Prediction of US Presidential Primary Results Using Twitter; A Comparison of Naive and Lexical Predictive Approaches

Dale Flamm
Utah State University
Logan, UT, USA
dale.flamm@aggiemail.usu.edu

Sarbajit Mukherjee
Utah State University
Logan, UT, USA
sarbajit.usu@gmail.com

James McCabe Nickerson
Utah State University
Logan, UT, USA
j.m.nickerson@aggiemail.usu.edu

ABSTRACT

As the prevalence of Social Media applications has increased, the size of the data set that may be mined from Social Media has increased. On Twitter alone, over 500 million messages are written and sent every day. Even when analyzing only a small fraction of that data, it is easy to achieve statistically significant sample sizes. Naturally, this has resulted in attempts to analyze Social Media to determine trends on nearly every topic [11], [10].

However, when applied to election prediction, analyzing Twitter data has not had reliable results. There are many factors to consider, including the methods for data pruning, the viability, how appropriate a given metric may be, and, most importantly, how consistent the results for a given metric are. In our experiments we endeavor to test a wide range of feature metrics over multiple distinct elections with disjoint populations. To give a baseline for how appropriate and viable a feature is we compare the accuracy of predictions as well as the precision across multiple elections. We also classify our predictive features into two categories, Naive approaches and Lexical.

CCS Concepts

•**Data Mining** → Social Media Mining;

Keywords

Twitter; Data Model; Naive Bayes; Lexical Analysis, Data Mining

1. INTRODUCTION

The potential for using social media to predict political trends has become of increasing interest over the past decade, especially as participation in social media has risen. Traditionally, in political trend prediction there are a number of characteristics that are undesirable, such as the inability to obtain real time opinion changes, and a potentially smaller than desired sample size. Social Media makes it possible to automate collection of statistically significant sample sizes, perform appropriate data pruning and scrubbing, and analyze and make predictions [11], [10], [6].

There is, however, significant criticism regarding the use of Social Media data to predict elections. The methods used have varied greatly across research in terms of the data features analyzed and the methods used to analyze. Some of the most prominent research papers on this topic have made use of very simplistic methods that, when applied over multiple elections, achieve results only slightly better than chance [9], [5], [3], [4].

WOODSTOCK '97 El Paso, Texas USA

DOI: 10.475/123_4

2. PREVIOUS WORK

Among the earlier work on the feasibility of using Twitter Data to predict election outcomes is that of OConnor, Balasubramanyan, Routledge, and Smith. In their research, they made use of lexical analysis to determine positive or negative scores for each tweet in a Twitter data set. They then summed the positive and negative scores to compute a sentiment score. Finally, they performed time series analysis of the tweet sentiment across multiple topics relevant to the 2008 US Presidential Election. They reported strong correlation between Presidential approval rates and twitter sentiment, but not with electoral polls [2].

In a pair of works by Tumasjan, Sprenger, Sandner and Welpe, Mean Absolute Error was used to measure performance of Linguistic Inquiry and word count related to parties. In their work, they concluded a correlation between tweet count mentioning candidates, however this conclusion has since been strongly refuted for not considering all parties running, and a lack of relevant time window [5], [3].

In an article by Ceron, Curini, Iacus, and Porro, supervised sentiment analysis was applied to determine popularity of Itallina political leaders during the 2011 French Presidential and legislative elections. They collected tweets during a 1 week window before the elections, and performed sentiment analysis to predict vote rates by candidate and by political party. They were able to reasonably predict the vote rates of multiple elections, however their Mean Absolute Error was worse than traditional polling methods. They did identify one potential issue regarding self-selection bias,

stating that vote rates of left ideological parties were overestimated, and left leaning was underestimated [11].

In further works, other data features have been analyzed, including but not limited to, number of users mentioning a candidate, count of users only mentioning a single candidate, elimination of candidate promotional accounts, and the number of tweets containing hyperlinks. Recently there have been attempts to obtain regression models using preelectoral polling combined with Twitter data. It is important to note that all of these methods reported mixed results, or were refuted later based on poor performance on future elections [10], [9], [4], [1].

Since the data used in these works were collected before 2010, and there has been a dramatic change for social media since, it is worth reevaluating the predictive ability of social media in political issues such as elections. In order to find opinions about political issues, we investigated the relationship between public opinion towards republican presidential candidates during the primary election period of 2016. Millions of tweets were collected and analyzed from Twitter website through public available Twitter API since March 2016. Twitter allows users to post short messages (up to 140 characters) that are publicly visible through the Internet. By capturing tweets mentioning each presidential candidate and analyzing the sentiments behind those tweets, we could track people's opinions about each candidate and thus predict the final primary election results.

The rest of this paper is organized as follows: First, we will briefly introduce our data collection method and evaluate the traditional methods. Second, we will discuss the data cleaning and initial evaluation method of the tweet we used in our model. Third, we will introduce the features that our data suggest correlate with the election and propose our prediction model. Then, our prediction results will be presented with comparison to traditional poll results and election results. Finally, we will make conclusions from our experiments and propose our future research.

3. CHARACTERIZATION OF TWITTER BASED ELECTORAL PREDICTION METHODS

Twitter data have been mined to determine the public opinion on several topics and pre–electoral and electoral polls have been studied as part of that public opinion. The number of reports on this issue has grown considerably, most claiming promising to positive results.

It must be noted that it is commonly implied that any method to predict electoral results from Twitter data is an algorithm. Such algorithms are devised as a pipeline that starts with the collection of data from Twitter, goes on processing that data, and finishes with a prediction that needs to be evaluated against the actual results of the elections. Needless to say, the algorithms can be parameterized to adapt to different scenarios, and predictions can be more or less detailed (for instance, the algorithms can provide just the winner, or they can provide vote rates for different candidates). Thus, there are a number of features defining any method to predict electoral results from Twitter;namely:

1. Period and method of collection: that is, the dates when tweets were collected and the parameterization used to collect them.
2. Data Cleansing measures:
   - Purity: that is, to guarantee that only tweets from prospective voters are used to make the prediction.
   - Debiasing: that is, to guarantee that any demographic bias in the Twitter user base is removed.
   - Denoising: that is, to remove tweets not dealing with voter opinions (e.g., spam or disinformation) or even users not corresponding to actual prospective voters (e.g., spammers, robots, or propagandists)
   - Prediction method and its nature:
     - The method to infer voting intentions from tweets.
     - The nature of the inference: that is, whether the method predicts individual votes or aggregated vote rates.
     - The nature of the prediction: that is, whether the method predicts just a winner or vote rates for each candidate.
     - Granularity: that is, the level at which the prediction is made (e.g., district, state, or national)
3. Performance evaluation: that is, the way in which the prediction is compared with the actual outcome of the election.

So, in short, any method to predict elections from Twitter data should take into consideration the four different aspects: (1) the data collection approach – which could be focused on candidates and parties or could be much broader; (2) the approach taken to deal with noise – that is, removing most of it or accepting its presence; (3) the method of prediction – for example, giving the winner of the race, the vote share; and (4) the evaluation metrics to be used.

4. DATA COLLECTION AND DESCRIPTION

The first question we should address is where and how to collect data. We chose to collect data using Twitter because it's very popular in the United States of America, largely because of its 140 character tweeting capability where people can simply use their smartphones or desktops to tweet about different topics. We managed to use Twitter streaming API in order to collect tweets in real-time. We filtered twitter stream in order to only collect tweets related to US 2016 election. For such, we were searching for political key words such as 'donald trump', 'ted cruz', 'hillary clinton' and 'bernie sanders'. We wrote a simple listener in Python for collecting political tweets. This script has run on one of our servers since March 18, 2016. After 1.5 months we have collected around 50 millions tweets, which is significantly large. Even with the max tweet size of 140 characters, the memory needed to collect and store this much data creates a significant challenge.

Our approach focuses on two phases, first collection and pre–processing, then analysis. In the first phase we collected data for the top two candidates for both the Democratic National Primaries, and the Republican National Primaries. This allowed us to collect significant quantities of data for

Table 1: Correlation between the tweet volume and the RCP score

	Trump	Cruz	Sanders	Hillary
Correlation	0.75	0.32	0.45	0.62

more than a dozen primary elections each with distinct and non-overlapping populations.

By testing our methods across multiple distinct elections, we hope to use this as a control against the possibility of obtaining better results than should be expected due to chance. This also gives us the ability to measure precision over multiple elections, determining how often each method is correct, or the degree of error in each method .

We removed spammers from our collection of tweets by not including any tweets created by users who had tweeted the exact same text as another user. We then filtered out the tweets based on the user location for the four days prior to the primary and on the day of election. After pruning, we separated the data set by State Primary for per-candidate analysis.

5. INITIAL EXAMINATIONS AND OBSERVATIONS

This section describes our preliminary observations:

5.1 Methods of Prediction

Initial predictions were computed based on Twitter volume as in [10] and sentimental analysis as in [6] with slight modifications. The prediction results were then directly compared against the poll results in Realclearpolitics website (*www.realclearpolitics.com*). Realclearpolitics website calculates poll results (RCPscores) by taking the average of poll results from several popular media and independent survey institutes. RCP scores are the closest public opinions to the presidential candidates. Thus in this paper, we are using the RCP score as our baseline to compare our prediction results.

5.2 Results of the Prediction Methods

As shown in Table 1, the tweet volume has a strong correlation with poll result for some candidates, but there is less correlation for the other candidates. This result conflicts with what Oconnor and Tumasjan found in [6] and [10]. One reason is that they only evaluated two candidates in their experiments. The other reason may be that Twitter had not been widely used as a platform for discussing presidential elections when Oconnor and Tumasjan did their experiments.

To further analyze the capability of tweets, we also evaluated the prediction results by using sentimentally analyzed tweets similar to the method used in [6], [10], [2]. We found that the predicted results using only sentimental analyzed tweets, or the volume of positive tweets, was very similar to using only the tweet volume. Additionally, in our experiments, we had similar conclusions with Gayo-Avello in [2], that by using the volume of tweets or positive tweets, we can't predict the election result correctly. In order to predict public opinions towards presidential elections precisely, we need to create new models. In the following sections, we will illustrate the model we built to solve this problem.

5.3 Sentiment Analysis of Tweets

In order to analyze the public perception of candidates, we used statistical analysis and machine learning techniques to analyze the collected tweets. Thus, we decided to run sentiment analysis on top of the collected data in order to find out tweets polarity (if they are positive or negative). We used the *'Naive Bayes'* classifier in order to compute the sign of tweets (+/-) where positive means that a tweet supports the corresponding candidate and negative mean that a tweet does not support the corresponding candidate. We use the NLTK Python package for NLP phase. To train the classifier we used the data collected by people from Cornell University where they have published polarity data for IMDB reviews.

We faced a few challenges while doing sentiment analysis. First, the dictionary words people use to review movies differs from the language people use to express their opinion regarding a political candidate, especially when limited to 140 characters. Thus, the first question is how accurate our prediction will be if we use the IMDB review dataset for training the classifier. Let's check an example tweet: Tweet: ''If Cruz doesn't win I'm leaving the country".

If we feed the above tweet to our naive bayes classifier, it gives it a 'negative' for Cruz; however, the above tweet is actually positive for Cruz. The reason our classifier failed to detect the correct polarity is that while the whole meaning of sentence is negative, however the view toward the noun is positive. To fix this issue, we should use more sophisticated classifier than Naive Bayes. From this example, we demonstrate the need for an algorithm that not only considers the polarity of words for computing the whole tweet polarity, but it also has to take into account the relationship between words and the roles of words in the sentence.

5.4 Why naive sentiment analysis algorithm may fail?

The training dataset is IMDB review and the testing dataset is political tweets. These two datasets have different probability distributions. The IMDB reviews are written by expert people in movie industry where they use different words (dictionary) to explain their lengthy opinions about movies. However, tweets are written by normal people and they are very short usually one sentence (less descriptive). This makes it much harder for the algorithm to detect the label of tweets correctly.

In the case of the algorithm using the naive bayes classifier, it just considers words independently and ignores the relationship between words. This means that it ignores what role this word plays in a sentence. The example that in the previous subsection highlights our point. Even if the sentence is negative by itself but considering the role 'Cruz' word plays in the sentence (grammatical role), it is a positive sentiment for him!

5.5 Simplistic Methods for Comparison

To compare with our more complicated techniques, we implemented some simplistic features to indicate candidate popularity. These features are:

- Number of Tweets
- Number of times the tweets are favorited
- Number of times the tweets are retweeted
- Number of followers of the tweet authors

These values of these features were normalized and computed using weight and without using the weight. These features will provide a baseline to compare our methods with.

6. OUR APPROACH

The objective of sentiment classification is to predict the overall sentiment orientation conveyed in a piece of text such as a user review, blog post or editorial. Previously proposed supervised learning techniques using in–domain data do not scale well across different domains: words that make good predictors within a domain are not easily generalized, for example in the case of actor or director names being good predictors of author opinion on film reviews and thus making useful features to train a classifier but which naturally will have limited applicability on an unrelated domain. In this context, interest on techniques that rely less on domain knowledge has grown considerably. These include methods that leverage properties of natural language, discourse analysis and lexicons.

6.1 Opinion Lexicons

Opinion lexicons are resources that associate sentiment orientation and words. Their use in opinion mining research stems from the hypothesis that individual words can be considered as a unit of opinion information, and therefore may provide clues to document sentiment and subjectivity. We would derive opinion terms from the SentiWordNet database of terms and relationships, typically by examining the semantic relationships of a term such as synonyms and antonyms. Lexicons built using this approach would be applied to sentiment classification of tweets.

6.2 SentiWordNet

SentiWordNet could be a valuable resource for performing opinion mining tasks since it provides a readily available database of term sentiment information for the English language. This means SentiWordNet can be a replacement to the process of manually deriving lists of terms containing sentiment information for opinion mining tasks. It can also be noted that SentiWordNet is built from a semi automated process that derives opinion information from the WordNet database, and has the potential to be applied to documents on different domains. The semi-automated approach also indicates the process can easily be replicated on other languages, where lexicons similar to WordNet are available. Thus, SentiWordNet offers potential benefits to opinion mining and to the task of sentiment classification in particular. For each term in WordNet, a positive and a negative score ranging from 0 to 1 is present in SentiWordNet, indicating its polarity, with higher scores indicating terms that carry heavy opinion bias information, whereas lower scores indicate a term being less subjective In our approach we used 'SentiWordNet v3.0'.

6.3 Designing Features using SentiWordNet

In order to use SentiWordNet as a tool for performing sentiment classification, a set of features that capture as much sentiment information as possible from textual documents needs to be devised. Then, once a feature set is generated from text documents with SentiWordNet, these can be used as input to a classifier algorithm and results on classification performance and execution speed can be analyzed.

Table 2: SentiWordNet Database Record Structure

Field	Description
POS	Part of speech associated with synset. This can take four possible values: a=adjective, n=noun, v=verb, r=adverb
Offset	Numerical ID which associated with part of speech uniquely identifies a synset in the database.
PosScore	Positive score for this synset. This is a numerical value ranging from 0 to 1.
NegScore	Negative score for this synset. This is a numerical value ranging from 0 to 1.
SynsetTerms	List of all terms included in this synset.

6.3.1 The SentiWordNet Database

SentiWordNet is a database containing opinion scores for terms derived fro m the WordNet database version 2.0. It is built using a semi-supervised method to obtain opinion polarity scores from a subset of seed terms that are known to carry opinion polarity. Each set of terms sharing the same meaning, or synsets, is associated with three numerical scores ranging from 0 to 1, each indicating the synset's objectiveness, positive and negative bias. One important characteristic of SentiWordNet is that positive and negative scoring is graded for any given term, and it is possible for a term to have non-zero values for both positive and negative scores, according to the following rule: For a synset *'s'*, we define

- Pos(s) → Positive score for synset 's'
- Neg(s) → Negative score for synset 's'
- Obj(s) → Objectiveness score for synset 's'

Then the following scoring rule applies:

- Pos(s) + Neg(s) + Obj(s) = 1

The SentiWordNet database is provided as a text file where term scores are grouped by synset and the relevant part of speech. The table below describes the columns for one entry in the database reflecting opinion information of a synset.

6.4 Feature Generation

After analysing the database structure of SentiWordNet, this section explores key aspects that need to be taken into consideration when designing features to be used in sentiment classification.

6.4.1 Part of Speech Tagging

Data in SentiWordNet is categorized according to part of speech, but there are considerable differences in the level of objectiveness a synset might carry, depending on its grammatical role. Information on part of speech in the source documents being classified will need to be extracted, so that SentiWordNet scores can be accurately applied. To achieve this, a part-of-speech tagging algorithm can be employed to automatically classify words into categories based on parts of speech from the source documents. Each word in a sentence was associated with a relevant tag indicating its role

in the sentence, such as verb, noun, adjective, etc. Several standards exist for tag formats, of which the most popular are related to the *Penn Treebank annotated corpus*.

6.4.2 Word Sense Disambiguation

When evaluating scores for a given term using SentiWordNet, an issue arises in determining to what specific WordNet synset the term belongs to and which score to take into account. A simple approach is taken to solve the disambiguity:

- Evaluate scores for each synset for a given term
- If there are conflicting scores – e.g. positive and negative scores exist for the same term, then calculate the average of all positive scores and all negative scores
- Return the averaged SentiWordNet score with higher value only if the positive and negative scores differ by more than a given threshold

6.4.3 The Polarity Data Set

The polarity data set is a set of film review documents available for research in sentiment analysis and opinion mining. It was first introduced as a research data set along with Bo Pang and Lillian Lee's initial results on machine learning methods for sentiment classification presented in [8]. The most recent available data set is version 2.0, and is the one being used for this dissertation's experiment. It comprises 1000 positive labeled and 1000 negative labeled film reviews extracted from the Internet Movie Database Archive [7]. In this section, the polarity data set is further evaluated with considerations on how SentiWordNet can be used to extract opinion bias information from documents contained in it. The data from the polarity dataset had to undergo some pre-processing:

- All text is converted to lowercase.
- Each line in a document corresponds to a single sentence.
- All HTML tags are stripped from the document – e.g. documents are plain text.
- Ratings information is removed from the data set since author bias should be indirectly implied from the text, and not from the rating scale given.
- Removal of stop words.

6.5 Features for sentiment lexicons

Following are the features that were derived from the sentiment lexicons.

6.5.1 Overall Scoring per Part of Speech

Intuitively, the overall positive and negative scores for all terms in a document extracted from SentiWordNet terms can be taken as a measure of opinion polarity.

6.5.2 Positive and Negative Ratios

For each part of speech, this metric calculates the percentage of positive and negative occurrences out of total terms found, to give an indication of positive and negative term usage within the document.

6.5.3 Scores per Document Segment

To evaluate the contribution of individual document areas to overall sentiment, each document is divided into N segments of equal size, and for each segment the total positive and negative scores for a given document segment, per part of speech. For the case of adjectives, other metrics such as strength and ratios are to be calculated for each segment.

6.6 Why use sentiment lexicons?

One advantage of this method is that, unlike supervised learning approaches, it can be applied to text with no training data required, making it an interesting alternative for cases where training data is non existent or when we're dealing with data from multiple domains. In particular, it is an interesting strategy (although not the only one) for dealing with domain dependence problems seen on supervised learning sentiment classification methods.

7. SENTIMENT ANALYSIS USING LEXICON

We want to explore lexicon-based approaches to sentiment classification. These methods use a language resource - a sentiment lexicon (SentiWordNet v3.0) - that associates words and expressions to an opinion polarity - usually a numeric score, representing common knowledge about this word's opinion.

We use SWN3.0 lexicon as a class, which does two things upon initiation: reads a specific language resource into memory (in this case SentiWordNet v3.0), and compiles word frequency data based on the frequency distribution of lexicon words in NLTK's Brown corpus.

Using this class we can query a word when used as a specific part of speech (adjective, verb, noun and adverb). These methods return a tuple of numeric values (positive, negative) indicating word polarity known to the lexicon. Another simplification implemented here applies when words carry multiple senses: the opinion of each of those is averaged out to obtain the output tuple - ie. other than separating words by part of speech, no word sense disambiguation takes place.

7.1 How to classify test with a sentiment lexicon?

We scan a document counting the words found in the lexicon, and make a classification decision based on the total counts for positive and negative words found. SentiWordNet contains words part-of-speech information, so it makes sense to perform some pre-processing on input text: we'll use NLTK's part-of-speech tagger to find out what part-of-speech to query in the lexicon for each word found.

The output of this script is a tuple (positive, negative) containing the overall positive and negative sentiment for the document (in the above example, we classify the document as positive). It also generates an annotated version of the input document with part of speech tags [7] and additional tags indicating sentiment scores for each word.

7.2 Use of Hash Tags

We used *hashtag* count to assign a weighted score to each tweet on the premise that individuals over 35 are more likely to vote, and less likely to use hashtags in their tweets. This premise is not completely confirmed in our opinion, and

we are using it to compare our results that use the weight against the Naive classifiers. We are doing this in order to explore *hashtag* count as a better control against the difference between the demographics that use Twitter and demographics that vote.

7.3 Evaluation Metrics

For each tweet, we ran our system and calculated the respective positive and negative score. For example, the following sentence ***If Cruz does not win I am leaving the country***, gives a result *'negative'* when we run using Naive Bayesian Classifier. The same sentence when run in our system gives the following score as the result; $(0.1899, 0.0460)$, which says that the above statement is more positive then negative, which is essentially correct and shows the correctness of our proposed system.

But only using the positive and negative score as our evaluation metric did not serve us any good, as we were missing some vital information regarding the use of adjectives in the tweets and also use of hash tags in the tweets. To us the use of adjective in the tweet carries a significant weight in the evaluation as we can infer from the use of 'descriptive' words, as to how likely that person is likely to vote for the candidate. Thus, along with the positive and negative score from the tweets we also collect the positive and negative score of the adjectives used in the tweets.

We define *'SuccessScore'* as an evaluation metric for our system. It is measured for each state on per candidate basis. The calculation steps are as follows:

1. Obtain positive and negative scores of each tweet as well as from the adjectives used in each tweet. Let those variables be 'posTweet','negTweet','posAdj', 'negAdj' respectively.
2. Calculate the weights to be used for each tweet based upon the number of hashtags used.
3. Calculate the weighted score for each tweet. Let the new variables be 'wtposTweet','wtnegTweet','wtposAdj', 'wtnegAdj' respectively.
4. Calculate *'SuccessScore'* as follows: SuccessScore = (wtposTweet+wtnegTweet)-(wtposAdj+wtnegAdj)

We next define *'CandidateVotePercent' (CanVotePer)* to evaluate our system. It is measured for each state taking into consideration both the participating candidates of the same party. The evaluation steps are as follows:

1. For each state and same party let the *'SuccessScore'* (SuCC) be $SuCC_{cand1}$ and $SuCC_{cand2}$ respectively.
2. $CanVotePer_{cand1} = \frac{SuCC_{cand1}}{SuCC_{cand1}+SuCC_{cand2}} * 100\%$

   $CanVotePer_{cand2} = \frac{SuCC_{cand2}}{SuCC_{cand1}+SuCC_{cand2}} * 100\%$
3. $CanVotePer_{cand1}$ and $CanVotePer_{cand2}$ gives us the percentage of the votes each candidate got on per state basis. The candidate with the higher percentage of votes is the winner.

In our next step we compared our results against the poll results in Realclearpolitics website *www.realclearpolitics.com* based on the percentage of vote received and also the actual winner against our predicted winner.

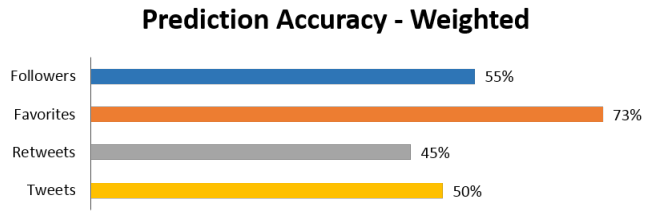

Figure 1: Accuracy rates for the simplistic features.

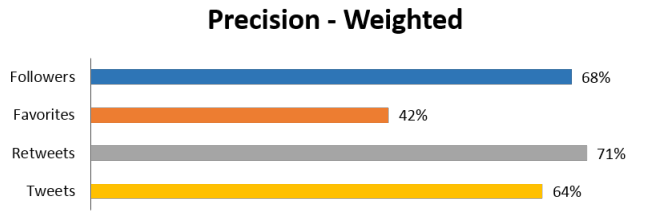

Figure 2: Precision rates for the simplistic features.

Along with the above metrics we use 'accuracy' and 'precision' to further evaluate our system. We define the *accuracy* as the percentage of time the candidate with the majority of the votes is correctly identified. The *precision* is the difference between the maximum error and the minimum error subtracted from 1. The *error* is the average difference between the election results and our system's prediction.

8. RESULTS

We collected data from Twitter starting 18^{th} March until 26^{th} April, 2016. In this period we had the presidential primaries of the following 14 states: 'alaska', 'arizona', 'hawaii', 'idaho', 'northdakota', 'utah', 'washington', 'wisconsin', 'wyoming', 'newyork', 'connecticut', 'delaware', 'maryland', 'pennsylvania', 'rhodeisland'. We collected around 50 million tweets to evaluate our proposed method. Table 3 gives a detail description of our results. We compared our system against the results obtained from *'Real Clear Politics'* and also against 'Bayesian Analysis'.

8.1 Simplistic Methods

Figures 1, 2 and 3 illustrate the impreciseness of the simplistic approach along with the inaccuracy and high error rates. The simplistic features are no better than randomly picking which candidate would win.

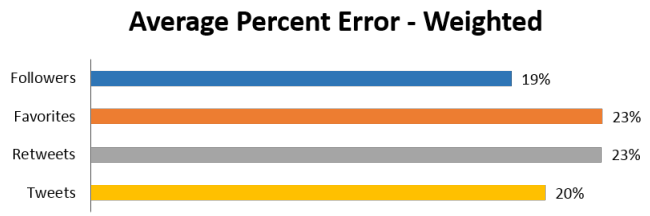

Figure 3: Error rates for the simplistic features.

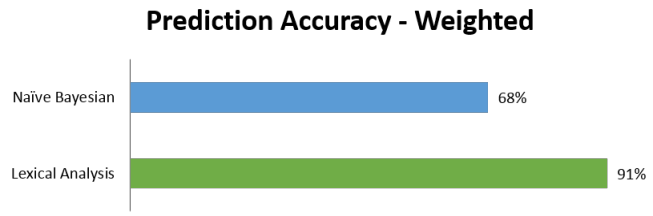

Figure 4: Comparison of the accuracy between the lexical analysis and the naive bayesian.

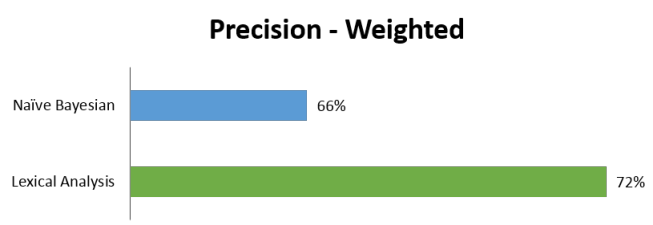

Figure 5: Comparison of the precision between the lexical analysis and the naive bayesian.

8.2 Lexical Analysis

Our lexical approach was also compared to a naive bayesian classifier. Figures 4, 5 and 6 show how the lexical analysis was able to achieve 91% accuracy while the naive bayesian is not much better than the simplistic approaches. The precision was on-par with the other approaches and the error was the lowest using our lexical analysis.

9. CONCLUSIONS

Social media provides a platform that should be ideal for gathering public sentiment. Using social media for predicting elections is a logical application of such large amounts of data. Previous attempts to predict elections using social media have shown it a very difficult task. Two significant challenges are identifying users who are likely to vote, and the complexity of reliable sentiment analysis. In this paper we present a method for removing users who are unlikely to vote and a robust sentiment analysis using lexicons to address these issues. We have shown how the sentiment lexicon analysis described in this paper was able to correctly pick the state winner with 91% accuracy. There are a number of interesting ways that this research could be expanded. It would be beneficial to use our approach on more elections, especially elections in other countries where social media may be used differently. A more comprehensive method for determining the likelihood that a given twitter user would vote or not would likely improve the accuracy of our methods.

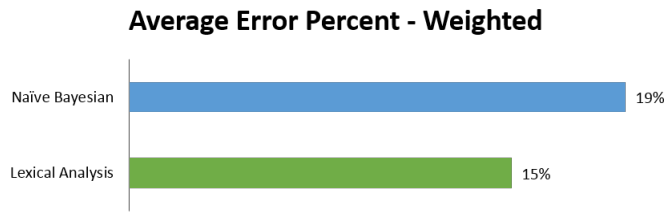

Figure 6: Comparison of the error between the lexical analysis and the naive bayesian.

10. REFERENCES

[1] A. Ceron, L. Curini, S. M. Iacus, and G. Porro. Every tweet counts? How sentiment analysis of social media can improve our knowledge of citizens' political preferences with an application to Italy and France. *New Media & Society*, 16(2):340–358, Mar. 2014.

[2] D. Gayo-Avello. "i wanted to predict elections. *CoRR*, abs/1204.6441, 2012.

[3] A. Jungherr, P. Jürgens, and H. Schoen. Why the pirate party won the german election of 2009 or the trouble with predictions: A response to tumasjan, a., sprenger, t. o., sander, p. g., & welpe, i. m. "predicting elections with twitter: What 140 characters reveal about political sentiment". *Soc. Sci. Comput. Rev.*, 30(2):229–234, May 2012.

[4] H. ME. Multi-cycle forecasting of congressional elections with social media. pages 23–30, 2013.

[5] P. T. Metaxas, E. Mustafaraj, and D. Gayo-Avello. How (not) to predict elections. In *SocialCom/PASSAT*, pages 165–171. IEEE, 2011.

[6] B. O'Connor, R. Balasubramanyan, B. R. Routledge, and N. A. Smith. From tweets to polls: Linking text sentiment to public opinion time series. In *ICWSM*. The AAAI Press, 2010.

[7] B. Pang and L. Lee. A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In *Proceedings of the 42Nd Annual Meeting on Association for Computational Linguistics*, ACL '04, Stroudsburg, PA, USA, 2004. Association for Computational Linguistics.

[8] B. Pang, L. Lee, and S. Vaithyanathan. Thumbs up?: Sentiment classification using machine learning techniques. In *Proceedings of the ACL-02 Conference on Empirical Methods in Natural Language Processing - Volume 10*, EMNLP '02, pages 79–86, Stroudsburg, PA, USA, 2002. Association for Computational Linguistics.

[9] E. T. K. Sang and J. Bos. Predicting the 2011 dutch senate election results with twitter. In *Proceedings of the Workshop on Semantic Analysis in Social Media*, pages 53–60, Stroudsburg, PA, USA, 2012. Association for Computational Linguistics.

[10] A. Tumasjan, T. Sprenger, P. Sandner, and I. Welpe. Predicting elections with twitter: What 140 characters reveal about political sentiment. In *Proceedings of the Fourth International AAAI Conference on Weblogs and Social Media*, pages 178–185, 2010.

[11] H. Wang, D. Can, A. Kazemzadeh, F. Bar, and S. Narayanan. A system for real-time twitter sentiment analysis of 2012 u.s. presidential election cycle. In *Proceedings of the ACL 2012 System Demonstrations*, ACL '12, pages 115–120, Stroudsburg, PA, USA, 2012. Association for Computational Linguistics.

Table 3: Results showing percentage of votes obtained: Comparison with RCP and Naive Bayes

State	Candidates	Real Clear Politics	Our Result	Naive Bayesian
ALASKA	Clinton	18.4	47.82	57.68
	Sanders	81.6	52.17	42.31
ARIZONA	Clinton	57.6	48.15	52.37
	Sanders	39.9	51.84	47.62
	Trump	65.41	45.87	41.91
	Cruz	34.58	54.12	58.08
HAWAII	Clinton	30	44.39	50.46
	Sanders	69.8	55.6	49.53
IDAHO	Clinton	21.2	15.33	36.73
	Sanders	78	84.66	63.26
UTAH	Clinton	20.3	13.77	32.81
	Sanders	79.3	86.23	67.18
	Trump	16.8	18.14	25.36
	Cruz	83.17	81.85	74.63
WASHINGTON	Clinton	27.1	32.27	59.29
	Sanders	72.7	67.72	40.7
WISCONSIN	Clinton	43.1	40.55	40.25
	Sanders	56.6	59.44	59.74
	Trump	42.13	41	41.76
	Cruz	57.86	58.99	58.23
WYOMING	Clinton	44.3	69.28	56.57
	Sanders	55.7	30.71	43.42
NEW YORK	Clinton	58	59.5	57.47
	Sanders	42	40.4	42.52
	Trump	80.64	67.58	45.76
	Cruz	19.35	32.41	54.23
CONNECTICUT	Clinton	51.8	59.03	64.53
	Sanders	46.4	40.96	35.47
	Trump	83.18	77.99	50.51
	Cruz	16.81	22	49.48
DELAWARE	Clinton	59.8	67.86	52.13
	Sanders	39.2	32.13	47.86
	Trump	79.26	50.05	49.1
MARYLAND	Clinton	63	58.29	54.64
	Sanders	33.2	41.7	45.35
	Trump	74.21	87.17	53.88
	Cruz	25.79	12.8	46.11
PENNSYLVANIA	Clinton	55.6	65.77	58.51
	Sanders	43.6	34.22	41.48
	Trump	72.41	59.44	47.74
	Cruz	27.58	40.55	52.25
RHODE ISLAND	Clinton	43.3	31.52	63.06
	Sanders	55	68.47	36.93
	Trump	85.98	73.33	51.35
	Cruz	14.01	26.66	48.64