Sentiment Analysis and Political Party Classification in 2016 U.S. President Debates in Twitter

Tianyu Ding[1] and Junyi Deng[1] and Jingting Li[1] and Yu-Ru Lin[1]

[1] University of Pittsburgh, Pittsburgh PA 15260, USA

Abstract. We introduce a framework of combining tweet sentiment analysis with available default user profiles to classify political party of users who posted tweets in 2016 U.S. president debates. The main works focus on extracting event-related information in short event period instead of collecting tweets in a long-time period as most previous works do. Our framework is not limited in debate event, it can be used by researchers to build rationale of other events study. In sentiment analysis, we show that all three Naïve Bayes classifiers with different distributions obtain accuracy above 75% and the results reveal positive tweets most likely follow Gaussian or Multinomial distributions while negative tweets most likely follow Bernoulli distribution in our training data. We also show that under unbalanced sparse term document setting, instead of using "Add-1" parameter, tuning Laplace smoothing parameter to adjust the weights of new terms in a tweet can help improve the classifier's performance in targeted direction. Finally, we show sentiment might help classifying political party.

Keywords: President debate, twitter, sentiment analysis, event study, Naïve Bayes, political party classification

1 Introduction

Sentiment analysis is a useful and well-developed filed of research in text mining filed. In 2014, Medhat and others have categorized most existing sentiment analysis algorithms and applications into two parts: machine learning and lexicon-based approaches [1]. In term frequency study of 2016 U.S. president debates in Twitter, we discover two problems: 1) Individual words' distributions maybe different between positive and negative sentiment tweets; 2) Positive and negative sentiment tweets are usually unbalanced in real world. Based on these two problems, our study has two key contributions:

1. We use Gaussian, Multinomial and Bernoulli Naïve Bayes classifiers in sentiment analysis and observe the differences of words' distribution between positive and negative sentiment tweets. Based on the observation, we propose an idea on building a mixture word distribution in Naïve Bayes in future work.
2. We find instead of using "Add-1" Laplace smoothing parameter value to adjust the weight of new terms never appear under unbalanced settings, simply tuning it can

provide better weights to new terms and improve the performance of Naïve Bayes classifier in targeted direction.

2 Related Work

Sentiment analysis can be implemented by dictionary-based approach [3] which is the baseline in our sentiment analysis work or machine learning approach. The main problem of dictionary-based approach is that it usually performs poorly in short text such as tweets and fail to capture the semantics. To solve the problems in dictionary-based approach, machine learning classifiers are developed. Both supervised and unsupervised machine learning techniques have been studied for many years and achieve good results [4][9]. Pang and Lee have used several machine learning methods in sentiment analysis of movie reviews and achieved high accuracy [4]. However, one problem of previous machine learning sentiment analysis is lack of considering an appropriate time interval of collecting training data and unbalanced data problem in events study.

Many previous Twitter sentiment analysis studies selected equal size of positive and negative tweets in a long-time interval [7]. This is inappropriate if we want to study sentiment in a special event in Twitter. Some previous events study in Twitter has proved this point: in 2010, Diakopoulos [10] has studied the sentiment trend in 2008 president debate with collecting tweets just posted in the debates period; in 2012, Wong's study on quantifying political leaning from tweets, retweets and retweeters of 2012 president election indicated bigger data often means noisier and sparser [6]. To study tweets' sentiment in a certain event, the first step should collect data during the event period and utilize term frequency to check whether the collected data is representative of this event.

Next, unlike many previous studies applied several machine learning algorithms, our study focus on Naïve Bayes classifier. Naïve Bayes is one of the oldest, straightforward and easily-implemented technique in text analysis [8] and one advantage of it we utilize is that we can set different distributions in Naïve Bayes classifier to calculate likelihoods of existing data.

Another common problem in text analysis is giving appropriate weights of new terms never appear in existing training data. Many previous studies use "Add-1" Laplace smooth parameter in Naïve Bayes classifiers which is inappropriate in Twitter event study because the data is sparse and unbalanced then "Add-1" might overweight new terms under this setting. General traditional smoothing methods are summarized in [11] and in [2] a shallow semantic smoothing method has been proposed. However, we observe that by simply tuning the Laplace smooth parameter we may obtain better sentiment classification results and explanations of the results matching real world.

3 Data and Methods

Our raw data includes 12168 tweets posted in debate 1, 11204 in debate 2 and 5354 in debate 3. Selected features from this data set are: *tweet* i.e. text posted by user; *favor-*

ite_count i.e. number of favorites generated; *follower_count* i.e. number of followers; *friends_count* i.e. number of friends; *state_cd* i.e. abbreviate of living state. There are 3 user-defined features: *candidate_label* i.e. indicator of whether this tweet mentions Hillary Clinton or Donald Trump's names; *sentiment_label* i.e. indicator of whether the tweet is positive or negative sentiment; *topic* i.e. which LDA generated topic the tweet belongs to.

There are two main problems and one add-on problem in sentiment analysis we study: 1) Is the word distribution different between positive and negative sentiment tweets posted on 2016 U.S. president debates? 2) What is the overall sentiment of this president election and can we tune the Laplace smoothing parameter in Naïve Bayes classifier to find an appropriate value to weight the new words never appear if the dataset is unbalanced? 3) The add-on problem is that can we utilize the sentiment information in tweets to classify users' political party?

For the first problem, we compare Gaussian, Multinomial and Bernoulli Naïve Bayes classifiers with a lexicon based method. For the second problem, we tune the Laplace smoothing parameter in Gaussian Naïve Bayes. For the third problem, we add sentiment related features to a baseline model and use SVM, decision tree and logistic regression classifiers.

3.1 Sentiment Analysis.

Our sentiment analysis with Naïve Bayes classifier focus on these two main aspects:

- **Performance under different term distribution Naïve Bayes classifiers.**
  Baseline of Sentiment Analysis: Classify text sentiment based on score calculated from lexicon created by Hu and Liu [7].
  Gaussian and Bernoulli Naïve Bayes classifier: Tweets are text documents limited in 140 characters so most words will only appear once in a tweet. Bernoulli Naïve Bayes might be most appropriate, but we also use Gaussian and Multinomial Naïve Bayes to verify our guess of different term distributions under different sentiment circumstances.
- **Performance under different Laplace smoothing parameter settings.**
  As described in previous section, our idea is very intuitive: we tune Laplace smoothing parament from close 0 to 2 with smaller gap 0.25 each time and observe the trend of performance of Gaussian Naïve Bayes classifier in sentiment analysis.

3.2 Party Classification

We study party classification on Democratic and Republican based on three methods mentioned above.

Baseline of Party Classification: Favorite_count, follower_count and friends_count because they reflect activity of Twitter users.

Full model: Includes candidate label and sentiment label besides the baseline model because it reflects tweets sentiment towards mentioned candidate.

4 Evaluation

4.1 Evaluation Set-up

First, we have studied the terms' frequency and proportion in 3 debates respectively to ensure our selected tweets are matching real world. Here are our filters of feature selection of terms: 1) Remove all targets which are terms following by "@" symbol; 2) Remove hashtags because even some of them might be useful in categorizing tweets sentiment but most of them are like "#debatenight" which note what event is going on; 3) we include more stopwords such as name of two president candidates because no matter in which debates, their names will show very frequently.

The result is excited, we select top 10 most frequent meaningful terms in 3 debates and we find some high frequency terms: "tax", "black" are common topics, so their average counts decreasing by time and other hot topics such as "email", "isi" ("isi" is stem of "ISIS") are increasing from debate 1 to debate 2 which we all know the "email scandal" happened in debate 2.

Second, for sentiment analysis, we manually label 187 tweets include 70 positive and 117 negative labeled tweets from our raw tweets based on following filters:1) Tweets mention exact one of these two candidates' name because without mentioning the candidates' names we don't know whether this tweet is related to the president election or not; 2) Retweets are removed because it will inappropriately increase the weights of terms of the retweets; 3) we neglect sarcasms because it is hard to detect sarcasm both technically and manually, but it could be discussed in future work. Next, we apply the same cleaning rules in term frequency study on our labeled tweets to prepare a tweet corpus for following sentiment analysis.

4.2 Sentiment Analysis and Party Classification Results

Here, we use unigram feature to build the models. We let "Positive Sentiment" be 1 and "Negative Sentiment" be 0 in outcome.

Sentiment classification: we use 10-fold cross validation to relieve bias problem in small sample size setting. The results are summarized in Table 1. All 3 Naïve Bayes classifiers increase the accuracy from *59.36%* in baseline to above *75%*. The exciting result here is the decreasing trend of precision and increasing trend of recall in Table 1 which indicates Gaussian NB detects most true positive sentiment tweets while Bernoulli NB detects most true negative sentiment tweets and Multinomial NB is the medium. This result verifies our guess of word distribution is different between positive and negative sentiment. More evidences are found in our term frequency study of our labeled tweet: the most frequently used words in positive sentiment tweets such as "love", "yes", "proud" are almost twice frequent than words used in negative sentiment tweets such as "lie", "can't", "idiot". This result indicates main words used in positive sentiment tweets are highly repeated than words used in positive sentiment tweets which implied they might follow a Gaussian or Multinomial rather than a Bernoulli distribution, and opposite conclusion in negative sentiment tweets.

Table 1. Sentiment Classifier Performance

Classifier	Accuracy	Precision	Recall	F-score
Baseline	59.36	84.62	50.77	63.46
Gaussian Naïve Bayes	75.40	62.67	72.30	67.14
Multinomial Naïve Bayes	77.54	67.69	67.69	67.69
Bernoulli Naïve Bayes	75.94	70.83	52.31	60.18

Laplace smoothing parameter: Begin from a very small value 0.001, increased by 0.25 and end at 2, in Fig.1 we find "True Negative" (TN) is with increasing trend and "True Positive" (TP) with decreasing trend because unbalanced and sparse dataset has more trained negative terms than positive terms. If we give new terms a simply "Add-1" weight, it over-weights the new terms and lead to the increasing trend of major category and decreasing trend of minor category. Before using Naïve Bayes classifier, researchers can simply tune this parameter in training data and choose the most appropriate value for their studies.

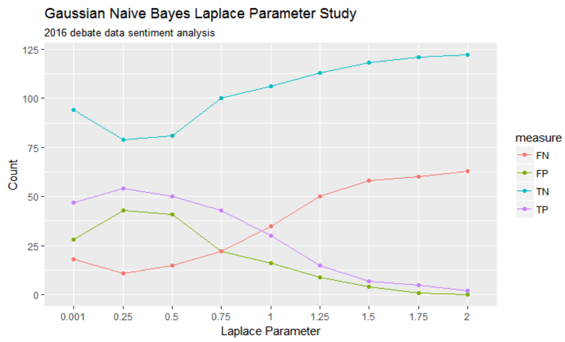

Fig. 1. Laplace Smoothing Parameter Analysis

Baseline vs Full model: This add-on study is built on 117 tweets with known sentiment label and political party as Democratic or Republican. The results are summarized in Table 2. We can find SVM on *Full model* over-perform other models which indicate sentiment information to candidates do help in party classification to some extent.

Table 2. Party Classifier 1 Performance

Classifier	Accuracy	Precision	Recall	F-score	AUC
SVM Base	72.97	54.55	37.50	44.44	64.83
SVM Full	81.98	67.65	71.88	69.70	72.90
Logistic Base	74.77	64.29	28.12	39.13	64.91
Logistic Full	67.57	41.67	31.25	35.71	70.02
Decision Tree Base	79.28	66.67	56.25	61.02	78.28
Decision Tree Full	71.17	50.00	50.00	50.00	66.87

5 Conclusions and Future Works

We show words might have different distributions under different sentiments. A well-selected Laplace smoothing parameter can help improving accuracy. Sentiment label might help improving user political party classification. There are several future works we can do: 1) Implementing a mixture distribution using EM algorithm in Naïve Bayes classifier to improve performance in polarized sentiment analysis, similar as implementing a Gaussian mixture model; 2) Utilizing emoticons to label tweets sentiment can save many human work [7] but might have lower accuracy;3) Using topics generated by Latent Dirichlet Allocation (LDA) and sentiment related to those topics might be more accurate than using *candidate_label* in our party classification model.

References

1. Medhat W, Hassan A, Korashy H. Sentiment analysis algorithms and applications: A survey. Ain Shams Engineering Journal. 2014 Dec 31;5(4):1093-113.
2. Saif, H., He, Y., & Alani, H. (2011). Semantic smoothing for twitter sentiment analysis.
3. Hu, M., & Liu, B. 2004. Mining and summarizing customer reviews. In Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining, 168-177.
4. B. Pang, L. Lee, and S. Vaithyanathan. Thumbs up? Sentiment classification using machine learning techniques. In Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 79–86, 2002.
5. Pennacchiotti, M., & Popescu, A. M. 2011. Democrats, republicans and starbucks afficionados: user classification in twitter. In Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining, 430-438.
6. Wong, F. M. F., Tan, C. W., Sen, S., & Chiang, M. 2016. Quantifying Political Leaning from Tweets, Retweets, and Retweeters. IEEE Transactions on Knowledge and Data Engineering, 28(8), 2158-2172.
7. Go, A., Bhayani, R., & Huang, L. 2009. Twitter sentiment classification using distant supervision. CS224N Project Report, Stanford, 1(12).
8. Schneider KM. A comparison of event models for Naive Bayes anti-spam e-mail filtering. InProceedings of the tenth conference on European chapter of the Association for Computational Linguistics-Volume 1 2003 Apr 12 (pp. 307-314). Association for Computational Linguistics.
9. Fernández-Gavilanes M, Álvarez-López T, Juncal-Martínez J, Costa-Montenegro E, González-Castaño FJ. Unsupervised method for sentiment analysis in online texts. Expert Systems with Applications. 2016 Oct 1;58:57-75.
10. Diakopoulos NA, Shamma DA. Characterizing debate performance via aggregated twitter sentiment. InProceedings of the SIGCHI Conference on Human Factors in Computing Systems 2010 Apr 10 (pp. 1195-1198). ACM.
11. Kaur G, Oberai EN. A REVIEW ARTICLE ON NAIVE BAYES CLASSIFIER WITH VARIOUS SMOOTHING TECHNIQUES.
12. Kumar A, Sebastian TM. Sentiment analysis on twitter. IJCSI International Journal of Computer Science Issues. 2012 Jul;9(3):372-8.