# Twitter Sentiment Analysis via Bag of Words Model in Python

Israel Nolazco
Introduction to Machine Learning
*Lewis University*
Romeoville, Illinois
israelnolazco@lewisu.edu

*Abstract*— **Bag of Words is one of several Natural Learning Processing techniques used in Machine Learning that converts categorical data, such as text or words, into a numerical form. After such conversion, the goal is to count the frequency of each text or word and convert them into vectors. These vectors will then be used to make predictions. This document will make use of the bag of words technique, create a training model with dataset fill with tweets, and make a prediction whether a tweet is or is not racist/sexist. Additionally, this document will explore the full capabilities of the Bag of Words model and its shortcomings.**

**Keywords**

**Bag of Words, Machine Learning, Data Analytics, Sentiment Analysis, Twitter**

## I. Introduction

Twitter is a social network platform used by millions of users across the world. Most recently, Twitter has taken the overwhelming task to analyze tweets and determine whether those tweets violate their community guidelines[1]. Tweets that encourage hate, racism, sexism, terrorism are a few examples of the categories that violate the aforementioned guidelines. Utilizing a dataset gather from the website Kaggle, we will try to create a model that will predict emotions through text. This is where Natural Processing Language subfield Sentiment Analysis, comes in to play since it is the most common tool utilized to classified emotions. This document will show the creation of a sentiment model by using the Bag of Words technique (BOW). Primarily, this model looks to predict whether a tweet is racist/sexist. Finally, this document will make a case for the use of Bag of Words and analyze the benefits and limitations of this technique.

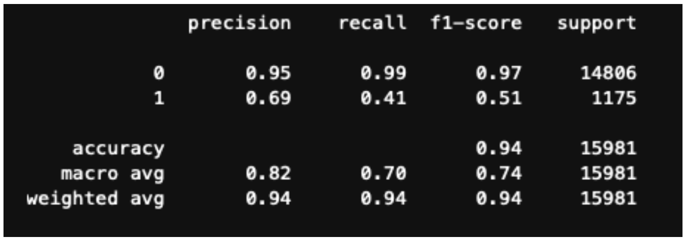
## II. Methods

The dataset gathered from Kaggle[2] was constructed two sets of data. One consisting of a training set, which was pre-labeled and contained about thirty thousand variables. The tweets deem racist/sexist was label as a "1" and for the rest deem NOT racist/sexist were label as "0." As for the testing set, the data consisted of unlabeled regular tweets. The testing set as the name suggests was meant to be used as testing point after creating the model. This model will be designed and tested with the programming language Python. This due to the size of the dataset, the overall flexibility, and resources available with this programming language.

The first step is to import our dataset into our Python environment, from there our first priority is to compress our data with only the most valuable pieces of information. Therefore, special characters such as, "", +, #, &, ( ) will be removed. Then, in order to maximize our accuracy, we need to simplify the grammar in our dataset. This means we are going to remove "-ed" and "-ing" from our tweets since they carry no statistical value in our model. Lastly, stop words such as, "to", "the", "at" will also be removed since they carry no statistical value and are only used for contextual purposes. Finally, we are going to randomize our tweet to ensure our model does not suffer from any predispose bias. Lastly, we created a matrix that will contain the tokenization of our words. From there, a model will be created and be used as a reference point in order to weight the probability of a tweet to be considered racist/sexist. together.

## III. Analysis

Given the privacy concerns from the users of Twitter, the actual usernames for each tweet were removed and replaced with the term "user." [3] Therefore, every single tweet had the word "user." It was then decided to add the term "user" to our list of stop words. As such when the function was applied to our dataset the term will be removed. When we create our bag of words, we invoke the function CountVectorizer. This function will transform our token into vectors. Those vectors will then be used in our sparse matrix with the shape of (31962, 41718). Additionally, we have counted a total number of 276004 non-zeros. This means we have gathered a total of 276,004 words deem not racist/sexist. Now that we have gathered our sparse matrix, it was time to split our training dataset into training/testing samples. Additionally, as previously stated our data was already randomized; we used the first 15981 tweets as the training sample and the last 15981 as the testing sample (half of 31962 is 15981). Then the function Multinomial() was invoked into our code. MultinomialNB() is a function of Naive Bayes classifier for multinomial models.  This model will provide us with a reference point when predicting the accuracy, precision, recall f1 score, and support of our testing sample.  As seen in Figure 1, our accuracy rate is that of 94% as for our precision it is higher in recognizing our non-racist/sexist content with 95% and only a 69% precision in recognizing our racist/sexist content.



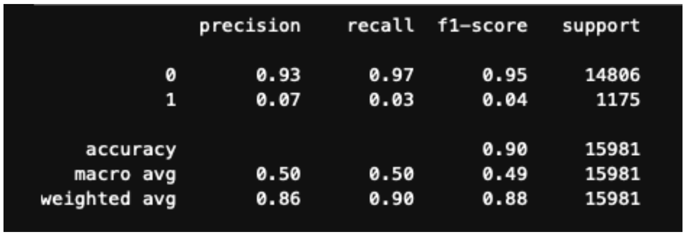
```
              precision    recall  f1-score   support

           0       0.95      0.99      0.97     14806
           1       0.69      0.41      0.51      1175

    accuracy                           0.94     15981
   macro avg       0.82      0.70      0.74     15981
weighted avg       0.94      0.94      0.94     15981
```

Figure 1. Statistical Analysis from the Training Dataset

Although, our accuracy rate is quite high; it is only logical to use our testing dataset and figure out how our model does.  In this case, the testing dataset was not pre label, and because our model had only included the first 15981 variables.  It was then decided to follow the same limitation with our testing dataset.  As seen in figure 2, our accuracy level had drop down to 90% and the precision score had dropped as well, and just like our training dataset the precision score to recognize racist/sexist tweets is lower with only 7%.  Therefore, the theory of this model being used as a general model cannot be guarantee given the precision score.



```
              precision    recall  f1-score   support

           0       0.93      0.97      0.95     14806
           1       0.07      0.03      0.04      1175

    accuracy                           0.90     15981
   macro avg       0.50      0.50      0.49     15981
weighted avg       0.86      0.90      0.88     15981
```

Figure 2. Statistical Analysis form the Testing Dataset

In light of this insight, a function was created from our model to predict a racist/sexist tweet.  We started out with some base text.  This base text had obvious negative and positive cognition as follows:

1. 'Horrible. Terrible. Dreadful. Awful. Pile of garbage. Junk.'
2. 'Fantastic. Amazing. Terrific. Classic. Best! Extraordinary. Authentic. Ideal. Vibrant. Powerful. Perfect. Imaginative. Incredible. Happy. Love. Pleasure.'
3. 'Okay. Great.'

The line of text 1 provided the following results [0.04 0.96]. The left value of 0.04 dictates a 4% probability the text is NOT racist/sexist and the right value shows a 96% probability of being racist/sexist; which is congruent with our expectations.  As for lines 2 and 3 we got scores of [1. 0.] and [0.983 0.017]. Thus far our model seems to catch positive and negative attitudes quite find.  A final test was then conducted with two more texts that contain racial slur and hateful messages.   The following quote was gathered from the news organization The Guardian and quotes the following line from Richard Spencer, a white supremacist[4] “Little fucking kikes. They get ruled by people like me. Little fucking octaroons. My ancestors fucking enslaved those little pieces of fucking shit.”  Unfortunately, this hateful message was given the following score. [1. 0.] Therefore, declaring this message not racist/sexist.

## IV. Conclusion

Bag of Words is an interesting technique to quantize emotion through text.  Although, our analysis shows a failure to register clear racist commentary; it showed the promising value in determining positive and negative terms.  Let us remind ourselves that Natural Language processing is a field of study that is quite young and there is certainly room for improvement.  Either the dataset must contain better variables that affect our model or a higher number of data entries.  After all our model is only as good as its dataset.  The Kaggle dataset was only updated two years ago and human expression changes every minute; with the addition that our dataset was only limited to thirty thousand variables.  Human language is far too complex to be determined with a simple digit.  Our speech narrative can include sarcasm, double meaning, idioms, etc.  The model created with the training data is not enough data to create a robust model.  As for Twitter and any social network platform, there is a clear benefit in implementing these types of models to enforce the community guidelines.  However, at this point, an auditor is clearly needed to make the final decision.

## References

[1] Help.twitter.com. 2020. The Twitter Rules. [online] Available at: <https://help.twitter.com/en/rules-and-policies/twitter-rules> [Accessed 29 August 2020].

[2] Toosi, A., 2020. Twitter Sentiment Analysis. [online] Kaggle.com. Available at: <https://www.kaggle.com/arkhoshghalb/twitter-sentiment-analysis-hatred-speech?select=train.csv> [Accessed 29 August 2020].

[3] Malik, U., 2020. Removing Stop Words From Strings In Python. [online] Stack Abuse. Available at: <https://stackabuse.com/removing-stop-words-from-strings-in-python/> [Accessed 29 August 2020].

[4] Wilson, J., 2020. White Supremacist Richard Spencer Makes Racist Slurs On Tape Leaked By Rival. [online] the Guardian. Available at: <https://www.theguardian.com/world/2019/nov/04/white-supremacist-richard-spencer-racist-slurs-tape-milo-yiannopoulos> [Accessed 29 August 2020].