EAST WEST UNIVERSITY

Fall-2021

Project Report on

Fake News Detector

Course Code: CSE475

Course Title: Machine Learning

Section: 02

Group Information:

Member Name	Member ID
Abdul Kyume	2018-1-60-212
Md. Nahid	2018-1-60-213
Md. Musfiqur Rahman	2018-1-60-215
Md. Irfan Azad	2018-1-60-165

Submitted To:
MMDR (Md. Mahmudur Rahman)
Adjunct Faculty
Department of Computer Science & Engineering

1. Introduction

The world is currently revolved around the concept of news. It not only gives us a thorough explanation of the situation around the world, but also serves as entertainment purpose as well. However, a propaganda that is currently present and serves as an endangered element to the news world is the widespread of news that are false. The term 'Fake news' is a kind of news that some people or organization spread intentionally to misguide and betray the expectations and beliefs of the reader. It is generally published in the form of genuine news so that no one recognize it as fake. This spreading of the fake news has been causing a problem among the general citizens for quite a long time. While no way of differentiating between them, the problem is still at large today. The problem is now even faster with the introduction of social media. It is a matter of great concern since fake news can manipulate people in more ways than once. It is not only creating a negative impact among societies, but also in terms of individuals as well. This is why, we developed our project to detect fake news using a model which can accurately determine if an article or news is fake or real using machine learning approach such as Logistic Regression and NLP techniques.

1.1. Objectives:

Our main objective of the project is to identify whether some news is real or fake from a dataset based on a list of real and fake news compiled from Kaggle. Moreover, some other objectives of this project are:

1. Taking the help of linguistic cues so that we can develop a machine learning based model to determine whether the news from the dataset is fake or not.
2. To generate high accuracy level in order to conclude whether the articles are fake or not.

1.2. Motivation:

The project we were tasked to complete was to be based on machine learning. As we were browsing through the internet brainstorming ideas on a good topic, we came across a lot of misleading websites which not only wasted our precious time but was an irritating experience as well. Upon some investigation, we found that fake news is harming the general people in many ways possible. The extensive spread of fake news is continuing to create a major negative impact on the general citizens of a society. It has taken down the authenticity of the news ecosystem as it is spreading even more faster on social media than various popular verified news portals. It is becoming one of the biggest problems which can change opinions and influence decisions and interrupt the way of people responding to real news. We aim to establish a model by only using the machine learning algorithms to summarize whether some news are fake or not.

1.3. Existing Works:

Several algorithms were used before in order to create this project. Various papers were published as well. Various approaches were taken as well. We present two of the works we studied on before engaging on our project:

Fake News Detection Using Logistic Regression and Boolean crowdsourcing algorithms: Social network sites (SNSs) are continuing to be transformed the way in which the spread of data is done by allowing the people to freely share any type of contents between them. Therefore, SNSs are also gradually getting used to as vectors for the dispersion of hoaxes and misinformation. In this experiment, the performance of the logistic regression and Boolean crowdsourcing algorithms via a set of experimental datasets was characterized. In general, by conducting laborious process of physical post inspection, the results indicated how much more of the investment were needed in physical labeling, which was likely the key to reap the benefits of automated classification. The second set of experiments measured how much information learning is needed to transfer from one set of pages to another[1].

Fake News Detection Using Knowledge Verification and Natural Language Processing: The major purpose was to explore the 'fake news' as a disinformation implement. They tried to develop a newer and more solid explanation of 'fake news' in terms of comparative bias and factual accuracy. For detecting fake news in the datasheet, NLP model was used as a proposed solution. This classification model generally consists of several NLP features which are later combined with some knowledge verification methods to maintain the reliability of the source. For textual input, NLP features like stop-words percentage, title length, readability and sentiment score had been used for analysis. On the other hand, knowledge verification features id the answer to the question whether the title look like several sources or however the label seems to be based on actual facts or public opinion[2].

1.4. Necessity:

News are the source of knowledge and source of power for a lot of people around the world. Not everyone engages in watching televisions or have the luxury to own one. However, in the era of internet, news is easier to search and read. That is why news should be filtered and an approach should be made so that the real news is to be reached to the general people. Fake news can change the opinion of a person towards something, can create imbalance and hate, can manipulate people into doing wrongdoings. A great example of fake news is the US election of 2016 where Donald Trump hired a data processing company called 'Cambridge Analytica' to advertise his election campaign. However, his hidden motive was to manipulate the general people into voting him and

create hate towards his opposing party. Therefore, fake news detection is a necessity on the slowly developing field of the news platform on the internet.

2. Methodology:

We have used different kinds of algorithm in order to train and test our dataset. In different algorithm we have find different kinds of algorithm showing different kinds of result. Every algorithm has different kind of approach in order to train dataset and because of that we have got different accuracy for same dataset. We have used:

- Logistic Regression
- Multinomial Naïve Bayes
- Decision Tree Classifier
- Random Forest Classifier

3. Implementation:

3.1. Data Collection:

For conducting our project fake news detector, we collected our train dataset from Kaggle. Train dataset link is - https://www.kaggle.com/c/fake-news/data

Train dataset attributes are given below:

- id: for each article of distinctive id
- title: news article's title
- author: news article's author
- text: news article's text. It might be empty.
- Label: indication false and real news. Here,
  - 1 is indicating to False News
  - 0 is indicating to Real News

3.2. Data Processing:

In order to train our train our dataset we must process our data in the dataset. For processing our data, we must do follow some steps:

- id in our dataset is not necessary in order to train our dataset. So, we have dropped this column from our dataset.
- In the dataset, there are some null values, we have replaced the null values with empty strings.
- In order to train our dataset, we have used news's author's name and the news's title. We joined author's name and news title together and labeled that as content.
- We only kept letters in the content and delete all the other things.
- We changed all the letters into small letters.

- Then we delete unnecessary words like 'me', 'i', 'my', 'myself', 'we', 'our', 'ours', 'ourselves', 'you', "you're", "you've", "you'll", "you'd", 'your', 'yours', 'yourself', 'yourselves', 'he', 'him', 'his', 'himself', 'she', "she's", 'her', 'hers', 'herself', 'it', "it's", 'its', 'itself', 'they', 'them', 'their', 'theirs', 'themselves', 'what', 'which', 'who', 'whom', 'this', 'that', "that'll", 'these', 'those', 'am', 'is', 'are', 'was', 'were', 'be', 'been', 'being', 'have', 'has', 'had', 'having', 'do', 'does', 'did', 'doing', 'a', 'an', 'the', 'and', 'but', 'if', 'or', 'because', 'as', 'until', 'while', 'of', 'at', 'by', 'for', 'with', 'about', 'against', 'between', 'into', 'through', 'during', 'before', 'after', 'above', 'below', 'to', 'from', 'up', 'down', 'in', 'out', 'on', 'off', 'over', 'under', 'again', 'further', 'then', 'once', 'here', 'there', 'when', 'where', 'why', 'how', 'all', 'any', 'both', 'each', 'few', 'more', 'most', 'other', 'some', 'such', 'no', 'nor', 'not', 'only', 'own', 'same', 'so', 'than', 'too', 'very', 's', 't', 'can', 'will', 'just', 'don', "don't", 'should', "should've", 'now', 'd', 'll', 'm', 'o', 're', 've', 'y', 'ain', 'aren', "aren't", 'couldn', "couldn't", 'didn', "didn't", 'doesn', "doesn't", 'hadn', "hadn't", 'hasn', "hasn't", 'haven', "haven't", 'isn', "isn't", 'ma', 'mightn', "mightn't", 'mustn', "mustn't", 'needn', "needn't", 'shan', "shan't", 'shouldn', "shouldn't", 'wasn', "wasn't", 'weren', "weren't", 'won', "won't", 'wouldn', "wouldn't" etc. from our content.

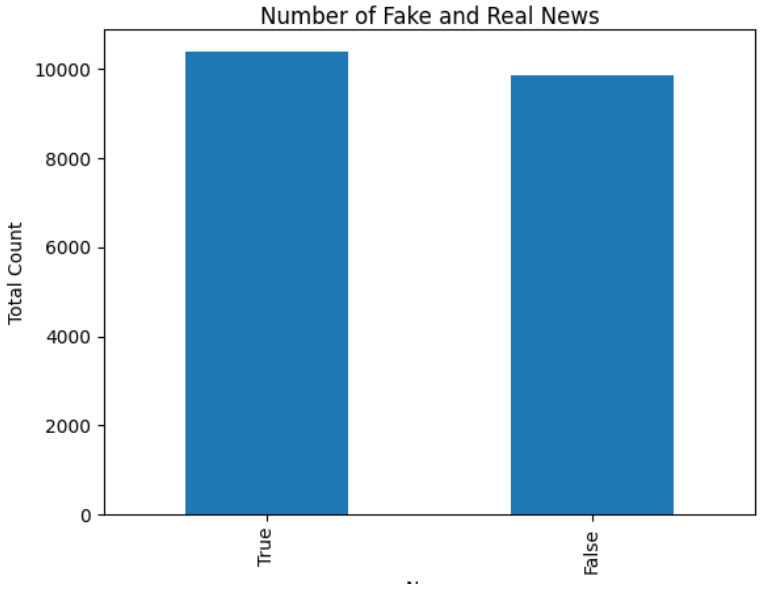

Figure 1: Total Real and Fake News

In the Figure 1, we can see that the total number of real and false news in our dataset in a bar chart.

```
Dataset After Stemming:
0        darrel lucu hous dem aid even see comey letter...
1        daniel j flynn flynn hillari clinton big woman...
2                   consortiumnew com truth might get fire
3        jessica purkiss civilian kill singl us airstri...
4        howard portnoy iranian woman jail fiction unpu...
                              ...
20795    jerom hudson rapper trump poster child white s...
20796    benjamin hoffman n f l playoff schedul matchup...
20797    michael j de la merc rachel abram maci said re...
20798    alex ansari nato russia hold parallel exercis ...
20799                             david swanson keep f aliv
Name: content, Length: 20800, dtype: object
```

Figure 2:Dataset After Stemming Process

In the figure 2, we can see the dataset after the stemming process. In the stemming process we only kept the letters and delete all the numerical data, commas and many more things. Then we convert all the letters into small letters. Then we have deleted all the unnecessary words from the dataset and finally by joining all the words we have prepared our dataset.

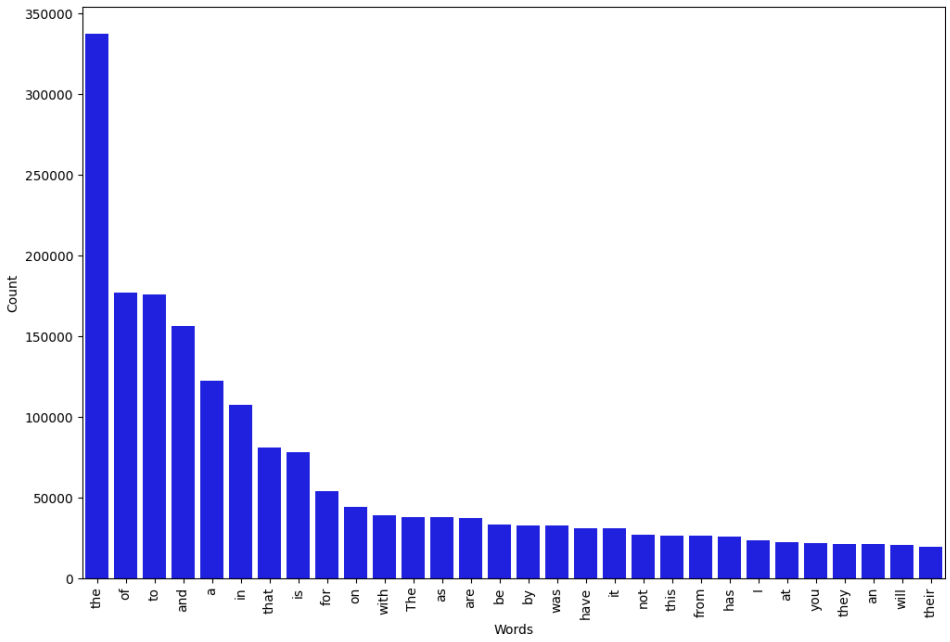

Figure 3: Word Count of False News

In the figure 3, we can see the words count of the false news from the dataset. This bar chart represents the chain of most used of words that were used in the false news. Since there were a lot of words, so we are only showing the first fifty most used words in the chart from the dataset.

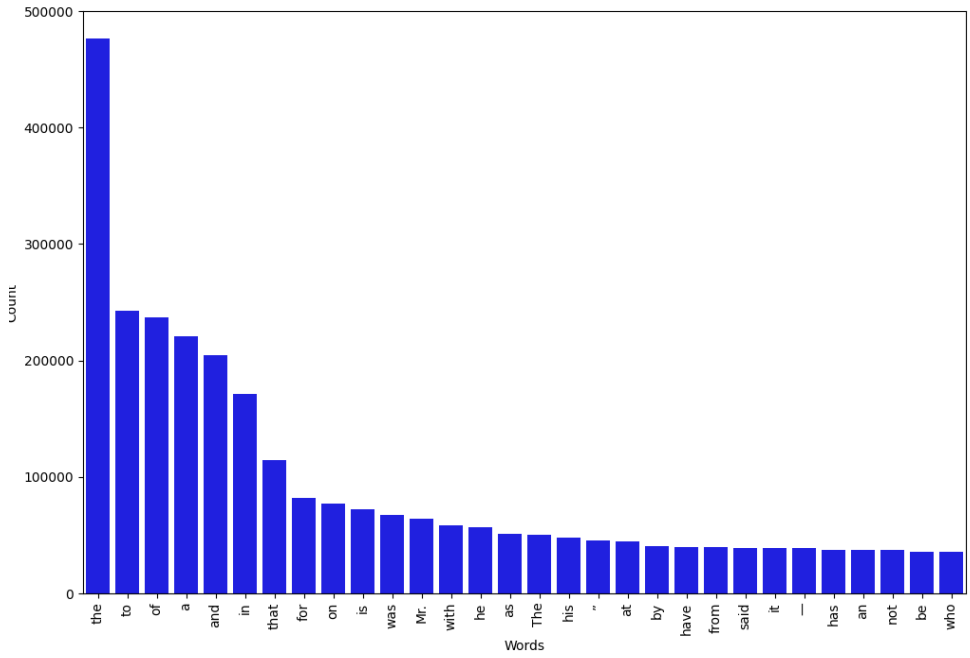

Figure 4:Word Count of Real News

In the figure 4, we can see the words count of the real news from the dataset. This bar chart represents the chain of most used of words that were used in the real news. Since there were a lot of words, so we are only showing the first fifty most used words in the chart from the dataset.

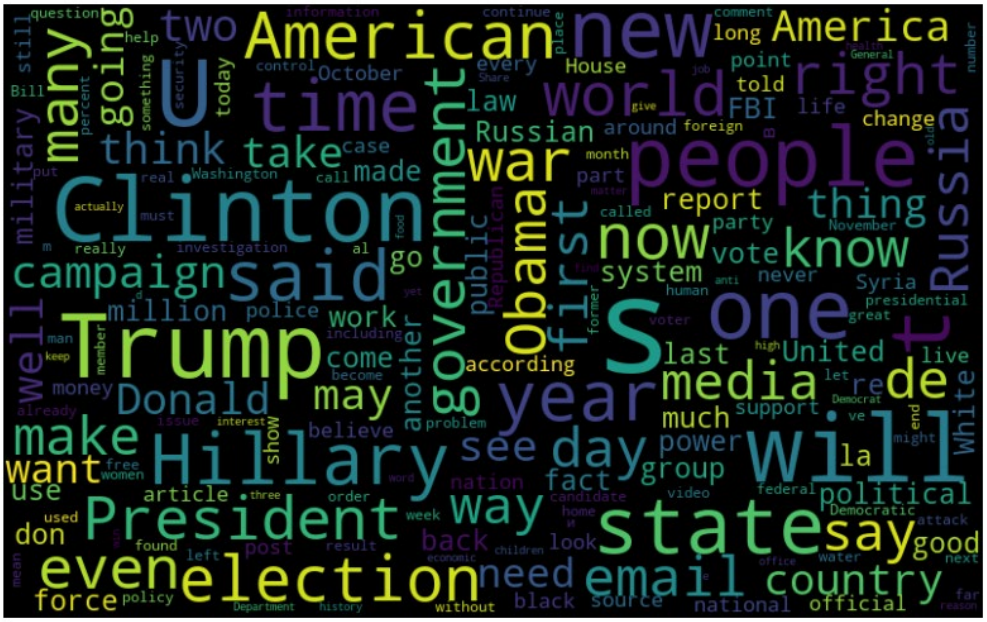

Figure 5: Word Cloud from False News

In the figure 5, we can see the word cloud of false news in a visualization method. Here we can see the most used words in a false news from the dataset.

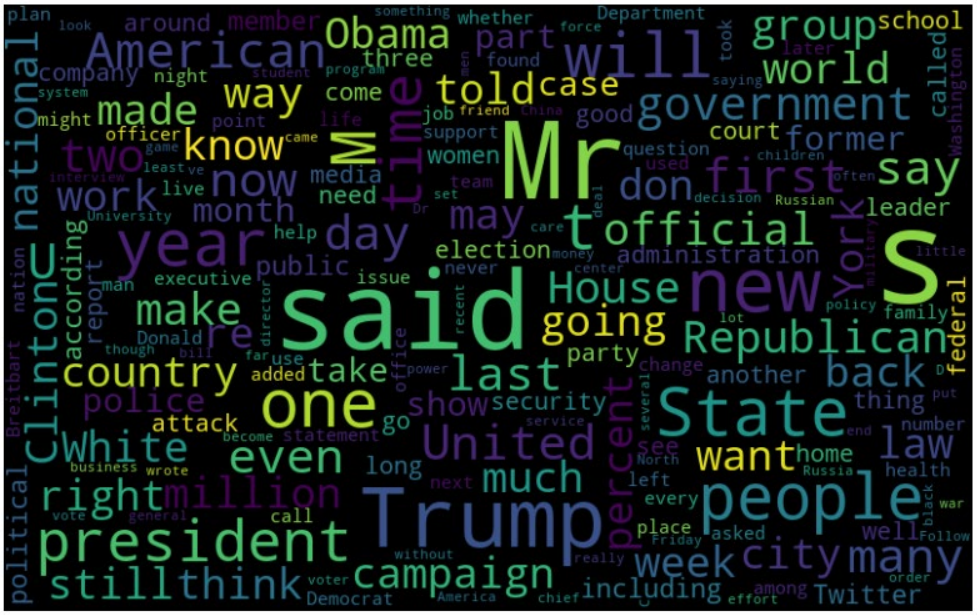

Figure 6: Word Cloud from Real News

In the figure 6, we can see the word cloud of real news in a visualization method. Here we can see the most used words in a real news from the dataset.

3.3. Model Development:

For developing our model, we have used python interpreter and we have used IDE-Pycharm Community Edition. We have used some library functions such as 'numpy', 'pandas', 're', 'matplotlib', 'seaborn', 'nltk', 'itertools', 'sklearn', 'wordcloud'.

3.4. Results:

We have tested several methods and algorithm to train and test our dataset. We get excellent results from those methods and algorithm. The tested algorithm results are given below:

3.4.1. CountVectorizer & Logistic Regression

```
CountVectorizer & Logistic Regression:

Best score: 0.9859374023807699
Train score 0.9975360576923077
Test score 0.989423076923077
```

Figure 7: Accuracy Score using CountVectorizer & Logistic Regression

In the figure 7, we can see that using countvectorizer and logistic regression we are getting these scores. We are seeing that the train dataset score is 99 % which is showing a great training score. The test dataset score is almost near to 99 % which is a great result for the dataset. Confusion matrix for this method is given below:

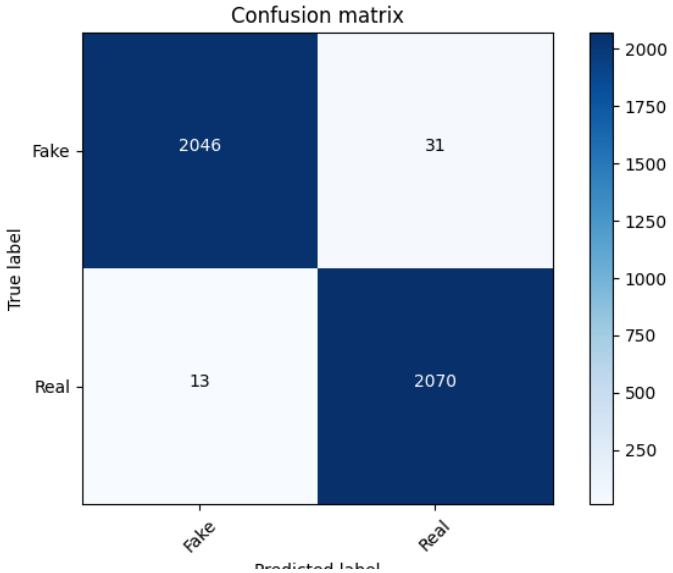

Figure 8: Confusion Matrix using CountVectorizer & Logistic Regression

In the figure 8, we can see the confusion matrix, which was built by using count vectorizer method and logistic regression algorithm and it shows how much accurate result is giving this model on the dataset.

3.4.2. TfidfVectorize & Logistic Regression

```
TfidfVectorize & Logistic Regression:

Best score: 0.9861778917390698
Train score 0.993329326923077
Test score 0.9901442307692307
```

Figure 9: Accuracy Score using CountVectorizer & Logistic Regression

In figure 9 we can see the train and test dataset scores both are 99% and the best score for this method is showing 98%. Confusion matrix for this model is given below:

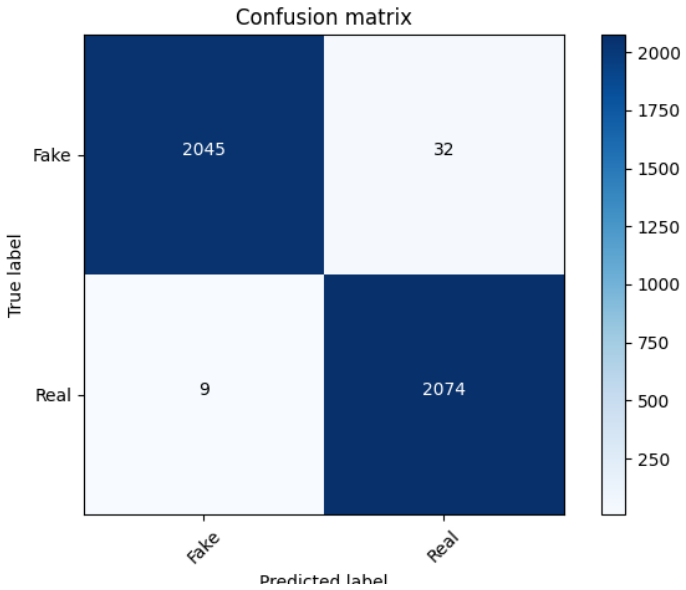

Figure 10: Confusion Matrix using TfidfVectorize & Logistic Regression

In the figure 10, we can see the confusion matrix, which was built by using TfidVectorizer method and logistic regression algorithm and it shows how much accurate result is giving this model on the dataset.

3.4.3. CountVectorizer & MultinomialNB

```
CountVectorizer & MultinomialNB:

Best score: 0.9775239371697687
Train score 0.9984375
Test score 0.979326923076923
```

Figure 11: Accuracy Score using CountVectorizer & MultinomialNB

In the figure 11, we can see the train and test dataset accuracy score using multinomial naïve bayes algorithm. The train dataset is showing 99% score and the test dataset is showing 97% score. For this algorithm we are getting 97% accuracy for our dataset. Confusion matrix for this algorithm is given below:

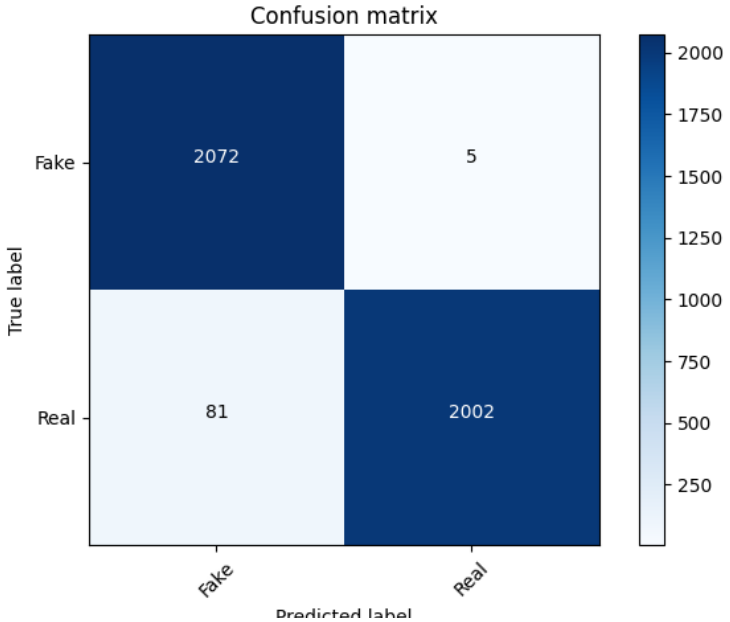

Figure 12: Confusion Matrix using CountVectorizer & MultinomialNB

In the figure 12, we can see the confusion matrix, which was built by using CountVectorizer method and MultinomialNB algorithm. Here we can see that the confusion matrix is showing excellent result in showing the correct result.

3.4.4. TfidfVectorizer & MultinomialNB

```
TfidfVectorizer & MultinomialNB:

Best score: 0.9713942307692307
Train score 0.9861778846153846
Test score 0.9711538461538461
```

Figure 13: Accuracy Score using CountVectorizer & MultinomialNB

In the figure 13, we can see that the in the TfidVectorizer method using multinomial naïve bayes algorithm we are getting 98% accuracy score for the dataset which is used for training and 97% accuracy score for the dataset which is for testing. Best score for our train and test dataset we are getting 97%. Confusion matrix for this method is given below:

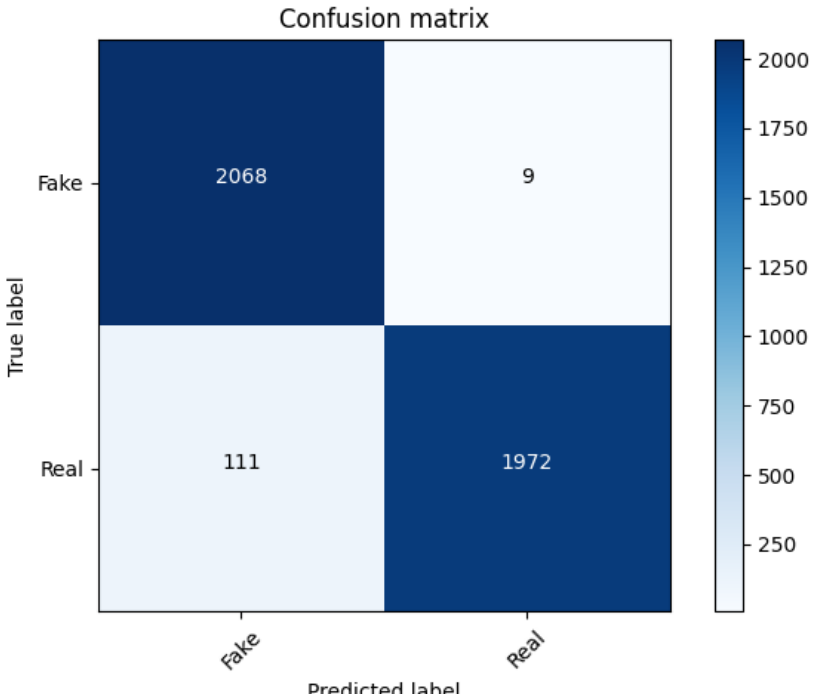

Figure 14: Confusion Matrix using TfidfVectorizer & MultinomialNB

In the figure 14, we can see the confusion matrix, which was built by using TfidfVectorizer method and MultinomialNB algorithm.

3.4.5. Decision Tree Classifier

```
Decision Tree Classifier:

Score:  0.9848557692307692
```

Figure 15: Accuracy Score using Decision Tree Classifier

In the figure 15 we can see that using countvectorizer method along with tfidtransformer and applying logistic regression algorithm we are getting accuracy score for our dataset is 98%. This is known as decision tree classifier accuracy score. Confusion matrix for decision tree classifier is given below:

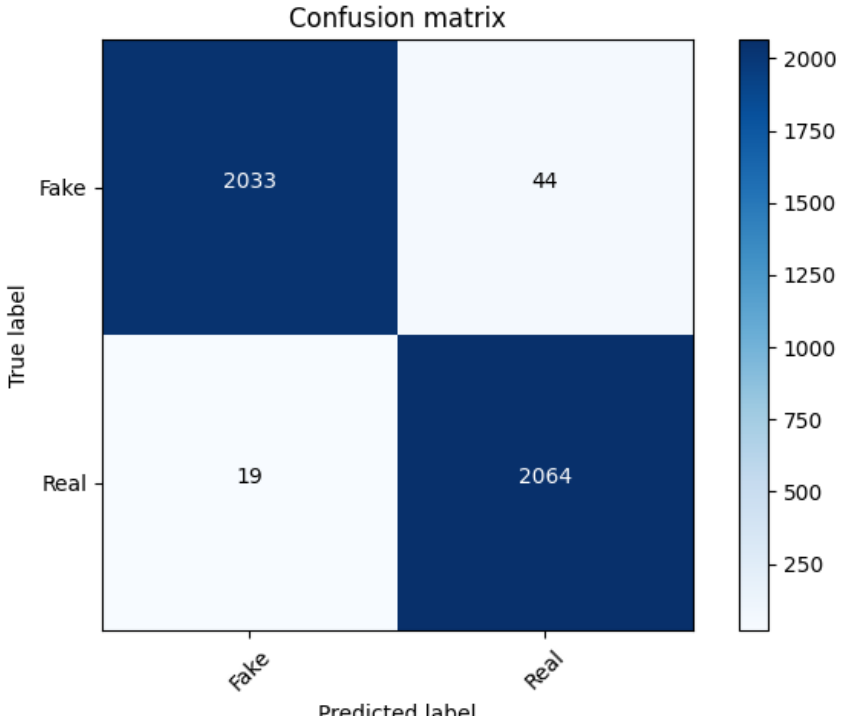

Figure 16: Confusion Matrix using CountVectorizer, TfidfTransformer & Logistic Regression

In the figure 16, we can see the confusion matrix, which was built by using TfidfVectorizer method, TfidfTransformer method and MultinomialNB algorithm. Here we can see that the confusion matrix is showing excellent result in showing the correct result.

3.4.6. Random Forest Classifier

```
Random Forest Classifier :

accuracy:  0.9923076923076923
```

Figure 17: Accuracy Score using Random Forest Classifier

In the figure 17, we are seeing that using random forest classifier algorithm on our dataset we are getting 99% accuracy result on our dataset. Confusion matrix for this model is given below:

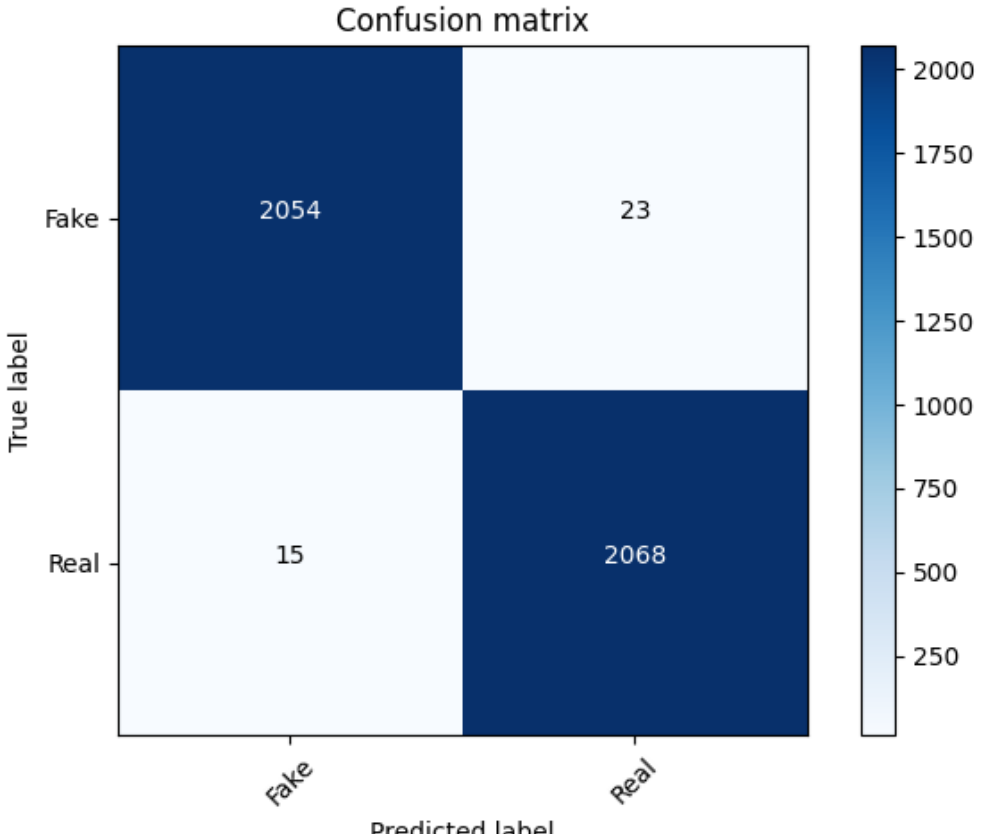

Figure 18: Confusion Matrix using CountVectorizer, TfidfTransformer & RandomForestClassifier

In the figure 18, we can see the confusion matrix, which was built by using TfidfVectorizer method, TfidfTransformer method and RandomForestClassifier algorithm.

4. Conclusion:

The vast spreading of fake news can create a bad impact on people. People are misled by fake news it is also the ultimate confusion. People with their naked eyes cannot distinguish between real news and fake news. And this is a real danger of fake news. We can predict news to be real or fake by using the power of machine learning and with this power, we can solve this problem.

4.1. Challenges:

In this project, we have used different kinds of algorithms. By using these algorithms, we can train and test our dataset and can find different kinds of results. But before using these algorithms we didn't know which algorithm is good for our dataset and which one would give us the accurate result. So, this was our challenge for our project. We have used - Multinomial Naïve Bayes, Logistic Regression, Random Forest Classifier, Decision Tree Classifier.

4.2. Limitations:

In this project, we have used different kinds of algorithms. We have used - Multinomial Naïve Bayes, Logistic Regression, Random Forest Classifier, Decision Tree Classifier. By using these algorithms, we train, test, and preprocess our dataset for getting results. But if we change our dataset our algorithms will not give an accurate result. For different datasets, we need to train, test and pre-process the dataset for getting results again. And this is our limitation.

Fake news spread very rapidly and reaches on social media. Without collaborative access and compromising speed it's hard to study and design technological, computational, and combat business strategies. Normally, fake news is related to our emotions, ideological prejudices, exploiting our cognitive skills, so it's hard to identify. Also, detecting fake news by using computational methods is challenging. Lack of Susceptibility and public awareness are also hard to control fake news. Social media users fail to differentiate between legitimate news and false and fake news can easily reach a huge number of people in a short amount of time.

4.3. Future Directions:

In this project, we have used different kinds of algorithms. We have used - Multinomial Naïve Bayes, Logistic Regression, Random Forest Classifier, Decision Tree Classifier. In the future, we will implement other algorithms of machine learning. We will try to find more accurate percentages than our current algorithms can do. We can use our system for new and for large number data set in the future. That will help us to understand the latency level and the computational speed in advance. In the future, we can also implement this same dataset for the different machine learning algorithms. For doing this we can compare which model or algorithm is better and outperformed.

References:

[1] E. Tacchini, G. Ballarin, M. L. della Vedova, S. Moret, and L. de Alfaro, “Some like it Hoax: Automated fake news detection in social networks,” in *CEUR Workshop Proceedings*, 2017, vol. 1960.

[2] M. D. Ibrishimova and K. F. Li, “A machine learning approach to fake news detection using knowledge verification and natural language processing,” in *Advances in Intelligent Systems and Computing*, 2020, vol. 1035. doi: 10.1007/978-3-030-29035-1_22.