Sentiment Analysis of Twitter Data

Student: Alexander Muyshondt
Texas State University
601 University Drive, San Marcos, TX
adm1@txstate.edu

Professor: Byron Gao
Texas State University
601 University Drive, San Marcos, TX
bgao@txstate.edu

Abstract

Twitter is a global micro-blogging internet service on which users post and interact with messages known as "tweets". An important characteristic of Twitter is its efficiency in allowing the dissemination of information to a broad audience in a short matter of time. The purpose of this project is to analyze these tweets to determine whether the sentiment of the language contains hate speech or not. This project will employ natural language processing, text analysis and computational linguistic methods as the main techniques for classification. Future applications of such analysis may allow users to monitor large customer markets with an active online user base to determine customer satisfaction, consumer trends and underlying social models.

1. Introduction

From its beginning in 2007, Twitter has enjoyed perpetual growth and the cultural limelight largely in part to its user-friendly interfaces and easily consumable content. Due to character constraints, users are limited to typing short and to the point messages known as "Tweets". Because of its massive popularity, the social media platform has become a staple of cultural relevance for celebrities, companies and politicians; allowing these public figures to interact with users on a mass scale. With so many users and an appealing user experience, Twitter user interactions have skyrocketed. As of March 2011, as many as 140 million tweets are posted each day. [1]

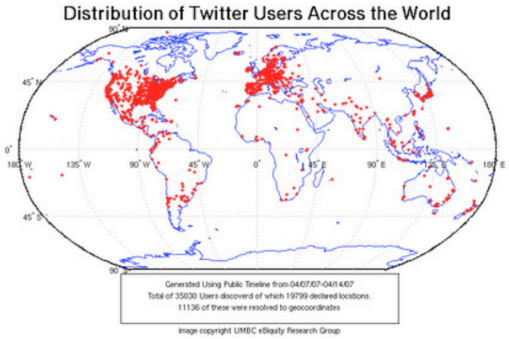

Fig. 1: A map of Twitter users worldwide (obtained from UMBC eBiquity Research Group)

This amount of tweeting leads to terabytes of data that may be analyzed to determine underlying bits of insight which in turn may be tailored to suit a multitude of needs such as a company gauging the interest of a new product, a politician polling their constituents on issues, or a social media persona attempting to broaden their influence by understanding their audience's needs. For this project, the purpose will be to analyze user's tweets using various data mining techniques and logistic regression to determine whether the content contains hate speech.

To achieve this goal, several different methods were employed in the analyzing of tweets such as Natural Language Processing, data mining, machine learning and statistics. In this report, we will attempt to conduct sentiment analysis on "tweets" using a machine learning algorithm to classify whether the tweet contains hate speech. By training an algorithm to identify hate speech we can analyze many tweets to determine their sentiment based on the accompanying language.

1.1. Past Attempts

Natural Language Processing (NLP) has become a major hotbed of research in a variety of fields such as Data Mining, Machine Learning, Analytics, Marketing and Business. The history of natural language processing began in the 1950s, although work can be found from earlier periods. In 1950, Alan Turing published an article titled "Intelligence" which proposed what is now called the Turing test as a criterion of intelligence. [2] Up to the 1980s, most natural language processing systems were based on complex sets of hand-written rules. Starting in the late 1980s, however, there was a revolution in natural language processing with the introduction of machine learning algorithms for language processing. This was due to both the steady increase in computational power and the gradual lessening of the dominance of Chomskyan theories of linguistics (e.g. transformational grammar), whose theoretical underpinnings discouraged the sort of corpus linguistics that underlies the machine-learning approach to language processing. [3] In the 2010s, representation learning and deep neural network-style machine learning methods became prevalent in natural language processing, due in part to an outbreak of results showing that such techniques can achieve state-of-the-art results in many natural language tasks, for example in language modeling, parsing, and many others.

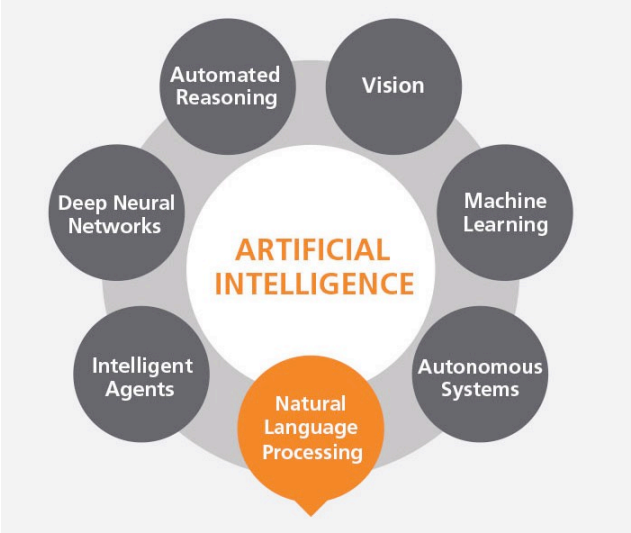

Fig. 2: A representative map of Natural Language Processing and its associated fields (obtained from Optum Inc.)

These days many companies and users have used Twitter analytic tools to track tweets about their brand or competition, to engage with leads or clients, and track the success of their campaigns. Their Twitter data is usually coupled with one or more of their other social media presences such as YouTube, Facebook, MySpace, or Snapchat; giving a larger picture focus on their level of influence. Twitter analysis has been employed by data scientists before such as IEEE Tweet Analysis for Real-Time Event Detection and Earthquake Reporting System Development. [4] Researchers could identify and localize natural disasters by recognizing increased usage of keywords around geo-tagged tweets. Twitter analysis has also been used for adaptive online language models, learning through tweets and user interactions.

1.2. Project Differentiation

This project will focus solely on sentiment analysis in regards to available Twitter data. By focusing strictly on Tweets, this approach will allow a narrow scope of inspection for better classification of tweets and their underlying intent. Data collection will also not consider any external factors such as age of account, hashtag usage, emoji's, or follower count to be as unbiased as possible and to preserve the integrity of the data collection process. By avoiding any sort of preference, we seek to eliminate any bias and maintain the integrity of the data itself.

2. Problem Description

The objective of this task is to determine whether a tweet contains hate speech based on its content. To simplify the matter, we say a tweet contains hate speech if it has a sexist or racist sentiment associated with it. So, the task is to classify racist or sexist tweets from other tweets.

Formally, given a training sample of tweets and labels, where label '1' denotes the tweet is racist/sexist and label '0' denotes the tweet is not racist/sexist, our objective is to predict the labels on the test dataset.

3. Data Description

A Kaggle dataset containing tweets split in the ratio of 65:35 into training and testing data was used for this project. Out of the testing data, 30% is public and the rest is private.

3.1. Data Files

1. **train.csv** - For training the models, a labelled dataset of 31,962 tweets was provided. The dataset is provided in the form of a csv file with each line storing a tweet id, its label and the tweet. There is 1 test file (public).

2. **test_tweets.csv** - The test data file contains only tweet ids and the tweet text with each tweet in a new line.

The data given is in the form of a comma-separated values files with tweets and their corresponding sentiments. The training dataset is a csv file of type tweet_id, sentiment, tweet where tweet_id is a unique integer identifying the tweet, sentiment is either 1 (racist/sexist) or 0 (not racist/sexist), and tweet is the tweet enclosed in quotations (“”). The test dataset is also formatted in a csv file with tweet_id and tweet. The dataset is a mixture of words, emoticons, symbols, URLs and references to people. Words and emoticons contribute to predicting the sentiment, but URLs and references to specific people do not and are thus ignored for this project. The mixture of words may also be misspelled, contain extra punctuations, and words that have repeated letters. To eliminate some of the excess noise from the dataset, the tweets must be preprocessed to normalize the dataset.

3.2. Evaluation Metric

F1-Score was the metric used for evaluating the performance of our classification model. In statistical analysis of binary classification, the F_1 score is a measure of a test's accuracy. It considers both the precision p and the recall r of the test to compute the score: p being the number of correct positive results divided by the number of all positive results returned by the classifier, and r is the number of correct positive results divided by the number of all relevant samples (all samples that should have been identified as positive). The F_1 score is the harmonic average of the precision and recall, where an F_1 score reaches its best value at 1 (perfect precision and recall) and worst at 0. [5]

The metric can be understood as -

True Positives (TP) - These are the correctly predicted positive values which means that the value of actual class is ‘1’ and the value of predicted class is also ‘1’.

True Negatives (TN) - These are the correctly predicted negative values which means that the value of actual class is ‘0’ and value of predicted class is also ‘0’.

False Positives (FP) – When actual class is ‘0’ and predicted class is ‘1’.

False Negatives (FN) – When actual class is ‘1’ but predicted class in ‘0’.

Precision = TP/TP+FP

Recall = TP/TP+FN

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

F1 is typically more useful than accuracy, especially if for an uneven class distribution.

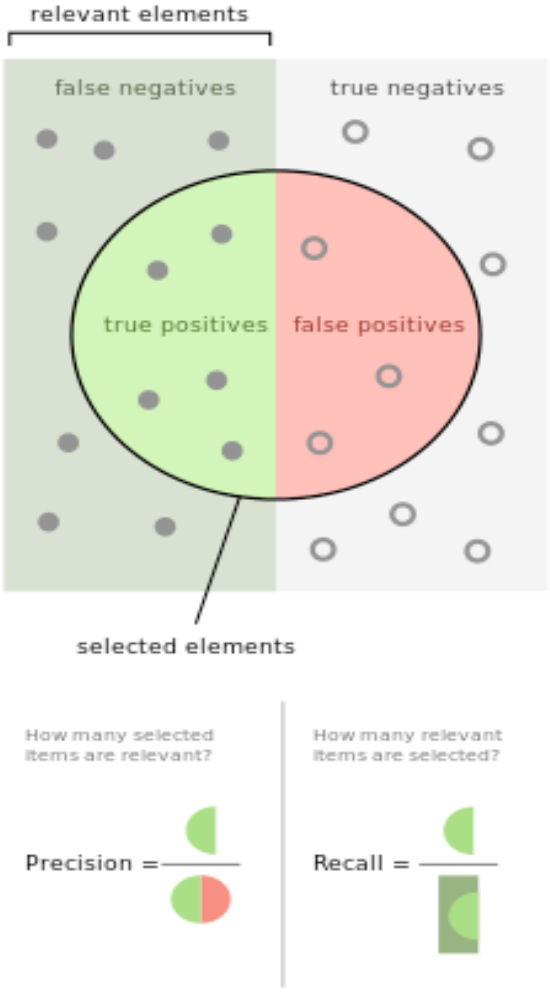

Fig. 3: A representative map of the components precision and recall that make up an F1 Score (obtained from Wikimedia Commons).

3.3. Tweets Preprocessing and Cleaning

Preprocessing of the text data is an essential step to make the raw text ready for data mining, i.e., it becomes easier to extract relevant information from the text and apply machine learning algorithms to. Skipping over this step leads to a high risk of working with inconsistent or noisy data. The objective of this step is to clean noise such as punctuation, special characters, numbers, and any terms that do not contribute to the sentiment of the tweet itself. These terms and characters do not contribute to the sentiment of the tweet and therefore can be safely removed for analysis.

In a later stage, we extract numeric features from our Twitter text data. This feature space is created using all the unique words present in the entire data. Therefore, by preprocessing the data well, we could achieve a better-quality feature space.

Data cleaning requirements that we put into effect are as follows:

- The removal of Twitter handles as these Twitter handles hardly give any information about the sentiment of the tweet.
- Removal of punctuations, numbers and special characters since they do not help in differentiating different kinds of tweets.
- Smaller words which do not add much value. In this case words of length three or less were arbitrarily chosen to be removed. For example, 'pdf', 'the, 'all'.
- Once the above three steps had been executed, we split every tweet into individual words or tokens which is an essential step in any Natural Language Processing task.
- Stemming was used to reduce variations of the same word down to its root. For example, the words 'loving', 'loved', 'loves' may all be reduced to 'love' without losing much context and keeping the intended sentiment of the Tweet.
- As mentioned above, tweets containing many twitter handles (@user) were removed.

A user-defined function was used to remove unwanted text patterns from the tweets. It takes two arguments, one being the original string of text and the other the pattern of text to be removed from the string. The function returns the same input string but without the given pattern. This function removed the pattern '@user' from all the tweets in our data.

A new column, tidy_tweet, was created to hold the cleaned and processed tweets. As discussed, punctuations, numbers and special characters do not help much and were removed from the text just as we removed the twitter handles. Here we replaced everything except characters and hashtags with spaces. To simplify the dataset further, all words having length 3 or less were removed. For example, terms like "hmm", "oh" are of very little use. It is better to get rid of them. Next, the processed tweets were then tokenized, the process of splitting a string of text into tokens. Tokens are individual terms or words. Stemming, a rule-based process of stripping the suffixes ("ing", "ly", "es", "s" etc) from a word, was then used to reduce words to their root form.

3.4. Story Generation and Visualization from Tweets

In this section we explore the processed tweets text in order to gain insight on the data. By evaluating the cleaned tweets we hope for more understanding of the data before moving on to work with any machine learning algorithms. We seek answers to such questions as: 'What are the most common words in the dataset?', 'What are the most common words in the dataset for negative and positive tweets?', 'Which, if any, trends are associated with this particular dataset?'.

To gain a broad understanding of the dataset and as to how the given sentiments are distributed across the training dataset we used a python library named wordcloud. A wordcloud is a visualization wherein the most frequent words appear in large size and the less frequent words appear in smaller sizes.

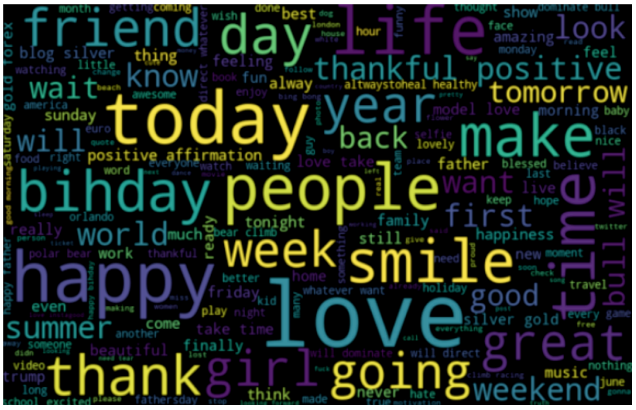

Fig. 4: A wordcloud containing the most common words of the *entire* dataset.

As shown in Fig. 4, most of the words in the dataset are positive or neutral, with *happy* and *love* being the most frequent ones. It doesn't give us any idea about the words associated with tweets containing hate speech. Hence, we will plot separate wordclouds for both the classes in our training data.

Fig. 5: A wordcloud containing the most *positive* words of the dataset.

Fig. 5 shows most of the words are positive or neutral. With *happy, smile,* and *love* being the most frequent ones. Consequently, most of the frequent words are compatible with the sentiment which is non-racist/sexists tweets. Similarly, we will plot the word cloud for the other sentiment, expecting to see negative, racist, and sexist terms.

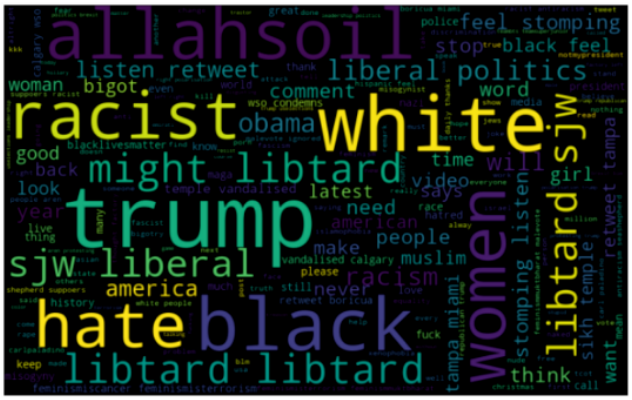

Fig. 6: A wordcloud containing the most *negative* words of the dataset.

As shown in Fig. 6, most of the words containing hate speech have negative connotations. So, it seems we have a pretty good text data to work on.

Hashtags in twitter are synonymous with the ongoing trends on twitter at any point in time. By checking whether these hashtags add any value to our sentiment analysis task, i.e., they help in distinguishing tweets into the different sentiments.

All the trend terms were stored in two separate lists — one for non-racist/sexist tweets and the other for racist/sexist tweets.

Once the two lists were created we could display the top ten hashtags associated with non-racist/sexist tweets and another graph for racist/sexist tweets as shown in Fig. 7 and Fig.8, respectively.

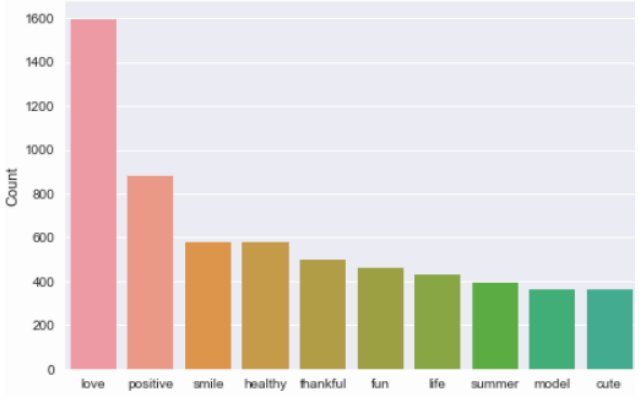

Fig. 7: Graph displaying the top 10 non-racist/sexist hashtags.

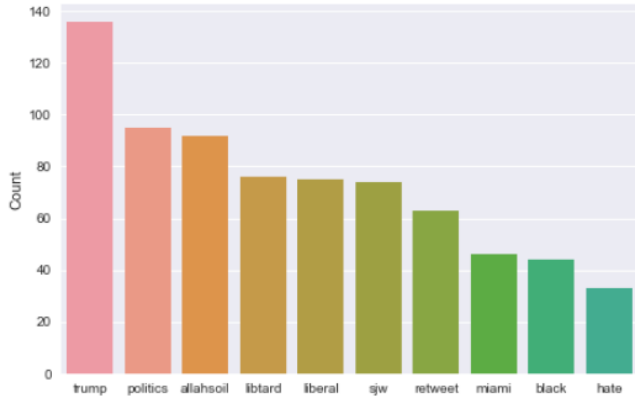

Fig. 8: Graph displaying the top 10 racist/sexist hashtags.

As expected, these hashtags contain useful information denoting the underlying sentiment of tweets and may be left in for analysis.

3.5. Extracting Features from Cleaned Tweets

To analyze the preprocessed data, it needs to be converted into features. Two data mining techniques: Bag-of-Words and TF-IDF, were used to construct features from the dataset using the text from tweets.

Bag-of-Words is a method to represent text into numerical features. Consider a corpus (a collection of texts) called C of D documents {d1, d2…..dD} and N unique tokens extracted out of the corpus C. The N tokens (words) will form a list, and the size of the bag-of-words matrix M will be given by D X N. Each row in the matrix M contains the frequency of tokens in document D(i). [6] Now the columns in the above matrix can be used as features to build a classification model. Bag-of-Words features can be easily created using sklearn's *CountVectorizer* function. We will set the parameter max_features = 1000 to select only top 1000 terms ordered by term frequency across the corpus.

TF-IDF is another method which is based on the frequency method but it is different to the bag-of-words approach in the sense that it considers, not just the occurrence of a word in a single document (or tweet) but in the entire corpus. TF-IDF works by penalizing the common words by assigning them lower weights while giving importance to words which are rare in the entire corpus but appear in good numbers in few documents. [7]

Let's have a look at the important terms related to TF-IDF:

- TF = (Number of times term t appears in a document)/(Number of terms in the document)
- IDF = log(N/n), where, N is the number of documents and n is the number of documents a term t has appeared in.
- TF-IDF = TF*IDF

3.6. Model Building: Sentiment Analysis

Once we have the data in the proper form and shape, we may began building predictive models on the dataset using the two feature sets mentioned earlier — Bag-of-Words and TF-IDF.

We used logistic regression to build the models as it predicts the probability of the occurrence of an event by fitting data to a logit function.

The following equation was used in our Logistic Regression:

$$\log\left(\frac{p}{1-p}\right) = \beta_0 + \beta(\text{Age})$$

Fig. 9: Logit equation for determining hate speech in tweets.

We trained the logistic regression model on the Bag-of-Words features and it gave us an F1-score of 0.53 for the validation set. We trained the logistic regression model on the TF-IDF features and it gave us an F1-score of 0.544. So, by using the TF-IDF features, the validation score has improved.

4. Conclusions

This project taught me how to approach a sentiment analysis problem using several different techniques related to data mining and machine learning. By combining the data mining techniques Bag-of-Words and TF-IDF with logistic regression, a meaningful association between tweets and its compositional language was ascertained. We started with preprocessing and exploration of data. Then we extracted features from the cleaned text using Bag-of-Words and TF-IDF. Finally, we were able to build a couple of models using both the feature sets to classify the tweets. The implications of the sentimental nature of tweets may be beneficial to a variety of interests.

4.1. Future Work

Future iterations of this project may employ several other techniques to utilize the full range of available data. The following are the top three considerations that may allow for the most insight to be gleaned from further attempts at this kind of Twitter Analysis.

- Emotional Ranges: A range of sentiments may be used to help improve the evaluation of this model. Tweets do not always have positive or negative sentiment. At times, they may have no sentiment at all i.e. neutral. Sentiment can also have gradations like the sentence, 'This is good.' is positive but the sentence, 'This is extraordinary.' is somewhat more positive than the first. We can therefore classify the sentiment in ranges, say from -2 to +2.

- Using symbols: During our pre-processing, we discard most of the symbols like commas, full-stops, and exclamation mark. These symbols may be helpful in assigning sentiment to a sentence.

- Emoji Usage: Emoji's have become vastly popular for expressing sentiment in regards to digital communications. Many people use them entirely in place of words to express their sentiment online. By not taking emoji's into account, our analysis loses a lot of possible data mining application. This may be remedied in future iterations where emoji's are considered.

- Combinatorial Data: Twitter is only one of a plethora of social media sites. Combining Twitter data with other online platform data may lead to a more defined characterization of texts. This may also be useful for building online profiles of specific ideas and how they change over time.

References

[1] Beaumont, Claudine (February 23, 2010). "Twitter Users Send 50 Million Tweets Per Day – Almost 600 Tweets Are Sent Every Second Through the Microblogging Site, According to Its Own Metrics". The Daily Telegraph. London.

[2] Anon (2017). *Turing, Alan Mathison*. *ukwhoswho.com*. Who's Who (online Oxford University Press ed.). A & C Black, an imprint of Bloomsbury Publishing plc. doi:10.1093/ww/9780199540884.013.U243891

[3] Chomsky, Noam. "The 'Chomskyan Era' (excerpted from The Architecture of Language)". Chomsky.info. Retrieved February 21, 2019.

[4] Elizabeth Williams, "GeoContext: Discovering geographical topics from social media", Advances in Social Networks Analysis and Mining (ASONAM) 2016 IEEE/ACM International Conference on, pp. 1342-1346, 2016.A. Alpher. Frobnication. Journal of Foo, 12(1):234–778, 2002.

[5] Powers, David M W (2011). "Evaluation: From Precision, Recall and F-Measure to ROC, Informedness, Markedness & Correlation" (PDF). *Journal of Machine Learning Technologies*. **2** (1): 37–63.

[6] Weinberger, K. Q.; Dasgupta A.; Langford J.; Smola A.; Attenberg, J. (2009). "Feature hashing for large scale multitask learning,". *Proceedings of the 26th Annual International Conference on Machine Learning*:

1113–1120. arXiv:0902.2206. Bibcode:2009arXiv0902.2206W.

[7] *Rajaraman, A.; Ullman, J.D. (2011). "Data Mining" (PDF). Mining of Massive Datasets. pp. 1–17. doi:10.1017/CBO9781139058452.002. ISBN 978-1-139-05845-2.*

[8] Monisha Kanakaraj, Ram Mohana Reddy Guddeti, "NLP based sentiment analysis on Twitter data using ensemble classifiers", Signal Processing Communication and Networking (ICSCN) 2015 3rd International Conference on, pp. 1-5, 2015.

[9] Joshi, P. (2018). Comprehensive Hands on Guide to Twitter Sentiment Analysis with dataset & code. Retrieved from https://www.analyticsvidhya.com/blog/2018/07/hands-on-sentiment-analysis-dataset-python/