Political Sentiment Analysis 2016 U.S. Elections using Tweets

Anton Kozačkov, Elisabeth Kräman, Julia Holst, Marta Turek, Will Chien

University of Amsterdam
Fundamentals of Data Science (5294FUDS6Y)
Assignment 1 - Group 5

Abstract. This report aims to determine the efficacy of using Twitter data to analyse public sentiment toward the 2016 U.S. presidential candidates via topic modelling, sentiment analysis and correlation analysis. The findings suggest that Latent Dirichlet Allocation (LDA) topic modelling does not clearly categorise presidential candidates, Donald Trump and Hillary Clinton, nor can tweets be effectively grouped as positive or negative. The Naïve Bayes classifier technique was used to conduct sentiment analysis and hashtags were used to automatically label tweets as positive, negative or neutral. There was little correlation found between tweet sentiment data and voting outcomes, suggesting that Twitter should be used cautiously as a basis for predicting the outcomes of elections.

Keywords: Political Sentiment Analysis · Topic Modelling

1 Introduction

The 58th U.S. presidential election has been named as one of the most controversial in history, where both the Republican candidate Donald Trump and the Democratic nominee Hillary Clinton were perceived as unfavourable by the general public. Social media platforms such as Twitter were heavily utilised by both political camps and voters to influence public opinion and provoke discord. Twitter, a social media service that is a popular method for expressing opinions in real-time is well suited for sentiment polarity detection. Analysing historic data further allows the measurement of Twitter's predictive power against de facto voting outcomes. At its core, this report summarises the results of topic modelling and sentiment analysis. The trained sentiment model is used to find correlations between tweet sentiment and external datasets containing demographic data and voting outcomes of the 2016 U.S. election.

1.1 Related Work

Various machine learning techniques can be used to conduct sentiment and correlation analysis on the Twitter data as illustrated by the comprehensive literature body. Previous related studies conduct various analyses, including sentiment analysis, polarity and subjective classifications,opinion and event correlation or attempt to predict future events [7]. As illustrated by Barnaghi, Ghaffari and Breslin [2], a sentiment analysis can only be conducted on preprocessed data, regardless of the study-specific purpose. This entails the tokenization of sentence string into a bag of words, where each word can be used to train the classifier [2]. Common preprocessing techniques also include the stemming of words to their root, removing punctuation, converting all text to lowercase, and removing high frequency but meaningless terms such as articles, which are coined stop words [10]. Subsequently, sentiment analysis can be conducted with supervised and unsupervised machine learning techniques. To compile sentiment words, a lexicon approach, manual approach or corpus based approach can be taken [10]. In an attempt to overcome inherent ambiguity in human language, a study on the 57th U.S. elections utilised annotators to classify the tweets in the baseline sentiment model [13]. Previous experiments suggest that part-of-speech features, which objectively describes text attributes such as the number of adjectives and verbs, are less effective compared to a combination of n-grams, a readily available lexicon and micro blogging features, which can be emoticons, abbreviations and intensifiers [6].

Challenges of political sentiment analysis using twitter include the imbalance of activities between users, general human language complexity and data sparsity, multilinguality, and bot activity [12] [11]. As pointed out by a sentiment analysis study using Twitter data on the Irish General elections, the share of volume of tweets per party displayed highest predictive quality, and was only marginally improved by the share of positively connotated tweets [3]. Other studies underpin that the predictive power of sentiment analysis is limited: [4]

2 Methodology

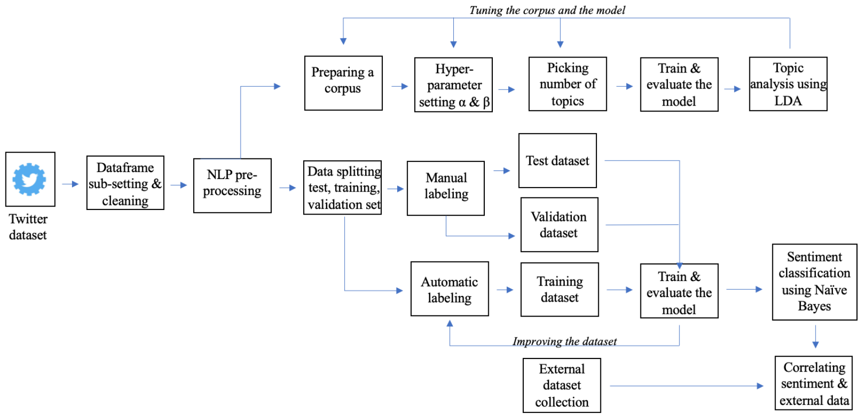

Tuning the corpus and the model
Preparing a corpus
Hyper-parameter setting α & β
Picking number of topics
Train & evaluate the model
Topic analysis using LDA
Twitter dataset
Dataframe sub-setting & cleaning
NLP pre-processing
Data splitting test, training, validation set
Manual labeling
Test dataset
Validation dataset
Automatic labeling
Training dataset
Train & evaluate the model
Sentiment classification using Naïve Bayes
Improving the dataset
External dataset collection
Correlating sentiment & external data

Fig. 1. Methodology Flow Chart

A representation of the methodology taken in this report can be seen in Fig. 1. It shows the different steps of data cleaning, filtering and preprocessing to train a model for sentiment analysis and topic modelling. As a final target, the trained sentiment model is used to find correlations between tweet sentiment and de facto voting outcomes.

2.1 Tweet Dataframe Cleaning

To overcome computational difficulties due to the size of the supplied 2.33 GB JSON file, the original dataset was subset to a dataframe that contained only 11 relevant columns but all 657,307 rows. Prior to Natural Language Processing (NLP) it was necessary to perform basic cleaning of the extracted dataframe and remove irrelevant geographic data which would hinder the sentiment model building itself.

A. Identifying Tags to Keep Given the focus of the sentiment analysis model on the candidature of H. Clinton and D. Trump, we needed to associate each tweet with at least one candidate. The relevant hashtags, words and account names related to these candidates were isolated. Some user tweets were a reply to an earlier tweet and missing the identified tags. Besides complicating the identification of the relevant candidate, the sentiment would depend on the original tweet. For example if an original tweet has a negative sentiment toward Trump and another user writes a positive retweet in agreement, it should be classified as negative. Owing to the difficulty in analysing the whole sentiment chain and accurately identifying candidates, any tweets not containing these tags were removed, reducing the row count by 8.87 percent (58,325 rows).

B. Reducing Geographic Footprint As we were only interested in tweets originating from the U.S. all other countries were removed, accounting for 9.98 percent of the data (59,777 rows). The geographic place data was further segmented at the following grains: “city”, “admin”, “neighborhood”, “poi” and “country”, which provided different levels of detail of “place”. We removed all other “place types” reducing the row count by 26.56 percent (143,224 rows), and split “city” and “state” into two new columns.

C. Data Tidying and Text Cleaning Simple text cleaning techniques on the ‘text’ column were applied, including the removal of punctuation, removing plain numbers and eliminating single character strings. Additionally, 171 instances of duplicate rows were removed, and the ‘text’ field was converted to lowercase to avoid misclassifying the same words with disparate cases as different words. We further ensured that there are no NaN values in the ‘text’ and ‘state’ columns. Ahead of tweet text preprocessing, our dataframe has 395,810 rows from the original 657,307 (39.78 percent reduction in rows).

2.2 Tweet Text Preprocessing and normalisation

In this section, we discuss the NLP steps taken on the tweet texts, in order to prepare them for training the model. To find a tweet's sentiment and to obtain accurate sentiment classification, it was necessary to filter various artefacts and noise from the tweet text.

A. Tokenization Tokenization is the process of breaking up a text corpus into words which are called tokens. Strings were split into a list of tokens and a bag-of-words was constructed. Tweets frequently carry contextual information along with the tweet text by including a hyperlink. These artefacts cannot be accounted for by the model, as such hyperlinks were detected and removed, prior to tokenizing the tweet text.

B. Removing Stop Words Stop words are commonly used words that include parts-of-speech such as articles and prepositions that add little contextual meaning. These words can be safely removed from the bag-of-words without impacting the final sentiment analysis. The NLTK library has a dictionary of stop words which we used to match against. Given the gender specific subject matter of the sentiment analysis, we kept gender identifying stop words, estimating that these words might be valuable in training the classifier.

C. Stemming Stemming is a technique used to reduce inflectional and derivationally related forms of a word to a common base form. We used the NLTK library PorterStemmer to perform the stemming operation. The goal of this step is to reduce the total number of distinct terms in the bag-of-words which in turn reduces the processing time of the final output [10].

3 Analysis and Results

3.1 Topic Modelling using Latent Dirichlet Allocation

Latent Dirichlet Allocation (LDA) is a technique used in topic modelling of text, which helps to discover differences in trends across documents. In this project the Python, gensim library, optimized LDA model was used to perform topic modelling.

A. Corpus To train the LDA model preprocessed tweets were converted into a word corpus. In this stage words that occur only in a low or high number of tweets should be filtered. They cannot represent a topic, because words that possibly represent the topic need to occur frequently and preferably only in the specific topic. It is difficult to estimate the correct lower and upper bounds that should be filtered to guarantee the optimal result. In this report, words that occurred in fewer than 20 tweets, or in more than 60% of tweets were removed.

B. Hyper-parameters To calculate the LDA model, topic and word distribution hyper-parameters, called alpha and beta, can be tuned. These variables affect the topic and word posterior distribution. If the topic or word distribution is not known, smaller alpha and beta values will give prior distribution less weight. In existing literature commonly used values are alpha=1/T and beta=0.1 [9] [5], where T is number of topics. We also tested Python LDA alpha 'auto' option, which learns an asymmetric prior from the corpus, but it performed more poorly.

B. Coherence Score To evaluate model performance, model coherence was calculated, by how semantically similar each topic's top words are. A higher coherence score indicates a better model. It is uncertain how the gensim library coherence model would semantically analyse tags, which may result in a lower score.

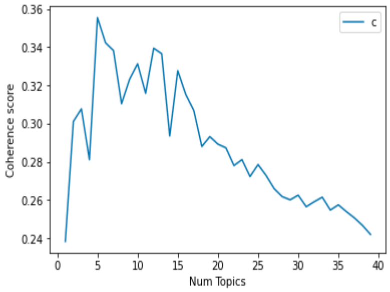
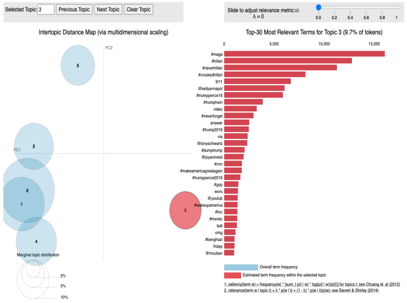


Fig. 2. LDA model coherence scores alpha=1/topics & beta=0.1



1. saliency(term w) = frequency(w) * [sum_t p(t | w) * log(p(t | w)/p(t))] for topics t; see Chuang et. al (2012)
2. relevance(term w | topic t) = λ * p(w | t) + (1 - λ) * p(w | t)/p(w); see Sievert & Shirley (2014)

Fig. 3. Example topic with alpha=1/6 beta=0.1 & 6 topics

A coherence score below 0.35 is considered poor. After assessing the topic words manually, it was difficult to label the groups. While with certain parameters the topics were slightly clearer, there was never more than one distinguishable topic.

3.2 Sentiment Analysis

A. Building and training the Classifier Owing to the fact that a single tweet can express different sentiment towards either candidate we use two Naïve Bayes classifiers, each trained on tweets containing a candidate's positive and negative sentiment. We then classify all tweets separately as positive, negative or neutral towards each candidate respectively.

B. Forming a training, validation and test dataset To obtain a sufficiently sized labeled dataset to build a feature list for the classifier, we used a set of hashtags strongly associated with a sentiment to label a subset of the tweets. Through Twitter's hashtag system, users are encouraged to self-label their tweets to categorise tweets of similar topics. It is assumed that using such a hashtag implies a strong orientation towards the respective sentiment.

The words from these tweets are then used as features for the classifier. Though this approach isn't as reliable, it wasn't feasible to manually label the required amount of data to train a model. To offset the bias created in this approach, the validation and test sets are strictly hand-labelled to catch any errors introduced. A total of 286 tweets were at random removed from the main dataset and labelled by all group members.

C. Applying the sentiment analysis Since the classifier is not trained on neutral sentiment cases the analysis is applied by only marking tweets with a very strong sentiment correlation as determined by the classifier. The probability of both positive and negative classification are subtracted to obtain a percentage of certainty. Only tweets above 99 percent classifier certainty are marked as belonging to either sentiment. If that condition is not fulfilled, the tweet is marked as neutral. With this approach a sentiment is found in around 64 percent of tweets.

This approach is chosen since an implementation of a Naïve Bayes Classifier trained on tweets with neutral sentiment showed poorer accuracy than the final model. (Trump accuracy: 0.41, Hillary accuracy: 0.59).

When comparing different cutoff probabilities, we notice that a higher percentage of required certainty yields higher accuracy overall with the classifiers used. Precision increases greatly at the cost of recall, which is caused by reducing type II errors in favor of type I error. An optimum is reached around 99 percent probability cutoff, as seen in Fig.4.

D. Evaluating the classifier When testing the classifier on the hand labeled data, the following measures of precision, recall, F-score and accuracy as seen in Table 1 were achieved. The relatively high measure of precision is explained by the use of a high probability cutoff of 99% when labeling neutral sentiment. This minimises the likelihood of type I errors as tweets without a very strong classification probability towards either positive or negative are marked as neutral. In the output of the sentiment analysis, neutral tweets derived from both models account for half of total tweets (52%). Positive tweets are slightly higher in Hillary data than that of Trump, at 27% and 23% respectively (Fig.5).

Table 1. Sentiment Analysis Performance

-	Recall	Precision	F-Score	Accuracy
Trump	0,54	0,70	0,61	0,5
Hillary	0,69	0,89	0,78	0,63

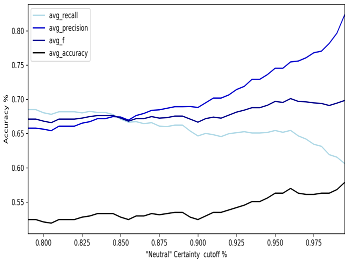


Fig. 4. Neutral labeling cutoff

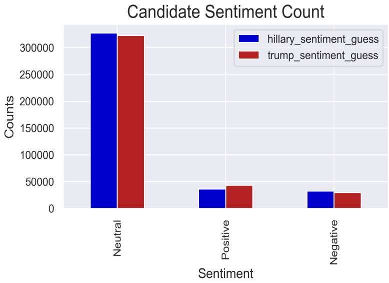


Fig. 5. Sentiment counts

The choropleth map depicting sentiment score gap in Fig.6 shows that candidate net positive score generally matches the geographic distribution of Republican and Democratic states, whereas the tweet sentiment over time chart gives a transparent overview of tweets per time, and which candidate was more widely supported, as seen in Fig.7.

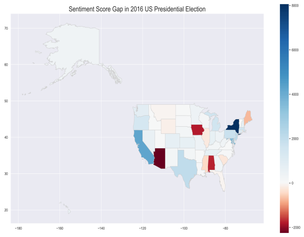


Fig. 6. Sentiment Score Gap

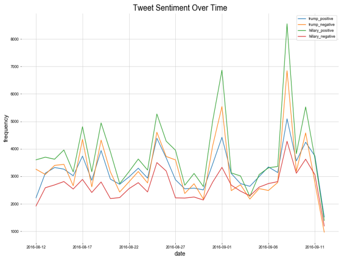


Fig. 7. Sentiments over Time

3.3 Correlation between sentiment and external dataset

A. Introducing external datasets To analyze the relationship between tweet sentiment and election outcomes or US demographics, datasets were imported from the United States Census Bureau and United States Election Project [8].

B. Correlation Analysis We examine the correlation between tweet sentiment and other factors, with the heatmaps in Fig. 8. In Trump-winning states, the more active users are, the more Trump-positive tweets they post. An increase in the total Trump sentiment score is associated with tweets per 10.000 people and tweets per user. Focusing on the earnings per family, within the states

Hillary won, higher median household income is associated with a tendency to tweet positively about her, with correlation being 0.64, while the same figures for Trump is -0.17. An increased voter turnout rate is associated with a higher Trump-positive ratio in Trump-winning states, with the correlation being 0.28. Although Clinton's supporters did tweet about her in a positive way, the correlation with voter turnout rate is marked as negative (-0.22).

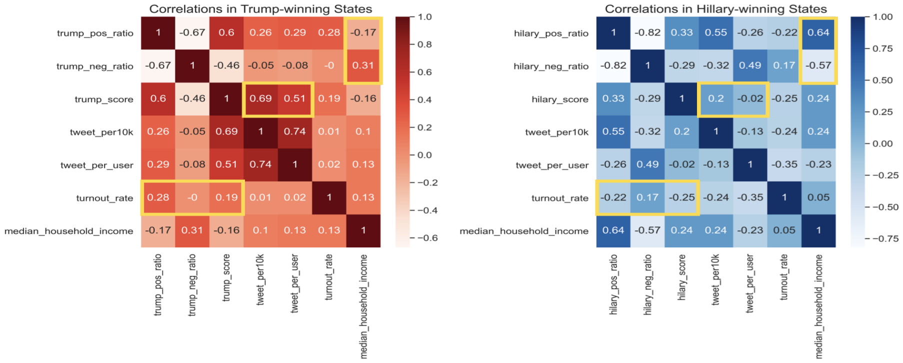


Fig. 8. Correlations in Trump and Hillary supporting states

4 Discussion and Conclusion

In summary, neither sentiment analysis, topic modelling nor correlation analysis yielded significant insights that contribute to the research question. The modest performance of models can be explained by the complexity of human language, which cannot be captured in an oversimplified model trained on a few hashtags. Even manual labeling proved to be challenging and prone to errors, misinterpretation or bias, illustrating that tweets are extremely difficult to understand even for humans due to their ambiguity. When it comes to drawing conclusions about actual voting results per location, tweets filtered for location do not serve as powerful predictors. Insightful correlations could also not be drawn from the topic modelling. In fact, the literature points out that Twitter is not accurate in representing all demographics of voters, and is subject to a self-selection bias in which users will select their own topics to tweet about. Even adjusting for tweets by unique users does not generate additional insights. [4]. Thus, further research should focus on more sophisticated techniques that account for complexity, to process the tweet text. For instance, network analysis can help to identify influential users, which can help to understand how positive and negative sentiments spread among groups [1].

References

1. Ausserhofer, J., Maireder, A.: National politics on twitter: Structures and topics of a networked public sphere. Information, communication & society **16**(3), 291–314 (2013)
2. Barnaghi, P., Ghaffari, P., Breslin, J.G.: Opinion mining and sentiment polarity on twitter and correlation between events and sentiment. In: 2016 IEEE second international conference on big data computing service and applications (BigDataService). pp. 52–57. IEEE (2016)
3. Bermingham, A., Smeaton, A.: On using twitter to monitor political sentiment and predict election results. In: Proceedings of the Workshop on Sentiment Analysis where AI meets Psychology (SAAIP 2011). pp. 2–10 (2011)
4. Gayo-Avello, D.: " i wanted to predict elections with twitter and all i got was this lousy paper"–a balanced survey on election prediction using twitter data. arXiv preprint arXiv:1204.6441 (2012)
5. Jónsson, E., Stolee, J.: An evaluation of topic modelling techniques for twitter (2015)
6. Kouloumpis, E., Wilson, T., Moore, J.: Twitter sentiment analysis: The good the bad and the omg! In: Fifth International AAAI conference on weblogs and social media. Citeseer (2011)
7. Martínez-Cámara, E., Martín-Valdivia, M.T., Urena-López, L.A., Montejo-Ráez, A.R.: Sentiment analysis in twitter. Natural Language Engineering **20**(1), 1–28 (2014)
8. McDonald, M.P.: Voter turnout data, http://www.electproject.org/home/voter-turnout/voter-turnout-data
9. S. Boussaadi, D.H.A..P.O.A.: Modeling of scientists profiles based on lda
10. Salunkhe, P., Deshmukh, S.: Twitter based election prediction and analysis. International Research Journal of Engineering and Technology (IRJET) **4**, 10 (2017)
11. Stieglitz, S., Brachten, F., Berthelé, D., Schlaus, M., Venetopoulou, C., Veutgen, D.: Do social bots (still) act different to humans?–comparing metrics of social bots with those of humans. In: International conference on social computing and social media. pp. 379–395. Springer (2017)
12. Tumasjan, A., Sprenger, T.O., Sandner, P.G., Welpe, I.M.: Predicting elections with twitter: What 140 characters reveal about political sentiment. In: Fourth international AAAI conference on weblogs and social media. Citeseer (2010)
13. Wang, H., Can, D., Kazemzadeh, A., Bar, F., Narayanan, S.: A system for real-time twitter sentiment analysis of 2012 us presidential election cycle. In: Proceedings of the ACL 2012 system demonstrations. pp. 115–120 (2012)