Fake News Detection Using Text Classification

Osama Shamout*

Lebanese American University
osama.shamout@lau.edu

Abstract

Opinion polarization and targeted propaganda generated from online fake news poses a major challenge for government institutions and private organizations. Such misinformation and disinformation can undermine government institutions and direct public opinion. On the macrolevel, this hinders policy makers and organizations from capturing real data to analyze and create recommendations, while on the microlevel, it can polarize and influence people's daily decisions. Further, with gigantic data being produced daily, it is almost impossible for human labor to detect and curb false information. Hence, the present study aims to develop a computerized model using Machine Learning (ML) methods, namely classification and text analysis from the field of Natural Language Processing (NLP) to predict online fake news. Two data files, fake and real news, are present and provided by Kaggle presenting our dataset. On this dataset, data exploration was performed to understand the available data. Preprocessing utilized stopwords, word tokenization, lemmatization, and part-of-speech tagging. Four different modeling classification techniques were used, Logistic Regression, Nayve-Bayes, Passive Aggressive, and kNN. Findings show that passive-aggressive model performs the best (99.45%) while kNN has performed the worst (87%), surprisingly even worse, when standardization is applied (57%). Further research shall explore the same methodology and steps on different datasets to test its applicability on different data as the current dataset might be heavily biased. Also, techniques involving deep learning could be applied to contrast results.

1. Introduction

Online fake news poses a threat to the current modern society. Social media has infiltrated the daily lives of billions of people across the world with about 4.7 billion active social media users sharing information across the world with the number projected to increase to 5.85 billion by 2027 [12]. Thus, to produce and spread false information has become easier than ever with the strides in current computer technologies and network capabilities. This false information disseminated via social media can generate social confusion and lead to an inadequate response during disasters and emergencies. Most recently, during COVID-19 pandemic [9], the raid of Capitol Hill in 2021 after Mr. Biden took presidency [5], or the cyber warfare in the Russian-Ukranian war aimed to influence the public opinion [10]. Hence, it is important for any platform that offers or shares information to be able to classify and flag false news to alarm users about the reliability of such spread information. However, given that we consume huge amounts of information daily, see figure 1, the need for automated models to detect fake news is pivotal. Hence, this paper aims to develop a Machine Learning (ML) model that addresses the problem of fake news. By using two datasets from Kaggle that have true and fake news, we will build a model that is able to classify news based on the given input text.

2. Related Work

Various literature is available on the topic of fake news as societies have begun to realize the devastating consequences of fake news. For

*Student at the Lebanese American University

starters, fake news detection that utilizes its article's author reputation, network metadata, image analysis, and text analysis can produce the most accurate results. However, this is rarely feasible. Hence, previous research has tried to tackle the issue of fake news through different ways. For example, one mechanism is to identify the news source or publisher to detect false information that aims to spread propaganda. This is clearly shown in Belcastro et al. paper that identifies "social bots" generating false information and polarize political opinion [3]. Their TIMBRE (Time-aware opInion Mining via Bot REmoval) methodology which optimizes on keyword-based classification has been applied on Tweets and proven to be able to detect the social bots tweets in political campaigns, as applied in the 2016 US presidential elections, and hence were able to filter the general consensus about the election and getting a more accurate result of the real opinions of voters. Another example by Ahmed et al., has applied ML classification, n-grams, and terms frequency metrics, achieving 98% accuracy in detection [1]. However, current issues with the model are that it labels Middle Eastern News as "Fake News", despite it being aggregated from Al-Jazeera, DW, Roya New English, which are trusted fact-checked local and international news channels across the MENA and Europe. Typically, the current model is giving correct answers with Linear Regression (LR) and Random Forest Classifier (RFC) techniques, meanwhile, it is failing with the Gradient Boosting Classifiers (GBC) and Decision Tree (DT) predictions. Finally, this paper will draw from Ahmed et al. methodology and focus on the detection of fake news with hopes of improving and adjusting the accuracy level, with the aim of correctly identifying false news, even in the Arab region and remediate the issues in the previously described Kaggle model.

3. Proposed Method

The methodology of the experiment involves the typical pipeline of NLP problems. A good technique for obtaining the relevant information of a text is to eliminate the elements that may be irrelevant and highlight more what the texts have in common than their differences. As such we will perform tokenization, stopwords, parts-of-speech (POS), and lemmatization.

3.1. Stop Words and Cleaning

Stop words are the words that are typically excluded from natural language processing. The text does not get much information from these terms, which are among the most common in any language (along with articles, prepositions, pronouns, conjunctions, etc.) alongside, we optimized it more by adding several terms that are usually missed (I'm, n't, 's, etc.) which were present in our text. In our dataset, we defined the stopwords using the wordcloud import and have amended our desired words. Further, we lower case the text to normalize it and prepare it for tokenization.

3.2. Tokenization

Tokenization is a technique used in natural language processing to break down phrases and paragraphs into simpler language-assignable elements. The collecting of data (a sentence) and its breakdown into comprehensible components are one of the first steps. They are particularly useful as they allow us to have visualization and analysis through the resulting text "tokens". In our case, it was particularly useful as it allowed us to perform part-of speech tagging and analysis.

3.3. Part-of-Speech Tagging

Part-of-speech tagging, also known as grammatical tagging in corpus linguistics, is the act of designating a word in a text (corpus) as belonging to a specific part of speech based on both its definition and its context. It is beneficial in trying to assign the "type" of speech a tokenized word is. For example, "Table" would be labeled with "N", meaning that it is a Noun. Taking into account the purpose of the words

can reveal important statistics about a texts' authenticity. For example, text that is filled with fill words and pronouns is more likely to be classified as fake (comparison of fake and real news based on morphological analysis).

3.4. Stemming

Even though we already have a small list of words, we can make it even smaller, we still need to make it smaller. This helps to reduce the dimensionality in the text and keeping only keywords that will assist us when modeling. Lemmatization is a key process in many practical NLP tasks that can be also called stemming. Though what makes lemmatization special is that it returns words to a meaningful root especially when given context, a stemmed word of. For instance, rock can either mean noun rock, or the verb rock. Thus, this is where our POS tagging comes in handful. By combining the POS tags and the lemmatizer we can get a clearer image about the text. However, this process has two costs. First, it is a process that consumes resources, especially time. Second, it is usually probabilistic, so in some cases we might get unexpected results. From the nltk library, we use the WordNetLemmatizer to stem the word and the wordnet to obtain a corpus. The corpus contains previously input texts and words by professionals to aid in the NLP process. By this stage, the words are ready for modeling.

3.5. WordClouds

WordClouds are used to visualize the frequency of the words in text. below are three visualization of the true dataset text, and fake dataset text. The results were not promising as they contained similar terms and very little differences, see figures 9 and 10

3.6. Feature Extraction

Feature extraction is a pivotal step in the NLP text analysis. Since we have words and sentences, the computer cannot recognize and manipulate such data. Hence, there needs to be a way to convert the text into a more readable format. Thus, a vectorization process occurs that transforms the text into numerical vectors.

3.7. Term Frequency-Inverse Document Frequency and Count Vectorize

The first step to vectorize is to use Count Vectorize. Count Vectorize creates a token count matrix of our text feature. This results in a sparse representation of the counts. Then, the counts are inputted in the Term Frequency-Inverse Document Frequency (TF-IDF). The TF-IDF is a statistical technique that assesses how pertinent a word is to a document within a collection of documents. The more a word is present in the text and across the whole document, the less score it has (hence the inverse). Thus, our models are now ready to receive our matrices.

3.8. Modeling

We will be using several models in this paper. But before we dive into that. We need to divide our model into split train test. We will set the training size to be = 0.80 and the test size = 0.20. Then, we perform the following models, Logistic Regression, Naïve-Bayes, and finally Passive Aggressive.

3.8.1 Logistic Regression

Logistic regression is statistical technique known to handle binary classification problems where the outcome can only have a dichotomous nature or have other categorical alternative values. It can be used, for instance, to determine the probability that an event will occur. The logistic regression model relies on the logit (log-odds) and sigmoid function. The log it uses the probability and produces a value between negative infinity to positive infinity while the sigmoid function translates this value into a probability between 0 and 1. This is useful in our case as we try to classify the texts into

either fake or real news. We use the sklearn library and obtain the LogisticRegression object to perform our modeling.

3.8.2 Naïve-Bayes

The probability of the classes can be derived from a specific sequence of various observations using the Naïve-Bayesian classifier. The model's underlying presumption is that given the class, the characteristic variables are conditionally independent. To use this model, import Naïve-Bayes from sklearn library.

3.8.3 Passive Aggressive

Famously known to be the most useful model in classifying fake and real news. The algorithm is simple and somewhat intuitive. While the model is classifying correctly, it does not do anything, if there is a mistake, it updates the weights assigned. Thus, it keeps adjusts with every new entry (if need be, an adjustment).

4. Experiments

4.1. Duplicates or Non-Duplicates, Raw or Preprocessed

This research involved an experiment to assess difference between duplicated values and non-duplicated over the processed and non-processed data. At every stage, we perform the identical steps during vectorization and modeling. In essence, the experiments are processed in two ways, the processed and the raw data. We assume the data is raw, however, indications during data exploration indicate otherwise (discussed in the Results). Then, we input the data into each model with duplicates or non-duplicates. This results in 12 different types of models. Figure 1 below shows how the experiments are made.

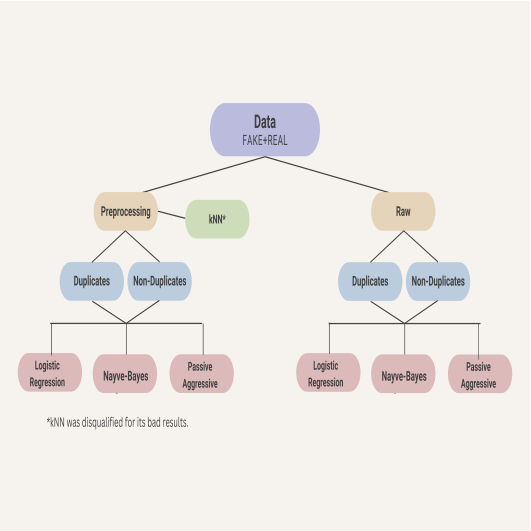

Figure 1: *Diagram depicting the process of our project*

4.2. Normalization

It is worth to mention that we applied normalization, the process of converting the available data into the range of 0 and 1 after receiving a low score for kNN. This was performed using the MaxAbsScaler (MaxMin does not work on sparse matrices).

4.3. Dataset

Data Exploration The current data is divided into two files the true news dataset "True.csv" containing 21417 values, and the fake news dataset "Fake.csv" containing 23481 values. Both datasets are provided from Kaggle. Both datasets contain four features of Object type (String) which are title, text, subject, and date. The date range for the Fake dataset is from 31 March 2015 to 19 February 2018 and for the True dataset from 13 January 2016 to 31 December 2018. Below is a basic summary of the two tables, refer to Table 1 for fake news, and Table 2 for true news. We append a label column to flag each row as either Fake = 1 or True = 0. This is necessary as we need to handle the categorical label to later merge the two datasets and perform functions and analysis.

	Fake Dataset
Number of Features	4
Number of Labels (appended)	1
Number of Values	23481
Date	31 March 2015 to 19 February 2018

Table 1: *Fake News Data Description*

	True Dataset
Number of Features	4
Number of Labels (appended)	1
Number of Values	21417
Date	13 January 2016 to 31 December 2018

Table 2: *True News Data Description*

No null values exist in both datasets. However, duplicates exist. See figures below depicting how duplicates increased the fake news count to be more than the real news. Also, in the Fake dataset there are more specific categories in defining the subject which are, News, Politics, Government News, Left-News, Middle-East US-News. Meanwhile the true dataset contains: Politics News and World News. This seems to be problematic comparisons might not be drawn if we want to weigh in the subject as a feature. Segmentation is also difficult as that would require human labor to read the true dataset to identify the text subject.
The distribution of true news is significantly higher in 2017. No true news data exist in years 2015 and 2018, giving all that year to be distributed with fake news (Bias originating from dataset). Also, fake news are significantly higher in year 2016. Refer to Figure 2 for months distribution, and Figure 3 for years distribution.

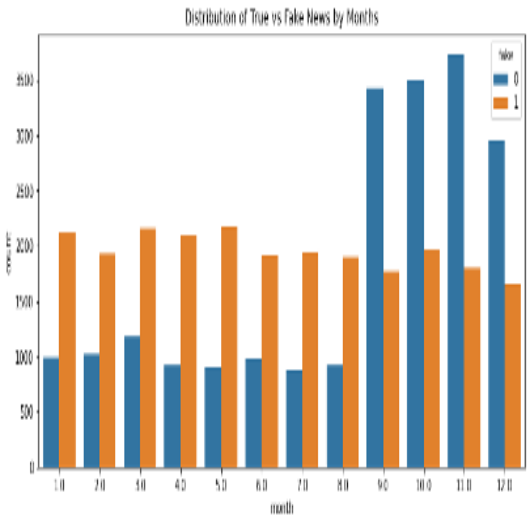

Figure 2: *Distribution of Fake and Real News by Months*

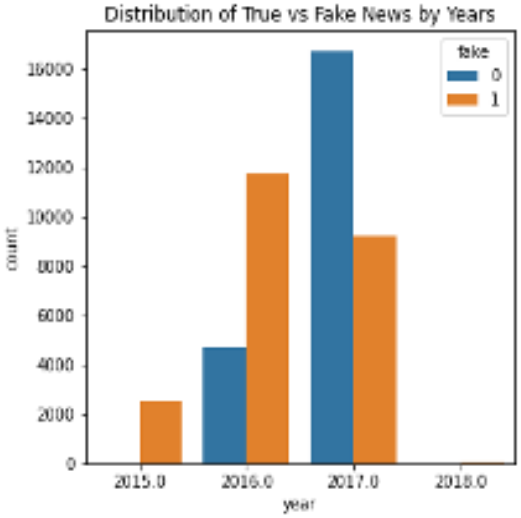

Figure 3: *Distribution of Fake and Real News by Years*

In an attempt to understand if word counts matter. We find that fake news is 48% higher on average in its title word count, also fake news are 9.7% on text word count. More interestingly, the unique average in and word average of title are similar for both fake and true news. Also, the unique text average is lower for true news than false news. However,

the ratio of average:unique is not as high as fake news is, refer to Table 3 below.

	Fake	True
Title Word Average	14.732805	9.954475
Title Unique Average	14.490609	9.876827
Text Average	423.197905	385.640099
Text Unique Average	240.901239	226.932764

Table 3: *Text Analysis Averages*

4.4. Software

The code was run on a MacOS device with Virtual Studio Code. Co-Lab was used to run the data exploration while VSC was later used for the heavier side of the computations. This report has been written in LaTeX on Overleaf.

5. Results and Discussion

5.1. Dataset Exploration

The results regarding the dataset raise questions about its authenticity. Through further exploration, we concluded with some insights. First, the fake news contained a lot more twitter usernames than the true news, see Figure 4 below.

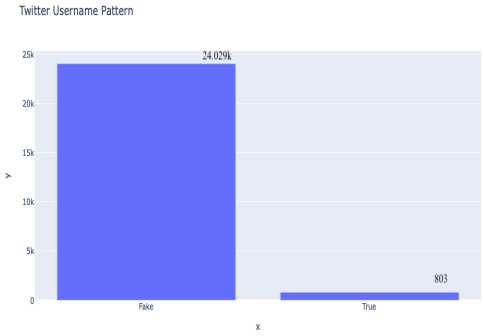

Figure 4: *Twitter Users in Fake vs True News dataset*

Second, it can be observed that the tokenized words in the real dataset make understandable splits, however, in the fake datasets, several instances can be spotted which have incompressible text and splitting positions. This lead to a higher uniqueness count despite its unusefulness. From this result, we hypothesize that the obtained dataset has been already preprocessed and published. Through further research, we found in the dataset providers files that it indeed has already been preprocessed. Third, the word frequency seems very similar for both data types, see Figures 7, and 8. It was apparent that the dataset has been influenced by the U.S., dataset especially that it has been collected during the elections period as previously mentioned. Yet, it does not seem likely that it would have such similarity between fake and real, indicating some bias in the data. Finally, the availability of duplicates in itself has such a low probability, especially that the news obtained are aggregated from a Political Fact checking platform. Hence, about 3000 duplicate entries is not a good indicator on the quality of this dataset alongside the previously mentioned issues. Refer to Figure 5 before removing the duplicates, and Figure 6 after removing the duplicates below.

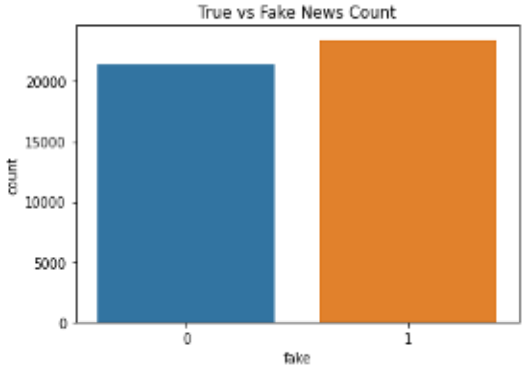

Figure 5: *Fake vs Real News Count before Removing Duplicates*

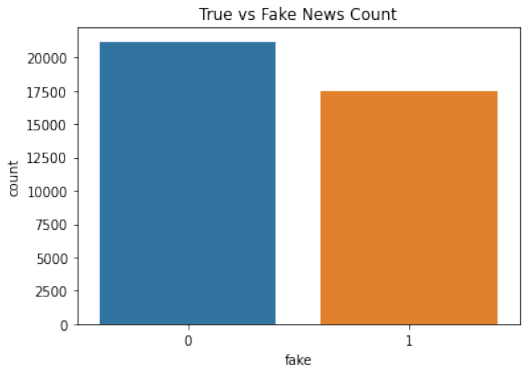

Figure 6: *Fake vs Real News Count after Removing Duplicates*

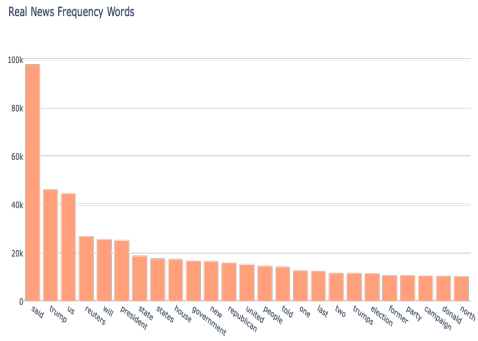

Figure 7: *Real News Word Frequency*

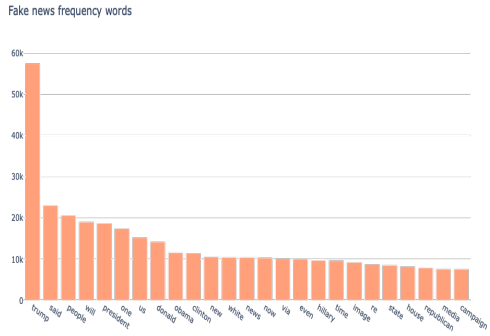

Figure 8: *Fake News Word Frequency*

The WordClouds did not reveal much information or difference between the datasets' frequent words, see Figure 9 and 10. Most results were as expected and depicted previously in the word frequency graph above revolving around "Donald Trump", "Elections", "White House", etc.

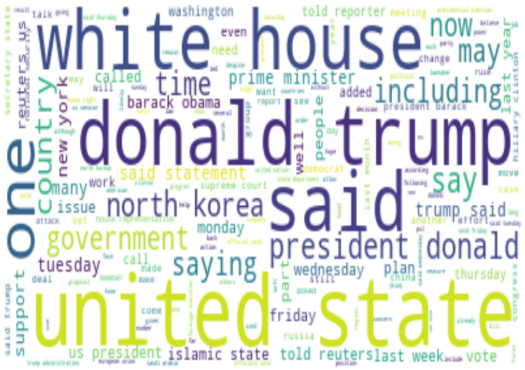

Figure 9: *Real News WordCloud*

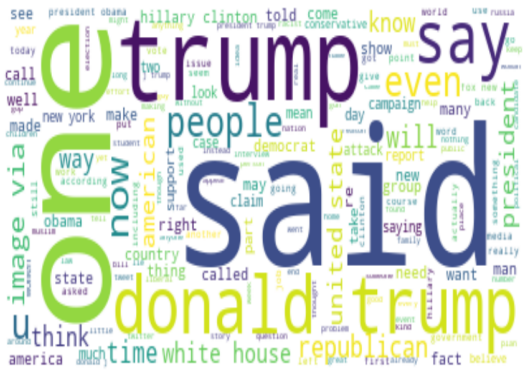

Figure 10: *Fake News WordCloud*

5.2. Modeling

Our model achieves a slightly higher percentage from 98% to 99.5% using the Passive Aggressive model. Below is a summary of the different scores obtained. For Logistic Regression, refer to Table 4, for Naive-Bayes, refer to Table 5, for Passive Aggressive, refer to Table 6, for kNN, refer to Figure 11, and Figure 12.

5.2.1 Logistic Regression

		Accuracy	Precision	Recall	F-1 Score	AUC
Processed	Duplicated	98.70	99	99	99	99
	Non-Duplicated	98.65	99	99	99	99
Not	Duplicated	98.81	99	99	99	99.8
Processed	Non-Duplicated	98.81	99	99	99	99.8

Table 4: *Logistic Regression Scores*

5.2.2 Naive-Bayes

		Accuracy	Precision	Recall	F-1 Score	AUC
Processed	Duplicated	94.10	94	94	94	98.42
	Non-Duplicated	92.17	93	92	92	99.00
Not	Duplicated	93.61	94	94	94	98.39
Processed	Non-Duplicated	93.61	94	94	94	98.39

Table 5: *Naive-Bayes Scores*

5.2.3 Passive Aggressive

		Accuracy	Precision	Recall	F-1 Score	AUC
Processed	Duplicated	99.51	1	1	1	99.96
	Non-Duplicated	99.39	99	99	99	99.94
Not	Duplicated	99.60	1	1	1	99.94
Processed	Non-Duplicated	99.61	1	1	1	99.93

Table 6: *Passive-Aggressive Scores*

5.2.4 kNN

For kNN it is best to sketch the number of clusters used for kmeans. With each number of clusters, a new accuracy score is achieved.

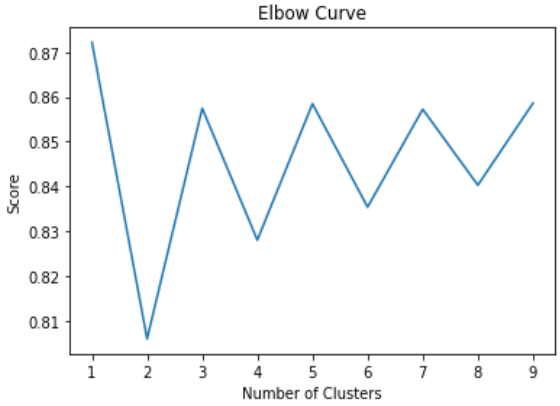

Figure 11: *Number of Clusters and Accuracy Score*

However, since this graph does not represent the usual elbow curve. We decided to normalize the values.

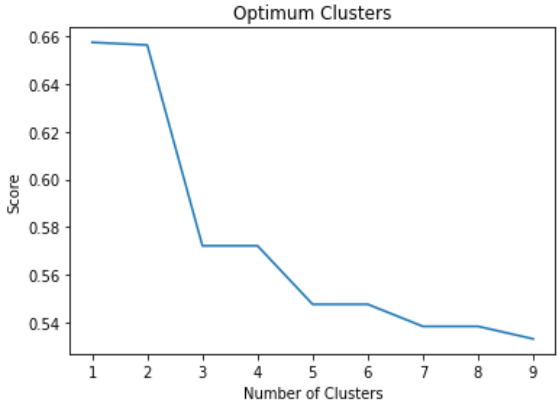

Figure 12: *Elbow Curve*

From the familiar concept of the elbow curve, the above graph gives a clearer picture about the accuracy. Despite the accuracy being lower at 57%, it still is a better choice than overly simplifying it. It seems that the steps for pre-processing did not yield much significance. In fact, accuracy is higher in the logistic regression and passive aggressive models. Given that we have a biased dataset the results are as expected. Also, the results confirm the same findings of Ahmed et. al [1] as the kNN has proved to be the worst model for this dataset.

6. Conclusions

The issue of fake news seems certainly is here to stay, various adverse entities are utilizing the online digital space to spread different types of false information, whether for political or economic gain. The present study adds to the scarce literature review about fake news detection by proposing a system that applies various text analysis techniques and feeding them to multiple machine learning models. Findings suggest that passive aggressive models are the most accurate when it comes to fake news text classification with 99.51%. This is a slight increase from the proposed 98% of Ahmed et. al. n-grams based model. No significant difference has been witnessed as a result of removing duplicates or adding POS tags in the preprocessing stage. Also, despite efforts and techniques to reduce bias and skewness stemming from the fake news data to produce a clean preprocessed data that can substitute Kaggle's biased set, we were not able to do so. We assume that the produced dataset has been purposefully biased with the inclusion of various twitter handles and incorrect preprocessing techniques. Further research is needed to gain more quality dataset to assess the accuracy of our proposed model and produce more meaningful results. Also, different techniques and comparisons can be made using other types of data preprocessing and analysis techniques. Namely, syntactic analysis, principal component analysis (PCA) or dimensionality reduction with T-SNE which are some deep learning techniques.

7. Contributions

This project has been a new endeavor for me. Certainly, I faced challenges during the semester and some unfamiliarity with NLP, yet, I found this project very useful to apply and hone various skills, most importantly, data analysis and critical thinking. Despite having a skewed result, I am proud to have been able to look critically at data and be careful when choosing datasets and not take the results at face value. I was also able to demystify somewhat the available dataset and to conclude with the reason of why I have such a high accuracy alongside others who have trained their data using this dataset. Finally, I was able to learn some techniques and mechanisms for NLP projects which is a definite added bonus to the Machine Learning course outcomes! By: Osama Shamout. For Dr. Sirine Taleb, I would like thank you for the guidance and assistance, thank you for letting me explore a topic that is of keen interest to me, and thank you for making us enjoy the Machine Learning course. It has truly been one of most valuable courses thus far in my academic journey.

[]

References

[1] Hadeer Ahmed, Issa Traore, and Sherif Saad. Detecting opinion spams and fake news using text classification. *SECURITY AND PRIVACY*, 1(1):e9, 2018.

[2] Hunt Allcott and Matthew Gentzkow. Social media and fake news in the 2016 election. *Journal of Economic Perspectives*, 31:211–236, 05 2017.

[3] L. Belcastro, R. Cantini, F. Marozzo, D. Talia, and P. Trunfio. Learning political polarization on social media using neural networks. *IEEE Access*, 8:47177–47187, 2020.

[4] Alessandro Bessi and Emilio Ferrara. Social bots distort the 2016 u.s. presidential election online discussion. *First Monday*, 21, 11 2016.

[5] Nancy Bilyeau. 'death to the decadent republic': How social media fueled dc riot, 01 2021.

[6] Riccardo Cantini, Fabrizio Marozzo, Domenico Talia, and Paolo Trunfio. Analyzing political polarization on social media by deleting bot spamming. *Big Data and Cognitive Computing*, 6:3, 01 2022.

[7] Michał Choraś, Konstantinos Demestichas, Agata Giełczyk, Álvaro Herrero, Paweł Ksieniewicz, Konstantina Remoundou, Daniel Urda, and Michał Woźniak. Advanced machine learning techniques for fake news (online disinformation) detection: A systematic mapping study. *Applied soft computing*, 101:107050, 2021.

[8] Jozef Kapusta, Petr Hájek, Michal Munk, and L'ubomír Benko. Comparison of fake and real news based on morphological analysis. *Procedia Computer Science*, 171:2285–2293, 2020. Third International Conference on Computing and Network Communications (CoCoNet'19).

[9] KyungWoo Kim, Hyeon-Suk Lyu, and Do Young Gong. Weeding out false information in disasters and emergencies: information recipients' competency. *International Review of Public Administration*, 25(4):261–278, 2020.

[10] Justin Pelletier. Intelligence, information warfare, cyber warfare, electronic warfare – what they are and how Russia is using them in Ukraine. `https://theconversation.com/intelligence-information-warfare-cyber-warfare-elect` 2022. [Online; accessed 8-December-2022].

[11] Elisa Shearer and Jeffrey Gottfried. News use across social media platforms 2017, 09 2017.

[12] Statista. Number of internet and social media users worldwide as of July 2022. `https://www.statista.com/statistics/617136/digital-population-worldwide/`, 2022. [Online; accessed 8-December-2022].