Political Ideology Detection in 2020 US Election Tweets

Annika Dahlmann, Ryan Song, Divya Ramamoorthy

University of Michigan

February 15, 2022

1 Introduction

Media sources play an influential role in many individuals political decisions. Specifically, Twitter seems to be a source in which people get a lot of their information from nearing election time. Objectivity is important in journalism, especially when reporting on political agendas that can influence everyday life for individuals. In this project, we aim to classify the ideological bias of a person's tweets regarding the United States 2020 Presidential election. Ideological bias for our purpose can fall into one of the three categories: conservative bias, liberal bias, or neutral. Ideology-detection has been an important and relevant task within the natural language processing space as people rely on media to seek fast and easy information to make political decisions. The rise of Twitter as a media source has introduced an increased possibility of such biases as not as much time or resource goes into curating Tweets and individuals are restricted in the number of characters to defend their point in a singular Tweet.

Previous work has highlighted certain biases that exist in common news sources. For example, Fox News stereotypically has more conservative bias. However, the goal of our project is to explore this same ideology bias classification with Twitter as our source. An example of such bias that can exist is how liberals use the phrase "estate tax" and conservatives use "death tax", two phrases to describe the same political statement where no neutral term exists. This will be the type of distinguishing we hope to extract from our Twitter data. Specifically, the input to our model when training will be a singular Tweet and the possible classes (conservative, liberal, neutral) and output the predicted class bias. When testing, we will only input unseen Tweets into our model and then output the predicted label.

2 Related Work

There has been a considerable amount of previous research done on sentiment analysis, specifically regarding political alignment classification in different forms of media.

Prior research on the subject includes work done in 2014 by researchers at the University of Maryland, who applied a Recursive Neural Network model to identify political positions suggested by given sentences. In their work, political annotations on both the phrase and sentence level were collected and used as training data. The use of Recursive Neural Networks captured syntactic and semantic composition together and did not rely on hand-made dictionaries or rule sets. The dataset contained a mix of ideological statements that were enriched, using the work of past research studies, to have a higher likelihood of containing bias and were balanced to contain an equal number of sentences corresponding to the two political parties in study. Given this labeled data, an element in the vector space representing a sen-

tence with liberal bias was distinct from the vector of a conservative-associated sentence. The cross entropy loss was minimized using optimal model parameters, and the main distinction between this research and prior studies was the eradication of bag-of-words modeling and hand-designed lexicon.

Our research explores a question similar to the work done by the Maryland researchers, but we plan on using Twitter data rather than crowd-sourced sentences. We also are using a Logistic Regression model instead of a Recursive Neural Network model to classify the political ideology evinced by Tweets. This angle differs from the research previously explained and our approach should provide a different angle at the research question.

3 Datasets/Environments

In order to learn our model we intend to use a dataset consisting of over 2.5 million 2020 US Election tweets, divided into tweets sent from politicians as well as non-politicians. We can easily train our classifier over this dataset because each tweet is annotated with the political alignment of the tweet itself, either Democratic or Republican. Tweets lend themselves well to training classifiers because they are short and quick to analyze–this means that we can more easily process a large number of tweets than say, news articles, and thereby train our classifier on a larger number of data points.

4 Methodology

Our group intends to use a Logistic Regression model in order to classify tweets by party affiliation. At a high-level we will implement the following procedure for training:

1. Pre-process data
2. Train LR model
3. Hyperparameter selection with Cross Validation

The first step of our training pipeline involves the pre-processing of our raw tweet data. We propose retrieving data from both the collection of tweets written by politicians as well as from the collection of tweets written by non-politicians in order to cover the largest variety of tweet authors possible from both sides of the political spectrum. When preprocessing data we intend to remove features such as URLs and tokenize the text strings into individual words. We also intend to commonly used words such as "and, is, a", etc., as these words do not tell us any useful information about the political alignment of the tweet. We also intend to tokenize each word, convert them to lowercase, and then convert each word into its stem so that we do not have separate words with the same meanings in our labels.

Afterwards, we will build a frequency dictionary, similar to HW1. We will take the tweet text and political affiliation as input and then count occurrences of each word in a dictionary object, where the key and value correspond to the word and count, respectively.

Following this we will pass our data into our logistic regression model, which will utilize a sigmoid function as well as a gradient descent function for the cost function. After extracting the features of each tweet we will train our model, and obtain a set of optimal weights. These optimal weights will serve as the final parameters for our trained model.

5 Evaluation

In order to properly analyze the quality of our model, we will have a portion of the data set strictly used for testing. This set will not have been used for any part of training the model to ensure our model is not overfit to the training data and will perform well on unseen data. We will use accuracy as the actual metric to evaluate the performance of our test data which will compare the number of times our predictions correctly matched the true labels assigned.

References

[1] A. Mashalkar. *Sentiment Analysis Using Logistic Regression and Naive Bayes*. Supervised ML Overview. 2020.

[2] *Political Ideology Detection Using Recursive Neural Networks*. University of Maryland Computer Science, Linguistics, iSchool, UMIACS. 2014.

[2] [1]