Politician Twitter Feed NLP Project Proposal

Triet Pham
Udacity Machine Learning Engineer Nanodegree
September 10, 2018

Domain Background

This project is an attempt at using current machine learning advancements in the field of natural language processing (NLP) to parse, understand, and predict text data from social media platform Twitter. This falls in the domain of social computing where we analyze the network and textual aspects to see the intersection of social behavior and computational systems. There have been many previous work done using the Twitter API and NLP to analyze tweets from different users. Among some interesting research on this social media research I found, is one that uses a part-of-speech tagger able to pick up on common acronyms or phrases found in the Twitter universe [1]. This tagger reveal interesting insight in the language and sentiment of different twitter netizens.

Problem Statement

I wish to try building a model that can take in tweets as input and predict whether that tweet is coming from a political Democrat or a Republican ideology. I would also like to see if we can we use Twitter data to gain political insight or possibly utilize such model to make presidential election forecast.

Datasets and Inputs

The dataset I'm using is from Kaggle, which contains tweets from various politicians and also information about which political party they represent. One column contains the tweet full string, another column contains the user handle name, and the last column contains the political party the user represent. This dataset is good for my goal since it has labeled data regarding political party of different tweet users.

The dataset was extracted from Twitter using the Twitter API (python wrapped) and publicly available on Kaggle. It contains the latest 200 tweets from various politicians.

Kaggle link: https://www.kaggle.com/kapastor/democratvsrepublicantweets [2]

Solution Statement

I will tokenize and use their tweets as features and the political party label as the target. With this, I should be able to derive some kind of NLP prediction model for predicting whether certain tweets are coming from Democrat or Republican representatives. I will also perform N-grams analysis and try to extract features importance from the trained model to see if we can gain any insight into political meta tweet.

Benchmark Model

Since this is a binary classification problem, a good benchmark model to start with is the logistic regression model. It is easy to implement and efficient at training. I can employ more complex algorithms or models afterward and compare it to the performance of the logistic regression model.

Evaluation Metrics

I will run cross validation using f1 as a scorer to benchmark the logistic regression model against other more complex algorithms or models.

The f1 score is a measure of how accurate a model is by taking the average between its precision and recall. As such, f1 score tells us how often a model is making correct prediction as well as how much of the data is the model able to cover with its accurate predictions.

The equation for f1 score is as follows:

$Precision = TP/TP+FP$

$Recall = TP/TP+FN$

$F1\ Score = 2*(Recall * Precision) / (Recall + Precision)$

Where TP = true positive, FP = false positive, and FN = false negative

Project Design

Since the Kaggle dataset is already cleanly provided, I will probably need little data cleaning and filtering. My first task will be to perform some exploratory data analysis. I wish to follow this guide on producing N-grams using Scikit-Learn CountVectorizer and how to perform N-grams analysis: https://towardsdatascience.com/hacking-scikit-learns-vectorizers-9ef26a7170af [3]

My second step will be to transform the generated N-grams to create a sparse matrix of features so that I can pass it into my machine learning model. At this stage, I will also split the data into training and testing set accordingly.

From here, I will apply my knowledge from the Udacity course to train, cross validate, and assess various prediction models performance. As mentioned earlier, I will cross validate all the models using f1 as the scorer to benchmark and compare them with each other.

I will take the best performing model and generate a normalised confusion matrix along with other metrics to show how well it can predict the politician party based on their tweets.

Moving on to the part I'm most excited to see will be the features importance extraction. With the trained model, I will analyze which N-grams features contributed most to the prediction. This will hopefully give us some insight into what kind of tweets or topics do politicians from certain party talk about.

[1] Archna Bhatia et co. 2013, www.cs.cmu.edu/~ark/TweetNLP/.

[2] https://www.kaggle.com/kapastor/democratvsrepublicantweets

[3] https://towardsdatascience.com/hacking-scikit-learns-vectorizers-9ef26a7170af