PREDICTING 2016 U.S. PRESIDENTIAL CANDIDATE SUCCESS USING TWITTER AND MACHINE LEARNING

Jennifer Steffens, Alexis Kulash, Eric Manley, and Timothy Urness
Department of Computer Science
Drake University
Des Moines, IA

ABSTRACT

Many researchers and pollsters are seeking to harness the vast amount of data available through various social media networks to make predictions and forecast trends in the population. Due to Twitter's rising prominence in the political arena—both professionally and personally—this research sought to determine if one could accurately predict U.S. presidential candidate success on a primary-by-primary basis by analyzing regional tweet data. This involves collecting geotagged tweets filtered for political discussion during the primaries in both Iowa and New Hampshire and then running various machine learning algorithms upon those tweets in order to make a prediction as to which candidates received more than 10% of the vote for each county in each particular state. It was concluded this prediction method was viable, however, the accuracy depended greatly upon the quantity of tweets collected. This work explores Twitter's usefulness as a potential prediction tool, and future studies will focus on making more precise election predictions on a larger context while comparing their accuracy and bias with actual polling data.

INTRODUCTION

Social media has revolutionized the way the general population expresses their beliefs and ideas. Platforms such as Twitter allow any user to publicly publish their opinions in the form of a tweet, where they can be consumed by others nationwide, instead of being contained to their immediate geographical area and social network. For data scientists, this innovation allows for the analysis of public opinion, and it has been used to forecast trends in the economy and population health [1]. In addition, Twitter has provided a thriving marketing opportunity for political figures, such as the presidential candidates, to solicit support.

Background

The most recent U.S. presidential race is infamous for the dramatically inaccurate predictions made by most major election forecast websites and news channels that relied upon traditional polling methods and has emphasized the need for alternative, more accurate polling methods [9]. A standard method for collecting polling data continues to be calling randomly selected phone numbers and inquiring about their voting intentions. However, modern phones will display to the user that the phone number calling is not from one of their stored contacts, and thus they are considerably less likely to answer the call in the first place. Furthermore, there has been a significant decline in the percentage of people willing to answer questions by phone: in the 1970's the average response rate was 80% – now, the average response rate is as low as 5% [11]. Combined with traditional polling's inability to intelligently identify likely voters or identify a significant

number of the respondent's opinions at a particular time leads to inaccuracies that are amplified by most major forecasting models [9].

In contrast, social media is flourishing. For Twitter alone, there are over 500 million tweets daily from over 100 million daily active users, which creates an ever-growing resource for public opinions [6]. There are, however, things to keep in mind when utilizing Twitter data in forecasts, as there are with any type of collected data. To use tweets as a predictive measure, it must be assumed that the tweets we have collected are trustworthy: not spam, false, or otherwise unreliable. Additionally, it is difficult to determine the age, gender, race, education, or income of the users whose tweets are included in the dataset, thus demographical biases cannot be accurately accounted for. Lastly, users are tweeting on a voluntary basis, therefore generally only those who are politically active and/or publicly vocal about their opinions produce the data collected [2]. With that considered, the aim of this study is to see whether machine learning algorithms trained on the tweets we collect can accurately predict whether a candidate was successful in winning more than 10% of the vote in a specified county.

Related Work

In 2010, several researchers at the Technical University of Munich made the first attempt to determine whether Twitter was a viable tool for online political deliberation and whether that deliberation also reflected offline political sentiment [13]. They collected over 100,000 tweets in the weeks before the federal election of the national parliament in Germany and applied Linguistic Inquiry and Word Count (LIWC2007) sentimental analysis software to extract each tweet's sentiment. The researchers were then able to make a prediction regarding the upcoming election based on the relative frequency of all combinations of two parties in all tweets mentioning more than one party. They were able to accurately predict the percentage of votes that went to each of the six main parties with a mean absolute error (MAE) of less than 2%. The research concluded that "the mere number of tweets mentioning a political party can be considered a plausible reflection of the vote share and its predictive power even comes close to traditional election polls" [13].

Daniel Gayo-Avello of the Universidad de Oviedo worked with Metaxas and Mustafaraj of Wellesley College in 2011 to perform an experiment to determine whether or not the results of past Twitter prediction models could be successfully replicated [2]. The researchers sought to use Twitter data to predict the results of several recent U.S. Congressional elections, although they pointed out that demographic bias was likely to be a key pitfall when relying on social media data (despite the results produced by some prior studies). For this study, they used the same methods as Tumasjan et al. and O'Connor et al. with some slight variations in order to account for the differences in the nature of electoral races in the U.S. versus Germany. Additionally, they chose to classify tweet sentiment as one of three categories (positive, negative, or neutral) versus marking a tweet as both positive and negative like O'Connor et al. allowed. Overall, the results from the study confirmed that the total number of tweets closely reflected the election outcome (similar to Tumasjan et al.) but the share of votes was often incorrect. They were only able to accurately predict 50% of the races. The researchers concluded that "the problem is that, in the past, some researchers have felt comfortable treating social media as a black box: It may give you the right answer, even though you may not know why." Gayo-Avello later articulated several recommendations for combatting issues

inherent to making election predictions with Twitter: check the influence that incumbency plays in the election; clearly define what is a vote (i.e. tweet); rely upon verifying credibility and sentiment analysis; and make sure to account for the silent majority [2].

More recently, researchers at Cardiff University and the University of Manchester created a baseline election-forecasting tool that relied upon Twitter data. They then used their tool to predict the results of the 2015 U.K. General Election [1]. In their study, they attempted to dispel some of the problems that Gayo-Avello identified in his research while also taking into account his advice. With that, they sought to make their predictions before the election had occurred—not afterwards—in order to make a "genuine" forecast. Additionally, the researchers adjusted their forecast to take into account tweet sentiment. Lastly, they calculated their predictions on seat rather than by vote share in order to recognize the existing distribution of parliamentary representation and party power. The researchers collected approximately 14,000,000 tweets and converted vote shares (predicted by sentiment) into seat forecasts. Ultimately, they were successful in more accurately predicting seat count than in the past whilst addressing several of Gayo-Avello's concerns.

METHODOLOGY

To construct the dataset, Twitter's Streaming APIs for Python were utilized to collect tweets continuously around the Iowa caucuses and the New Hampshire primary that Twitter labeled as originating from those states. The text of the tweet was stored as well as the coordinates of the Twitter user's location at that time, which are published automatically if a user has chosen to allow their location to be shared by Twitter. The coordinates are required to label tweets with their county of origin, allowing the matching of the tweet data with the corresponding polling results in that county. If a user had opted out of sharing their location, their tweets could not be used in this analysis, since it could not be know which county the user would have voted in, and thus these tweets were filtered out.

Once the developed program had placed all of the tweets in their correct county of origin, the tweet data was further filtered by topic and only the tweets in which the user had explicitly stated political keywords such as the presidential candidates' name, slogans, official campaign hashtags, or username (e.g. "Clinton", "@realDonaldTrump", "#FeelTheBern", etc.) were kept, as it was necessary the tweets utilized for the predictions were political and relevant. The program then labeled each remaining tweet with the candidate to whom it was referring and exported the dataset into state-specific CSV files that arranged the tweets by their candidate-county pairs (e.g. Clinton-Polk, Clinton-Adel). For the target column, the percentage of actual votes for each candidate in each county for Iowa was pulled from the CNN election results, and the New Hampshire percentages were pulled from the New Hampshire Secretary of State election results. This output was then joined on the tweet dataset using the county of origin and the candidate to which it pertained.

The Beautiful Soup library was utilized to extract the text from each tweet without any tags or markup and pass it to the program, which removed special characters and miscellaneous punctuation from the text. Common stop words (e.g. "an," "is," and "it") were then removed using the Natural Language Toolkit (NLTK) Stopwords Corpus**,** and the cleaned tweets were outputted as a list of strings. The CountVectorizer class was

then used to create a vocabulary composed of every word in the tweet dataset and model each tweet as a list of the number of times each vocabulary word appeared in the tweet.

Once the raw text had been transformed into numeric training features, machine learning algorithms were trained on the dataset. Three different classification algorithms were tested on the data: Random Forest, Support Vector Machine (SVM), and Multinomial Naïve Bayes (NB). The Random Forest model is an ensemble learning method that operates by constructing a specified number of distinct decision trees at training time and then returning the mode of all of the trees' predictions as the forest's prediction, and is known for its versatility and robustness. SVM is a discriminative classifier that constructs a hyperplane to separate the training data into categories and return its prediction for new examples, chosen because many text categorization problems are linearly separable and this model is designed to efficiently find those separators. In addition, SVM has a high dimensional input space and therefore can handle large feature spaces such as the ones generated by vocabulary word counts (Thorsten, 2002). The final algorithm tested was Multinomial NB, a model in the family of probabilistic classifiers based on applying Bayes' theorem with strong independence assumptions between the data features. Multinomial NB is a specialized version of Naïve Bayes that captures word frequency information in documents and is designed for analyzing text documents [4].

To test the algorithms, a small, randomly chosen portion of the entire labeled tweet and candidate polling result dataset from the New Hampshire primary was set aside to be used as a validation set, which the algorithms would not be able to see during the learning process. We then trained Random Forest, SVM, and Multinomial NB separately on the remaining training data, allowing them access to both the predictive features (candidate's name, county name, the word counts of the corresponding tweets), and the target column (the polling results of the candidate in that county). After they were fit on the training set, each algorithm was passed the predictive features of the validation set, but had no access to the target column of the validation set, and separately made a prediction for whether or not the value in the target column for each candidate-pair was above 10 percent. The predictions made by each algorithm were then compared to the actual values in the target column, and the number of correct answers and incorrect answers was recorded. This process was repeated multiple times, with different training and validation sets and sizes chosen each trial to determine the algorithm with the highest average accuracy, which would then be trained and tested on the Iowa caucus dataset in order to compare the two datasets' behavior and predictability.

RESULTS

All three algorithms performed well when the training set was high (greater than 85% of the total data) and testing set was low (15% or less), but the Multinomial NB algorithm consistently remained the most accurate prediction model as the size of the training set decreased, with an average accuracy of 93%, beating Random Forest and SVM's average accuracy of ~82% [Figure 1].

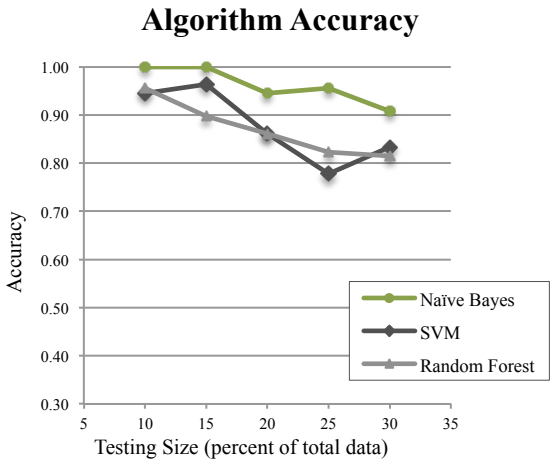

Figure 1: a comparison of the accuracy percentages of the algorithms

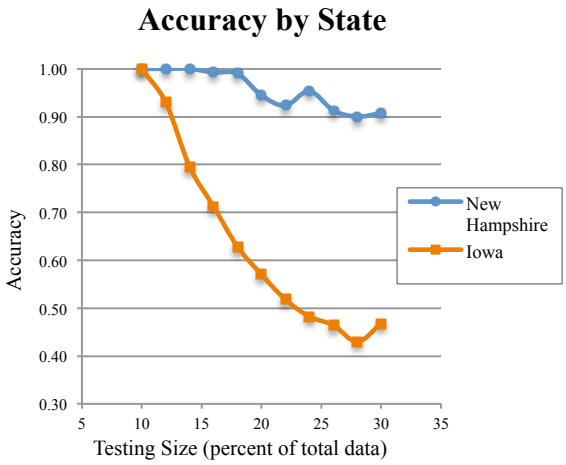

Figure 2: the accuracy percentages achieved by Multinomial NB by state

Because of this, the Multinomial NB algorithm was chosen for our comparison of the New Hampshire primary and the Iowa Caucus datasets. We found that the Iowa dataset was consistently less predictable, and if the size of the training set was less than 80% of the total Iowan dataset, the accuracy of the model was ~50%—equivalent to randomly guessing. In contrast, the model showed 90% accuracy on the New Hampshire dataset at the same training set size [Figure 2].

This drop in accuracy as the training set decreases and the is likely influenced by the number of usable tweets we were able to pull from the raw dataset streamed in using Twitter's API. Out of the entire raw dataset, only 17% were geotagged with coordinates to the tweet's location, and only 0.26% were geotagged and political in nature. In addition, the size of the filtered Iowa dataset was only 10% that of the filtered New Hampshire dataset, and the Iowan tweets had to be sorted into 99 counties, whereas the New Hampshire tweets only needed to be sorted into 10 counties. Furthermore, every county in New Hampshire had tweets originating from it, whereas in Iowa, due in part to the large amount of rural farmland, some counties did not have any tweets in the filtered dataset.

CONCLUSIONS

Training classification algorithms on Twitter data has been shown to be a viable technique for predicting candidate success on a county-by-county basis in a presidential election. Although the number of tweets collected dramatically affects the accuracy of the algorithms, the three algorithms tested performed well overall on the New Hampshire primary dataset.

Future Work

The effectiveness of utilizing Twitter data to forecast election results will be further investigated by implementing data analysis other than word count to train the algorithms with, such as sentiment analysis of the tweet text, the number of "retweets" of the tweet and the user's popularity, and the number of direct interactions between users and the candidate's accounts. In addition, the importance of identifying special characters such as emojis, hashtags, and usernames in the tweet text and weighing them differently than plain text will be evaluated. Lastly, the nature of the predictions will be expanded to include the final placing of the candidates in the presidential race as a whole.

REFERENCES

[1] Burnap, P., Gibson, R., Sloan, L., Southern, R., Williams, M., 140 Characters to victory?: using Twitter to predict the UK 2015 general election, *Electoral Studies,* 41, 230-33, 2016.

[2] Gayo-Avello, D., No, you cannot predict elections with Twitter, *IEEE Internet Computing*, 16, (6), 91-94, 2012.

[3] Joachims, T., *Learning to Classify Text Using Support Vector Machines,* Springer Science & Business Media, 2002.

[4] Kamal, N., Mccallum, A., Mitchell, T., Semi-supervised text classification using EM, *Semi-Supervised Learning*, 2006.

[5] Kaggle, Bag of Words Meets Bags of Popcorn, https://www.kaggle.com/c/word2vec-nlp-tutorial, published December 9, 2014.

[6] Lowe, L., 125 amazing social media statistics you should know in 2016, https://socialpilot.co/blog/125-amazing-social-media-statistics-know-2016/, published September 23, 2016.

[7] Metaxas, P. T., Mustafaraj, E., Gayo-Avello, D., How (not) to predict elections, *2011 IEEE Third Int'l Conference on Privacy, Security, Risk and Trust and 2011 IEEE Third Int'l Conference on Social Computing*, 2011.

[8] Mitchell, T. M., *Machine Learning*. New York City, NY: McGraw-Hill, 1997.

[9] Newkirk, V. R., What went wrong with the 2016 polls?, *https://www.theatlantic.com/politics/archive/2016/11/what-went-wrong-polling-clinton-trump/507188/,* published November 9, 2016.

[10] Secretary of State, 2016 Presidential Primary Election Results, http://sos.nh.gov/2016PresPrimElectResults.aspx.

[11] Sherwood, G., If traditional polling is dead, what's next?, *http://thehill.com/blogs/pundits-blog/presidential-campaign/306456-if-traditional-polling-is-dead-whats-next*, published November 17, 2016.

[12] Shi, L., Agarwal, N., Garg, R., Spoelstra, J., Predicting U.S. primary elections with Twitter, *Stanford Network Analysis Project (SNAP)*, 2012.

[13] Tumasjan, A, Sprenger, T. O., Sandner, P.G., Welpe, I.M., Election forecasts with Twitter - how 140 characters reflect the political landscape, *SSRN Electronic Journal,* 2010.

[14] CNN, 2016 Election Center, published May 14, 2016.