were focused on specific tasks that involved classifying tweets, and 2) papers that focused on improving certain aspects of a given model to perform better.

As for the papers in the first category, many have used tweets as a marker for social events, similar to our proposed goal. In [1], the authors focused on extracting opinions from tweets regarding two prime minister candidates in Australia, while [2] focused on using tweets about movies to determine that movie's eventual success. [1] uses opinion word extraction partnered with the word's intensity (Strong Negative, Negative, Neutral, Positive, and Strong Positive), which worked pretty well. This shows that a lexicon-based sentiment analysis model is doable and relatively straightforward. As for [2], they used a Naive Bayes classifier on a simple frequency ratio measure of how positive/negative a review is regarding the movie. We initially were going to do something similar for our original project with President Trump's tweets. Instead, we can adopt this as one of our models to experiment on.

[3] and [4] worked to improve some current state-of-the-art sentiment analysis models, using tweets as a dataset. [3] used an existing SVM model with word and character ngrams and attempted to improve specifically the example with negation by creating a whole separate tweet-specific sentiment lexicon for just the negated contexts. [4], like [2] also uses Naive Bayes, but it explored the usage of word embeddings on tweets regarding the US airlines. It tried different types of word embeddings, including creating its own word embeddings and using Stanford NLP's GLoVe embeddings, which ended up working the best. We will be using GloVe with a couple of our models, namely our CNN and RNN models.

3 Dataset

Here, we will briefly outline our dataset as it applies to both iterations of this project. For our initial project, our data consisted of English language coronavirus-related tweets in the U.S. every day from the dates 3/1/2020 to 4/15/2020 [5]. Our data also consisted of all of President Trump's original (not re-tweeted) tweets over the same time period. The three original sources of the data sets can be found at links in the references section. We conducted pre-processing of all three of these datasets. For the Kaggle datasets, we stripped out the text of the tweets and standardized the date values. We repeated the same process for President Trump's twitter dataset and also used the archive's search feature to generate the initial dataset for preprocessing. We also included a dataset of sentiment analysis on tweets about airlines [4]; we did this in order to train our GloVe model because we didn't have any labeled tweets in our dataset in terms of their

sentiment, and we needed a labeled dataset on which to form our initial model. This way, unlike our first attempt, we're not blindly and relatively arbitrarily assigning sentiments to tweets(given how easy it is for unigram analysis to be completely wrong), but rather building a model based on what humans have already classified and translating it to a different context but within the same language. We believed the transfer learning holds here.

After we had to pivot, we ended up ignoring Trump's tweets, just focusing on the initial dataset. We also had included far more tweets, but given how long it took to train each model and that one day consisted of around 10 to 15 thousand tweets, we decided a sample size of between 30 and 45 thousand tweets would be sufficient to test our model.

However, because this dataset was entirely unlabeled, we needed a different labeled dataset on which to train our initial models. We decided upon Sentiment140's training data [6], which is a dataset of tweets automatically labelled by using tweets with positive and negative emoticons, as stated in their paper. Preprocessing the data essentially consisted of just stripping out unnecessary data points connected with the tweets; the Kaggle dataset we used had a plethora of information about each tweet, and we really only cared about the tweet's text, id, and date/time created, so we removed all the other facets from the csv files we found. We had to make some small adjustments to the data files to make them suitable for training, such as removing null characters, but for the most part we left the datasets untouched so the models could classify on raw text.

It's also worth noting that the ground truth model is trained on a corpus of movie reviews [7], which were labeled positive or negative based on the star rating given in tandem with the text of the review. This corpus is used widely in sentiment analysis training as NLTK uses it to train its own classifiers and NLTK is widely regarded as a state-of-the-art model for sentiment classification.

4 Methods/Experiments

After giving up on our initial hypothesis/project of connecting Donald Trump's twitter sentiment with the general public's sentiment, we pivoted for the final report to an analysis of our dataset, which consisted of unlabeled Coronavirus tweets. Our goal was to assign sentiment labels to some of the tweets in our dataset using different state-of-the-art models, find disagreements among the models, and analyze tweets in which disagreement manifested to try to understand nuances of these models in the context of Covid-19. We found a research paper [9] that had implemented a variety of models, and we chose 6 of their models on which to perform our experiment: their baseline model (which was essentially the same as ours, just positive and negative unigrams), a logistic regression model, an SVM model, a Naive Bayes model, a decision tree model, an RNN model, and a CNN model. This paper had code for each of the models, which we modified slightly but structurally kept the same (which we have denoted with "_FROM[9]" in our code.

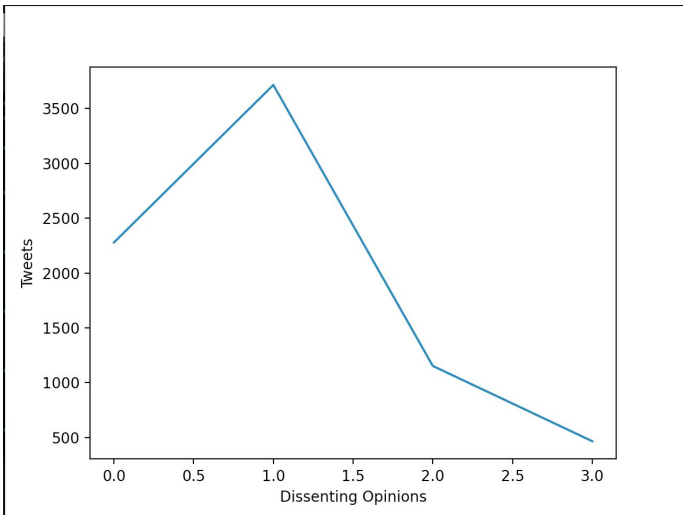
We trained all of these on a labeled twitter dataset that did not have any mention of Coronavirus because we were curious to see how the models would respond in classifying the sentiments of tweets during a global pandemic; we anticipated that the classifications would be largely negative, but we were interested to see where the models deviated from each other. Once

the models were trained, we made predictions on one day's worth of tweets (around 8,000 tweets) and compared these predictions across the models. For the RNN and CNN, we used GloVe embeddings (with 25 dimensions).

Because the state-of-the-art classifier generally took a while to run, we ultimately decided to sample 1000 tweets and compare the results of each of the models above to the results from the state-of-the-art classifier.

5 Results/Discussion

Below is a graph of the results of running each of the models except the state-of-the-art classifier on the 8,000 tweets in our dataset:



This illustrates the agreement of the vast majority of the classifiers; on the x axis we have the number of classifiers dissenting from the majority opinion. With 6 classifiers, a value of 3 on the x axis represents a perfect split between positive and negative sentiment, and a value 0 represents perfect agreement that the tweet is positive or agreement it is negative. As illustrated by the graph, a large majority of the tweets fall into the category of either having no dissent among the classifiers or one dissenting opinion of the 6.

Below is a table illustrating how often each model agreed with the state-of-the-art model and how often they disagreed on the 1,000 randomly sampled tweets:

	BASELINE	NAÏVE BAYES	DECISION TREE	SVM	CNN	LSTM
AGREEMENT	564	369	352	405	392	396
DISAGREEMENT	436	609	648	595	608	604

As is evident from the table, there is some level of disagreement between the state-of-the-art classifier and the models we trained. We attribute this to the overwhelming negativity of our training set, as illustrated by the table below, where we show the number of tweets each classifier decided was positive vs. negative:

	BASELINE	NAÏVE BAYES	DECISION TREE	SVM	CNN	LSTM	GROUND TRUTH
POSITIVE	5199	825	325	1739	1363	1526	666
NEGATIVE	2413	6787	7287	5873	6249	6086	334

As you can see, there is a definite skew for all the models we trained toward the negative side, which is juxtaposed with the more even split for the state-of-the-art classifier; we believe this was what caused the disparity in the classifications of the 1000 tweets we sampled at random from our dataset.

We decided to examine some of the tweets that spurred the most disagreement among the classifiers, of which there were 467, to see where the dissenting opinions came from and to see why these tweets might be difficult to classify. As one example, we looked at the tweet "CPAC will be the 2020 version of the Philadelphia Liberty Parade.#COVID—19 #CoronavirusOutbreak #CPAC #coronvirususa". Here were the decisions that each model made for this tweet:

	BASELINE	NAÏVE BAYES	DECISION TREE	SVM	CNN	LSTM	GROUND TRUTH
PREDICTION	1	0	0	0	1	1	0

One reason this might be difficult to classify is the large number of unfamiliar terms- because we elected to train our model on a non-coronavirus dataset, the models never saw coronavirus-related terms, so #COVID, #Coronavirus, and #coronavirususa wouldn't be detected as contributing to the sentiment of the tweet. Furthermore, this tweet references a parade in 1918 that caused thousands of people to contract the Spanish Flu in Philadelphia; this is an incredibly niche reference that's really only prescient during a pandemic, so this could also cause confusion for the sentiment classifiers given how implicit the sentiment is in the reference to that parade.

6 Conclusion/Future Work

There are several conclusions we can arrive at after experimenting with these six models. First, it seemed as the best baseline model for classifying the sentiment of coronavirus tweets, of the six we tested, was SVM, closely followed by LSTM. Based on the graphs above, they performed most closely to the ground truths in terms of their distributions of positive and negative samples (although, in general, the models didn't match the ground truth that well, as they all skewed negative). In terms of the SVM and LSTM, there were also more instances throughout the coronavirus dataset where they agreed on sentiments than disagreed. We would suggest this model be a potential base model for future development.

Additionally, the unique part of the coronavirus dataset which could warrant future model training is that the subject of almost every single tweet, "COVID" or "coronavirus" isn't in most lexicons, so it poses a uniquely interesting way to test for unknown phrases.

The next steps to be taken would be to explore more in depth some of the features that can be used in the feature selection process for SVM and LSTM, for example, using the opinion word extraction mentioned above. It would also be interesting to create a word embedding for coronavirus itself to see how it fits within the public sentiment.

7 Contributions

Michael: In the first and second iterations of the project, worked on creating the model where the inputs were the sentiments of the tweets of the US population and the outputs were the sentiments of Donald Trump the next day, starting with logistic regression and then moving into using GloVe word embeddings.. In the final iteration, worked on preprocessing the Kaggle dataset to fit into all of the models that we ran with (including parsing the tweets, accounting for emojis, repeated letters, etc). Finally, implemented and ran the Naive Bayes, SVM, and Decision Tree models.
George: In the first iteration, worked on pre-processing data and implementing a baseline unigram sentiment analyzer. In the second iteration, worked on gathering more data and analyzing a more complex model used for sentiment analysis and produced graphs to illustrate there was no such relationship. In the final iteration, ran the LSTM and CNN models and aggregated data from all models to produce graphs and tables showcasing their relationships. Additionally did research to determine which model constituted a state-of-the-art sentiment analysis model (ultimately decided on model from NLTK).

8 References:

[1] Zhou, Xujuan, et al. “Sentiment Analysis on Tweets for Social Events.” *IEEE 17th International Conference on Computer Supported Cooperative Work in Design*, June 2013.
[2] Jain, Vasu. “Prediction of Movie Success Using Sentiment Analysis of Tweets.” *The International Journal of Soft Computing and Software Engineering [JSCSE]*, vol. 3, no. 3, Mar. 2013.
[3] Zhu, Xiaodan, et al. “NRC-Canada-2014: Recent Improvements in the Sentiment Analysis of Tweets.” *National Research Council Canada*, 2014.
[4] Carremans, Bert. “Using Word Embeddings for Sentiment Analysis.” Mar. 2018.
[5] Smith, Shane. “Coronavirus (covid19) Tweets.” *Kaggle*, 12 May 2020, www.kaggle.com/smid80/coronavirus-covid19-tweets.
[6] Go, Alec, et al. “Twitter Sentiment Classification Using Distant Supervision.” *Stanford University*, 2010.
[7] Data, NLTK. “Movie Reviews.” *Kaggle*, 19 Oct. 2018. www.kaggle.com/nltkdata/movie-review.
[8] *Trump Twitter Archive*, www.trumptwitterarchive.com/archive.
[9] Fatir, Abdul. “Abdulfatir/Twitter-Sentiment-Analysis.” *GitHub*, 21 Nov. 2019, github.com/abdulfatir/twitter-sentiment-analysis.