Preliminary Analysis on Fake/Real News Dataset

By [Axel Nimbona](mailto:axel.nimbona@maine.edu) and Daniel Shipman

# Introduction

In this project, we will be working on creating a statistical analysis that differentiates fake news from real news by analyzing keywords and the amount of keywords per article. The population for this [[1]](#footnote-0)project is unique articles. The variable of interest is whether or not the article is real or fake.

The motivation for this research is that there has been a trend of false publications for ulterior purposes. An algorithm that could filter through false information would be incredibly beneficial. However that might be slightly outside of the scope of the problem. On the other hand, there might be noticeable trends between articles whose purpose is to educate versus articles that try to elicit an emotional response. By capitalizing on word choice and frequency, we hope to be able to differentiate between those two categories.

The consequences of this algorithm could vary. On one hand it could help reduce the spread of false information. On the other hand, the problem now shifts from trusting a false source to trusting a potentially biased algorithm. Either way, in case of success, it should help indicate what articles should be looked at more carefully.

# Background

Since the development of social media, the spread of information has grown at an exponential rate. However in many instances the information is not true, or the information serves an ulterior purpose. For example, governments use “the media to whip up patriotic fervour, boost support for the authorities, and marginalize and/or ridicule dissent and dissenters when these are not actively criminalized by the State” . This kind of spread of propaganda and other forms of biased information has a real effect on the population's views and actions. Therefore, there is a lot of current research being done into “fake news detection”.

The current state of fake news has been that detecting fake news has become a crucial problem attracting tremendous research effort. According to an article by a multinational research group “most existing methods of fake news detection are supervised, which require an extensive amount of time and labor to build a reliably annotated dataset.”[[2]](#footnote-1). However the article itself describes an unsupervised attempt.

# Methods

## Preprocessing

Since the focus of this paper is on word frequency, rather than coherent sentence structure, the original dataset needed to be altered. The original format contained the articles as plaintext and the population was an assortment of labeled articles; We opted to use unique words as predictors with word counts as data entries. We looked at the unique word count within every sample and saved them in a dataframe that contained all unique words across all samples. This simplified the problem from looking at english sentences and their complex structure to simply looking at how frequently specific words were being used within the article. However this method created three major problems. First of all, We had to parse all the words to remove differences between “this,”, “This” and “This.”. Secondly, duplicates existed in the data and the dataset was therefore not independent. Finally after compiling the data, we had over 60000 predictors and only 20000 samples. We solved the first problem by delimiting the string on spaces and then applying a regex of only capitals and lowercase letters. Afterwards, we transformed all characters to lowercase letters and counted the unique occurrences of each word. We opted to remove any numbers since we already had a massive amount of predictors and overfitting was a large concern. Afterwards we had to parse through the raw dataset to find and remove any duplicates. The high dimensionality of the dataset was dealt with by only considering data that was in the conjunction of the True and False datasets. Upon inspecting the “Fake” dataset, we noticed that there were many typos in the articles. By limiting the data to the conjunction, the amount of spelling errors was greatly reduced. Intuitively this makes sense, real news articles go through many more editors than fake articles and therefore inherently contain fewer errors. Additionally, to decrease the number of predictors further and to increase the uniformity of predictors we compared all predictors to a standard English dictionary. This helped completely remove typos and reduce the overall number of predictors further. The additional motivation for only incorporating real words is that the classifier might look at spelling mistakes as a clear sign of fake news. Finally, we ended up limiting the amount of samples used to 6000 since the size of the dataset became a computational challenge.

## Multinomial Naive Bayes Classifier

The first classifier we fit to the data was a Multinomial Naive Bayes Classifier. We opted for this model since its compulationally forgiving and commonly used in similar word count settings.[[3]](#footnote-2) The Classifier also works well with discrete value spaces.[[4]](#footnote-3) Meaning further regularization or scaling was not necessary. After fitting the model, we tuned the regularizing parameter alpha to minimize test error. We then computed a confusion matrix using the determined alpha.

## Random Forest Classifier

With sparse datasets, random tree classification tends to outperform Naive Bayes classifiers.[[5]](#footnote-4) After fitting the classifier we tried 10 different depths and recorded the optimal tree depth. We performed all calculations on a training/test split of 0.5. Meaning half the data was used for training and half the data was used for testing. Finally we computed a confusion matrix for evaluation.

# Results

## Preprocessing

The result of our preprocessing methods is a sparse dataset with 6000 samples and 11627 predictors. Each cell contains the frequency of a unique word specified by the column for each sample. All determining one dependent variable. The data set is about 99% 0’s and is therefore sparse. Containing 69762000 values and 733532 non-zero values. Our data allows for up to 30000 samples, however due to computational limitations, this is infeasible.

## Multinomial Naive Bayes Classifier

The following graph (Figure 1.1) determined the optimal alpha value for our data on a range of [0.05,1] with a step size of 0.01

*Error frequency versus alpha values*

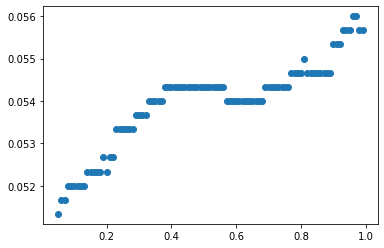


Fig. 1.1

Using the determined optimal alpha, our model scored 94.89% accurate on the testing data. With the following confusion matrix.

*Confusion Matrix for Multinomial Naive Bayes Classifier*

*0 represents false and 1 represents true*

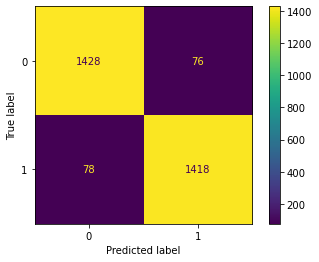


Fig. 1.2

Fig 1.2 was calculated with a test set of 3000 samples. Trained by 3000 samples.

## Random Forest Classifier

On a range from 1 to 10, the optimal depth was determined to be 8 (See Fig 2.1).

*Error Frequency vs Maximum depth*

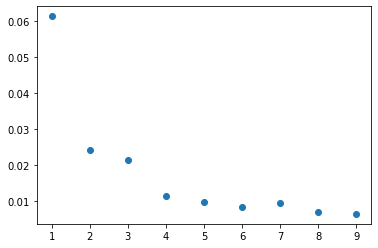


Fig. 2.1

With a max depth of 8, the random forest scored 99.4% accurate on a test set of 3000 entries. Trained by a sample of 3000 entries.

*Confusion Matrix for Random Forest Classifier*

*0 represents false and 1 represents true*

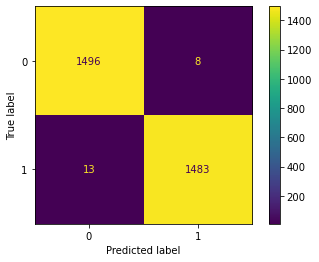


Fig. 2.2

# Conclusion

Comparing the two performances indicates that the Random Forest Classifier is the preferred method. However it should be noted that these models were produced using about 20% of the total possible sample size. When attempting with 20000 samples, we ran into computational problems with Random Forests. Nonetheless, the performance of both models was far greater than expected. Something as abstract as “Fake News” seems difficult to classify without properly researching the article. There is a strong possibility that the dataset contains key phrases that allow the classifier to perform so well on the test data.

1. Jones, M. (2019). The Gulf Information War| Propaganda, Fake News, and Fake Trends: The Weaponization of Twitter Bots in the Gulf Crisis. *International Journal Of Communication, 13*, 27. Retrieved from <https://ijoc.org/index.php/ijoc/article/view/8994/2604> [↑](#footnote-ref-0)
2. Yang, S., Shu, K., Wang, S., Gu, R., Wu, F., & Liu, H. (2019). Unsupervised Fake News Detection on Social Media: A Generative Approach. *Proceedings of the AAAI Conference on Artificial Intelligence*, *33*(01), 5644-5651. <https://doi.org/10.1609/aaai.v33i01.33015644> [↑](#footnote-ref-1)
3. https://en.wikipedia.org/wiki/Naive\_Bayes\_classifier [↑](#footnote-ref-2)
4. Ibid. [↑](#footnote-ref-3)
5. Caruana, R.; Niculescu-Mizil, A. (2006). *An empirical comparison of supervised learning algorithms*. Proc. 23rd International Conference on Machine Learning. [CiteSeerX](https://en.wikipedia.org/wiki/CiteSeerX_(identifier)) [10.1.1.122.5901](https://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.122.5901). [↑](#footnote-ref-4)