FAKE NEWS DETECTION

Major Project Report

Submitted in partial fulfilment of the requirement of the degree of

BACHELOR OF COMPUTER APPLICATION

to

K.R Mangalam University

By

AAYUSH UJJWAL (2201062040)

Under the supervision of

JYOTI KATARIA

Associate Professor, SOET

Department of Computer Science and Engineering

School of Engineering and Technology

K.R Mangalam University, Gurugram- 122001, India

May 2024

CERTIFICATE

This is to certify that the Project Synopsis entitled, “**FAKE NEWS DETECTION**” submitted “**Aayush Ujjwal (2201062040)”** to **K.R Mangalam University, Gurugram, India,** is a record of bonafide project work carried out by them under my supervision and guidance and is worthy of consideration for the partial fulfilment of the degree of **Bachelor of Computer Application** in **Computer Science and Engineering** of the University.

Jyoti Kataria	Dr. Pankaj Aggarwal
Associate Professor, SOET	Dean SOET

Date: 31/07/2024

DECLARATION

We declare that this written submission represents our ideas in our own words and where other’s ideas or words have been included, we have adequately cited and referenced the original sources. We also declare that we have adhered to all the principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in our submission. We understand that any violation of the above will cause disciplinary action by the Institute and canal so evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed. We further declare that if there is any violation of the intellectual property right or copyright, my supervisor and university should not be held responsible for the same.

Aayush Ujjwal (2201062040)

Place: K.R. MANGALAM UNIVERSITY

Date: 31/07/2024

ACKNOWLEDGEMENT

It gives us immense pleasure to express my deepest sense of gratitude and sincere thanks to my highly respected and esteemed guide Jyoti Kataria the Associate Professor, School of Engineering and Technology, for her valuable guidance, encouragement and help for completing this work. Her useful suggestions for this whole work and co-operative behaviour are sincerely acknowledged.

I would like to express my sincere thanks to **Jyoti Kataria** for giving me this opportunity to undertake this project. We would also like to thank my HOD Dr. Shweta A. Bansal and School Dean (SOET) DR. PANKAJ AGGARWAL for their wholehearted support.

I also wish to express my indebtedness to my parents as well as my family member whose blessings and support always helped me to face the challenges ahead. At the end would like to express my sincere thanks to all my friends and others who helped me directly or indirectly during this project work.

Place: - K.R. Mangalam University

Date: - 31/07/2024

Aayush Ujjwal

2201062040

INDEX

ABSTRACT

With the recent social media boom, the spread of fake news has become a great concern for everybody. It has been used to manipulate public opinions, influence the election. A 2018 MIT study found that fake news spreads six times faster on Twitter than real news. The credibility and trust in the news media are at an all-time low. It is becoming increasingly difficult to determine which news is real and which is fake. Various machine learning methods have been used to separate real news from fake ones. In this study, we tried to accomplish that using Natural Language Processing. There are lots of machine learning models that we can use to have better results.

Now there is some confusion present in the authenticity of the correctness. But it definitely opens the window for further research. There are some of the aspects that has to be kept in mind considering the fact that fake news detection is not only a simple web interface but also a quite complex thing that includes a lot of backend work.

Chapter 1: INTRODUCTION

1.1. INTRODUCTION OF THE PROJECT

Fake news is untrue information presented as news. It often has the aim of damaging the reputation of a person or entity or making money through advertising revenue. Once common in print, the prevalence of fake news has increased with the rise of social media, especially the Facebook News Feed. During the 2016 US presidential election, various kinds of fake news about the candidates widely spread in the online social networks, which may have a significant effect on the election results. According to a post-election statistical report, online social networks account for more than 41.8% of the fake news data traffic in the election, which is much greater than the data traffic shares of both traditional TV/radio/print medium and online search engines respectively. Fake news detection is becoming increasingly difficult because people who have ill intentions are writing the fake pieces so convincingly that it is difficult to separate from real news. What we have done is a simplistic approach that looks at the news text and tries to predict whether they may be fake or not.

Fake news can be intimidating as they attract more audience than normal. People use them because this can be a very good marketing strategy. But the money earned might not live upto fact that it can harm people.

Keywords:

- Fake news prediction,
- Data Analysis,
- Data pre-processing,
- Data Visualization,
- Word Cloud,
- Stop Words,
- Logistic Regressions,
- Decision Tree,
- Gradient Boosting,
- Random Forest,
- Natural Language Processing (NLP).

1.2. MOTIVATION

Social media facilitates the creation and sharing of information that uses computer-mediated technologies. This media changed the way groups of people interact and communicate. It allows low cost, simple access and fast dissemination of information to them. The majority of people search and consume news from social media rather than traditional news organizations these days. On one side, where social media have become a powerful source of information and bringing people together, on the other side it also puts a negative impact on society. Look at some examples herewith; Facebook Inc's popular messaging service, WhatsApp became a political battle-platform in Brazil's election. False rumours, manipulated photos, de-contextualized videos, and audio jokes were used for campaigning. These kinds of stuff went viral on the digital platform without monitoring their origin or reach. A nationwide block on major social media and messaging sites including Facebook and Instagram was done in Sri Lanka after multiple terrorist attacks in the year 2019. The government claimed that "false news reports" were circulating online. This is evident in the challenges the world's most powerful tech companies face in reducing the spread of misinformation. Such examples show that Social Media enables the widespread use of "fake news" as well. The news disseminated on social media platforms may be of low quality carrying misleading information intentionally. This sacrifices the credibility of the information. Millions of news articles are being circulated every day on the Internet – how one can trust which is real and which is fake? Thus, incredible or fake news is one of the biggest challenges in our digitally connected world. Fake news detection on social media has recently become an emerging research domain. The domain focuses on dealing with the sensitive issue of preventing the spread of fake news on social media. Fake news identification on social media faces several challenges. Firstly, it is difficult to collect fake news data. Furthermore, it is difficult to label fake news manually. Since they are intentionally written to mislead readers, it is difficult to detect them simply based on news content. Furthermore, Facebook, WhatsApp, and Twitter are closed messaging apps.

The misinformation disseminated by trusted news outlets or their friends and family is therefore difficult to be considered as fake. It is not easy to verify the credibility of newly emerging and time-bound news as they are not sufficient to train the application dataset. Significant approaches to differentiate credible users, extract useful news features and develop authentic information dissemination systems are some useful domains of research and need further investigations. If we can’t control the spread of fake news, the trust in the system will collapse. There will be widespread distrust among people. There will be nothing left that can be objectively used. It means the destruction of political and social coherence. We wanted to build some sort of web-based system that can fight this nightmare scenario. And we made some significant progress towards that goal.

Chapter 2: LITERATURE REVIEW

2.1. REVIEW OF EXIXTING LITERATURE

Traditional news media generally rely on news content for the identification of fake news, as opposed to social media, where additional social context auxiliary information can be used as supplementary information to help detect fake news. The use of supervised fake news detection models based on machine learning (ML) and deep learning (DL) techniques has significantly expanded in recent years due to their excellent detection accuracy. These methods extract the distinguishing characteristics of fake news using feature representation based on linguistic and visual data [1]. Linguistic-based characteristics are derived from many levels of textual content organization, such as characters, words, phrases, and documents. Visual-based features are derived from visual resources such as images and videos in order to recognize the numerous characteristics of fake news. With the reported increase in online fake news [2,3], automated methods for its detection on social media have attracted the attention of researchers worldwide [4,5,6]. COVID-19 and the numerous related hoaxes, rumors, and misinformation surrounding the cures, treatment, and prevention have further filled the interest of researchers in improved methods for detection [7]. Even with this increased attention, the task of detecting fake news is still reported as challenging [8].

Through the analysis of the literature relating to this area, it is evident that a diverse range of ML and DL approaches as well as hybrid and ensemble versions of these have been employed. This section presents the literature relating to the approaches mentioned above.

Several researchers have developed ML methods for the detection of fake news. Vicario et al. [9] built a logistic regression (LR) classifier to predict this type of news using a massive Italian dataset consisting of actual news and hoaxes published on Facebook, achieving an accuracy of 91%. The LR method also achieved the highest accuracy (96%) in the study by Stitini et al. [10] where Bidirectional Encoder Representations from Transformers (BERT) transformed the dataset text into vectors. Random forest (RF) often emerges as the method achieving the most accurate results, with an accuracy of 97.3% reported by Fayaz et al. [11]. The study used data from the ISOT fake news dataset and compared results with other state-of-the-art machine learning techniques such as gradient boosting machines (GBM), extreme gradient boosting machines (Boost), and the adaptive boost regression model. Support vector machine (SVM) models have also shown promising results, with an accuracy of 93.15% being achieved when applying the data from the fake news dataset extracted from Kaggle, outperforming the LR approach applied to the same data by 6.82% [12].

While many researchers investigate the performance of individual ML methods, some researchers chose to investigate the effect of applying an ensemble of ML methods on the data to achieve improved accuracy results. A blended ensemble machine learning method that applies the LR, SVM, linear discriminant analysis, stochastic gradient descent, and ridge regression techniques achieved 79.9% accuracy when data from the ISOT and LIAR datasets were used [13]. Accuracies over 95% have been achieved by many studies that have applied voting ensemble methods to the datasets, including Elsaeed et al. [14], Verma et al. [15], Biradar et al. [16], Kanagavalli and Priya [17], and Elhadad, Li, and Gebali [18], who achieved accuracy measures of 95.6%, 96.7%, 97%, 98.6%, and 99.7%, respectively. These works based their results on data from different datasets, including ISOT, WELFake, COVID19 Fake, LIAR, and researcher-created datasets.

DL methods such as convolutional neural networks (CNN), long-short term memory (LSTM), and bi-directional long-short term memory (BiLSTM) have attracted much interest in the area of fake news detection. Galli et al. [19] applied both ML and DL methods to datasets, comparing the results obtained. It was established that, by applying the CNN technique to the limited PoliFact dataset, an accuracy of 75.6% was achieved. The study reported that the CNN method outperformed the other approaches investigated, which include, among others, naive Bayes (NB); RF; LR; nearest neighbor (NN); decision tree; gradient boost; and BiLSTM. BiLSTM has also been investigated for its value in detecting fake news by many other researchers [20,21,22,23]. With most studies focusing on the English language, both Fouad, Sabbeh, and Medhat [21] and Nassif, Elnagar, Elgendy, and Afadar [22] investigated the accuracy of state-of-the-art classification methods for the identification of fake news in the Arabic language. Fouad, Sabbeh, and Medhat [21] evaluated the performance of eight machine learning algorithms and also experimented with five different combinations of deep learning algorithms, including CNN and LSTM, with the results indicating that the BiLSTM method outperformed the other methods, achieving an accuracy of 75% on the dataset of size 4561.

Nassif, Elnagar, Elgendy, and Afadar [22] created a customized dataset based on tweets that consisted of 5000 fake and 5000 true news instances. Their Arabic Bi-directional Encoder Representations from the Transformers model (ARBERT) achieved 98.8% accuracy on the data.

Ensemble deep learning approaches have also been investigated for their value as detection methods, with the novel MisRoBÆRTa technique proposed by Truică and Apostol [24]. The technique combines CNN and many BiLSTM, achieving an accuracy of 92.5% when tested on a dataset with a sample size of 100,000. Jang et al. [25] collected data from Twitter and classified tweets as fake news by using the temporal propagation pattern of the retweeted quotes. The authors applied a two-phase deep learning model based on CNN and LSTM for training and testing, achieving an accuracy measure of 85.7%. An ensemble-based deep learning technique for classifying news as real or fake achieved a significant accuracy of 89.8% using data from the LIAR dataset [26]. The approach used two deep learning models, with a Bi-LSTM-gated recurrent unit (GRU) being used for the textual "statement" attribute, while the deep dense learning model was used on the remaining nine attributes.

While these studies all reported on the accuracy of the employed methods, the literature also includes studies that survey and review current approaches. In the paper by Collins, Hoang, Nguyen, and Hwang [27], a synthesis of methods for combating misinformation and fake news on social media is presented, while possible solutions, methodological gaps, and challenges relating to current detection methods were presented in a systematic review by Choraś, Demestichas, Giełczyk, Herrero, Ksieniewicz, Remoundou, Urda, and Woźniak [28]. Similarly, Shahid, Jamshidi, Hakak, Isah, Khan, Khan, and Choo [30], through a survey of novel AI approaches, uncovered key challenges in the area while also highlighting potential future research to be considered. An approach-specific survey by Varlamis, Michail, Glykou, and Tsantilas [29] investigated and reported on the studies that apply graph convolutional networks (GCNs) for detecting rumors, fake content, and fake accounts, with the aim of the paper being to provide a starting point for those researchers wanting to further investigate GCNs for the detection of fake news. Both Khan, Hakak, Deepa, Dev, and Trelova [31] and Lozano, Brynielsson, Franke, Rosell, Tjörnhammar, Varga, and Vlassov [32] chose to rather review ML models, providing a set of advantages and disadvantages associated with the datasets used in the reviewed studies. Additionally, Shu, Sliva, Wang, Tang, and Liu [1] provided a thorough analysis from a data mining perspective and emphasized the future research prospects according to four categories: data-oriented, feature-oriented, model-oriented, and application-oriented. One of the potential study areas for fake news detection that Shu, Sliva, Wang, Tang, and Liu [1] suggested is model-oriented fake-news research, which opens the path for the development of more effective and useful models based on supervised and unsupervised approaches to fake news detection.

While the current literature provides insight into the latest methods being employed and highlights reviews that have been performed, there appears to be no single study that quantitatively analyzes the current methods proposed for fake news detection. Furthermore, no systematic, comprehensive study of model-oriented fake news detection based on supervised learning techniques such as ML, DL, and ensemble methods has been conducted. Lozano, Brynielsson, Franke, Rosell, Tjörnhammar, Varga, and Vlassov [32] also highlight the lack of literature that considers multiple datasets and multiple approaches for detection. With the increasing number of publications in this research area and the reported proliferation of fake news, a systematic analysis is required so that an objective and comprehensive understanding of current supervised approaches can be obtained. The results provide valuable insight to researchers in the field regarding the DL, ML, and ensemble methods that were applied. This study, therefore, aimed to identify current trends, approaches, and methods for online fake news detection. Through meta-analysis, the patterns and correlations that exist in the area of ML, DL, and ensemble methods were unveiled and reported on.

2.2. GAP ANALYSIS

From the numerous researches done for fake news detection by applying machine learning algorithms, artificial intelligence, Passive-aggressive classifier, logistic regression, LSTM can be used in fake news detection, the resultant detector is always a stronger product than the traditional version. But all these projects are carried out for the using purpose of professionals, elections or business field users, etc, but not many researches are carried out for the student’s understanding or for the student's beginning level projects, or, they do not provided the easiest set of codes required for the students understanding. Our project covers the beginning level knowledge for the students. Hence, we have covered a easiest coding part using NLP [33]model for the understanding of students with the good fitted algorithms or models as logistic regression, random forest, decision tree, gradient boosting according to their accuracy and performance metrices with the packages or libraries needed for them which will help the students to feel it as a simple

accessible codes with the help of a simple creation of web interface that will add up to the pros of the project.

We have used Term Frequency-Inverse Document Frequency which is also a method used to represent text in Fake News Detector a format that can be easily processed by machine learning algorithms. It is a numerical statistic that shows how important a word is to news in a news dataset. The importance of a word is proportional to the number of times the word appears in the news but inversely proportional to the number of times the word appears in the news dataset (fake or real) [34], it had reasonably good accuracy but if the news was a bit more sophisticated, it would be difficult to achieve good accuracy. Because this model picks up the sensational/clickbaity words as part of fake news. For example, if a news title says, 'Donald Trump is the greatest president ever, the model will pick it up as fake news with reasonable accuracy. If the title is more nuanced and written in a sophisticated way, it'd be difficult to do so. We believe that our NLP model is not enough by itself to detect fake news. Our model can act as a first step in detecting fake news and also for the beginning level project with the easiest way for the student's understanding.

2.3. PROBLEM STATEMENT

The advent of the World Wide Web and the rapid adoption of social media platforms (such as Facebook and Twitter) paved the way for information dissemination that has never been witnessed in the human history before. Besides other use cases, news outlets benefitted from the widespread use of social media platforms by providing updated news in near real time to its subscribers. The news media evolved from newspapers, tabloids, and magazines to a digital form such as online news platforms, blogs, social media feeds, and other digital media formats [35]. It became easier for consumers to acquire the latest news at their fingertips.

In this day and age, it is extremely difficult to decide whether the news we come across is real or not. There are very few options to check the authenticity and all of them are sophisticated and not accessible to the average person. There is an acute need for a web-based fact-checking platform that harnesses the power of Machine Learning to provide us with that opportunity.

Fortunately, there are a number of computational techniques that can be used to mark certain articles as fake on the basis of their textual content [36]. Majority of these techniques use fact checking websites such as "PolitiFact" and "Snopes." There are a number of repositories maintained by researchers that contain lists of websites that are identified as ambiguous and fake [37]. However, the problem with these resources is that human expertise is required to identify articles/websites as fake.

In this project we will work through NLP. We will remove the stopwords, use tf-idf vectorization, drop unnecessary columns, visualize our data and fit models like logistic regression, random forest, gradient boosting, decision tree and at last will create a web interface through which user can check the news article by giving them as an input and get their output as **" fake news or true news"** .

2.4. OBJECTIVES

Our project's primary goal is to determine the veracity of news in order to determine if it is real or phoney. the development of a machine learning model that would allow us to recognise bogus information.

It can be difficult to identify fake news only based on its content since it is intentionally produced to influence readers to believe false information.

The study has been carried out with the following objectives–

- To introduce the topic of fake news and the various machine learning algorithms we will build a model which accurately classify a piece of news as REAL or FAKE.
- Provided an overview of the history and implications of fake news.
- To detect bogus news we applied a range of methods and models through which machine learning makes it easy.
- To examine the relationship between two words, we will apply deep learning-based NLP.
- To eliminate stop words we will use this method as well.
- This advanced python project of detecting fake news deals with fake and real news. Using SK-Learn tools, we will a Tf-idfVectorizer on our dataset[34].
- To know how well our model fares, we will use various models, which will result in an accuracy score and a performance metrices
- To present a possible solution, at last we will create a web interface.

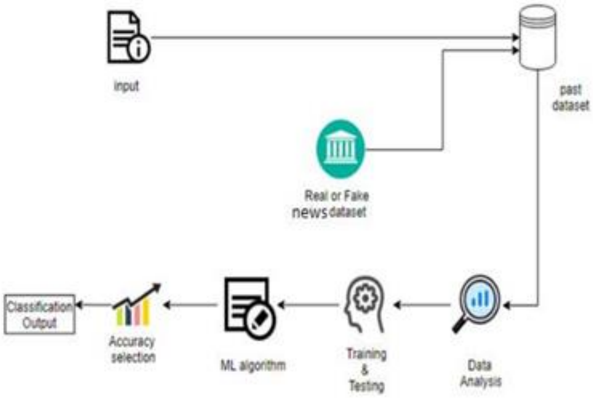


Fig 1. Architecture of the project

Chapter 3: REQUIREMENT ANALYSIS

3.1. Tools and Platforms Used

For this project, we have used various latest technologies :

- PROGRAMMING LANGUAGE: **PYTHON**

  We have used Python language as it is relatively new as compared to other languages like Java, C++, etc and comes with so many features. We can perform Machine Learning Models and algorithm with python and can create a web interface easily that can be achieved in Python.
  Python is a widely used general-purpose, high level programming language. It was designed with an emphasis on code readability, and its syntax allows programmers to express their concepts in fewer lines of code. Python is a programming language that lets you work quickly and integrate systems more efficiently. There are two major Python versions: Python 2 and Python 3.

- PLATFROM USED: **WINDOWS**

- MACHINE LEARNING ALGORITHM USED:
    - **LOGISTIC REGRESSION**
    - **DECISION TREE**
    - **GRADIENT BOOSTING**
    - **RANDOM FOREST**

3.2. ENVIRONMENTAL SETUP

SOFTWARE REQUIREMENTS

Below are the requirements to run this software :

1. Windows/Linux/Mac OS any version, hence it can run on any platform.
2. Python3, it needs python to be installed in system to run successfully.
3. Packages in python -
    a. NumPy;
    b. Panda;
    c. Seaborn;
    d. sklearn
    e. Nltk;
    f. Wordcloud;
    g. Streamlit;
    Etc.

HARDWARE REQUIREMENTS

In terms of hardware requirements there is not much required at all but still below requirements are must:

1. Working PC or Laptop
2. Proper Internet Connection

PLATFORMS ALREADY TESTED ON:

It is tested on Windows.

Chapter 4: PROPOSED METHODOLOGY

We will work according to these points -

- Collecting dataset related to Fake News Analysis.
- Loading and Analyzing dataset.
- Building of a new dataset from the set of two datasets.
- Preprocessing of the dataset.
- Visualization of the data according to the need and also used wordcloud for effective visualization.
- Splitting the dataset into training and testing sets.
- Choose a Learning Model or Schema for training the dataset.
- Fitting the Model with proper parameters and Pridicting a feasible outcome(likelihood).
- Determining the Model Accuracy Score.
- Testing the Model and creating a classification report of them.
- Testing the Model with a random news feed from the dataset to verify its performance and how well it can guess correctly.

4.1.FlowChart

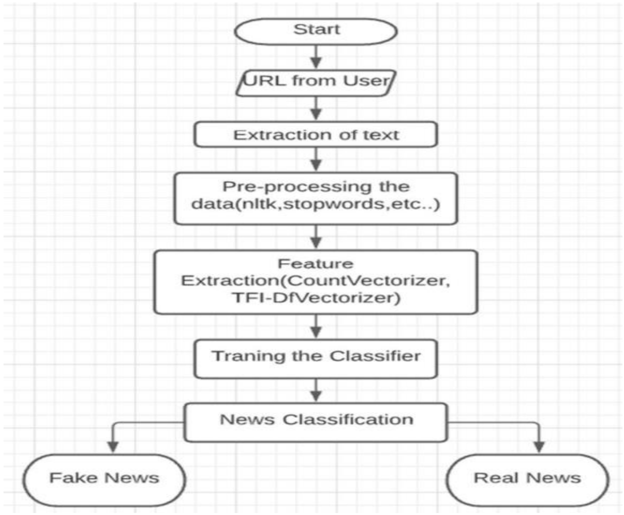

Start
URL from User
Extraction of text
Pre-processing the data(nltk,stopwords,etc..)
Feature Extraction(CountVectorizer, TFI-DfVectorizer)
Traning the Classifier
News Classification
Fake News
Real News

Fig 2. Flowchart for the project

4.2.Dataset

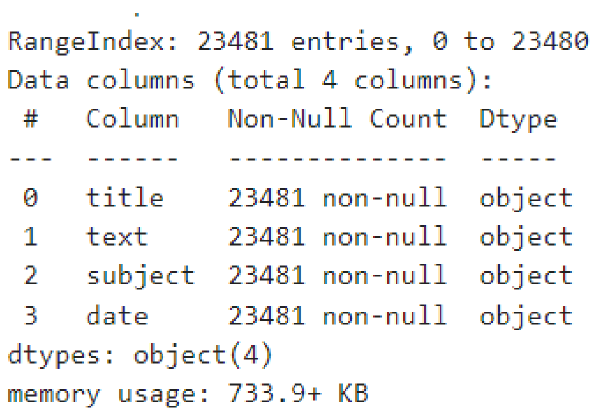
Two datasets are available, a mix of the two. There are 44898 news stories total in the csv file, which is a sizable quantity. While the true dataset only comprises 21417, the fraudulent dataset has 23481.

The dataset contains the following attributes:

The following elements are included in a news article:

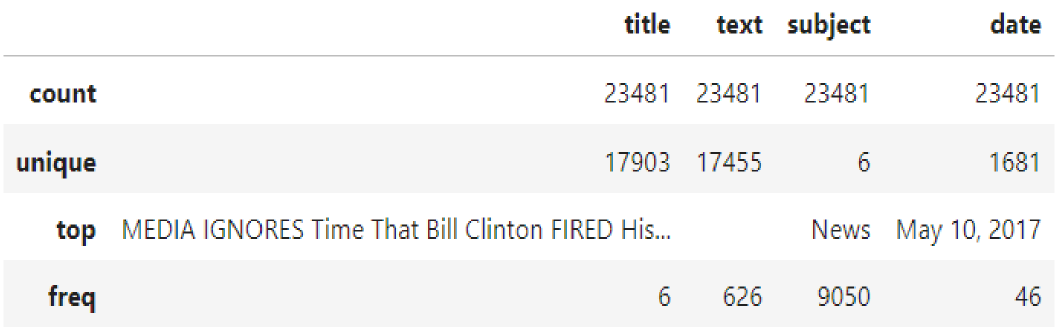
- Id: Special ID for News Article;
- title;
- text;
- Subject ▪ It describes the topic of the news;
- Date: It provides news's publication date.
- The conclusion that the information might not be trustworthy.
    - Fake : Untrustworthy or False News
    - true : Reliable or Accurate News

First of all, the dataset is quite balanced, as we have shown. There are 21417 accurate news items and 23481 false news pieces in it. This is a beneficial feature of the dataset.



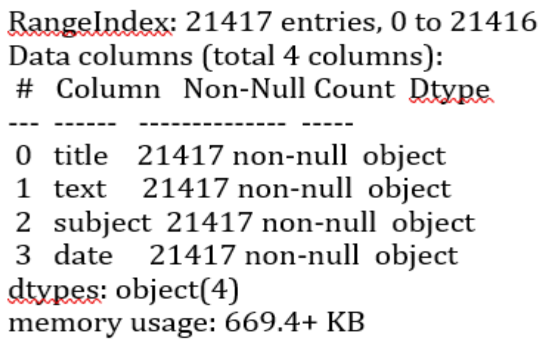
```
RangeIndex: 23481 entries, 0 to 23480
Data columns (total 4 columns):
 #   Column   Non-Null Count  Dtype 
---  ------   --------------  ----- 
 0   title    23481 non-null  object
 1   text     23481 non-null  object
 2   subject  23481 non-null  object
 3   date     23481 non-null  object
dtypes: object(4)
memory usage: 733.9+ KB
```

Fig 3. Datatype and memory usage



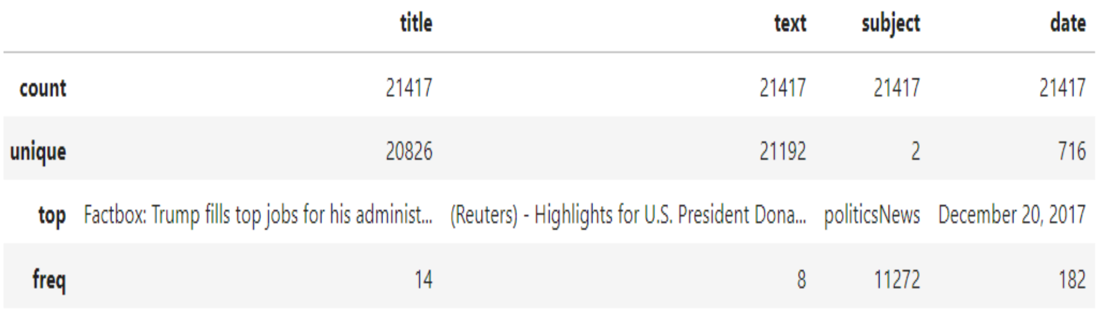
	title	text	subject	date
count	23481	23481	23481	23481
unique	17903	17455	6	1681
top	MEDIA IGNORES Time That Bill Clinton FIRED His...		News	May 10, 2017
freq	6	626	9050	46

Fig 4. Statistical values



```
RangeIndex: 21417 entries, 0 to 21416
Data columns (total 4 columns):
 #   Column   Non-Null Count  Dtype 
---  ------   --------------  ----- 
 0   title    21417 non-null  object
 1   text     21417 non-null  object
 2   subject  21417 non-null  object
 3   date     21417 non-null  object
dtypes: object(4)
memory usage: 669.4+ KB
```

Fig 5. Datatype and memory usage



	title	text	subject	date
count	21417	21417	21417	21417
unique	20826	21192	2	716
top	Factbox: Trump fills top jobs for his administ...	(Reuters) - Highlights for U.S. President Dona...	politicsNews	December 20, 2017
freq	14	8	11272	182

Fig 6. Statistical values

Statistical representation of the fake news dataset is shown in Fig 3. and Fig 4. same as for the statistical representation of the true news dataset in Fig 5. and Fig 6.

4.3.Data Pre-Processing

The dataset has undergone some processing, in this process we have done these steps –

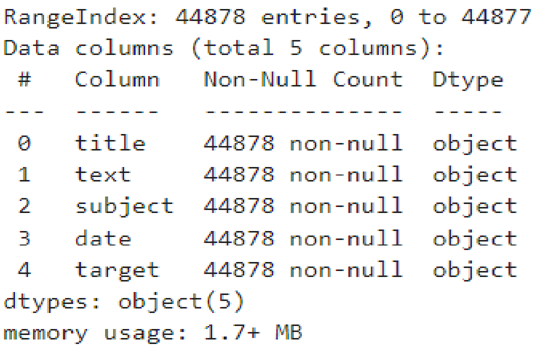
- removed 10 rows from both the true.csv and fake.csv file for the manual testing.

```
data_fake_manual_testing = data_fake.tail(10)
for i in range(23480, 23470, -1):
    data_fake.drop([i], axis = 0, inplace = True)


data_true_manual_testing = data_true.tail(10)
for i in range(21416, 21406, -1):
    data_fake.drop([i], axis = 0, inplace = True)



data_fake.shape, data_true.shape

((23461, 5), (21417, 5))
```

- then merged both the datasets and shuffled them for further.

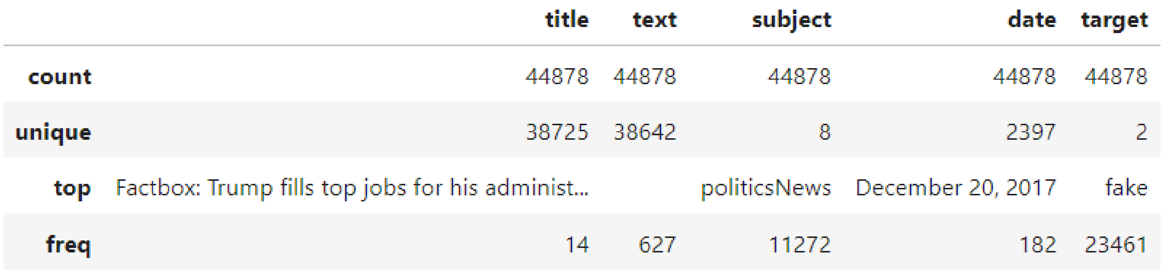
```
# concatenate dataframes

data = pd.concat([data_fake,data_true]).reset_index (drop = True)

data.shape

(44878, 5)


# shuffle the data

from sklearn.utils import shuffle

data = shuffle(data)

data = data.reset_index(drop = True)
```

Statastical representation of the shuffled data is shown in Fig 7. and fig 8.



```
RangeIndex: 44878 entries, 0 to 44877
Data columns (total 5 columns):
 #   Column   Non-Null Count  Dtype 
---  ------   --------------  ----- 
 0   title    44878 non-null  object
 1   text     44878 non-null  object
 2   subject  44878 non-null  object
 3   date     44878 non-null  object
 4   target   44878 non-null  object
dtypes: object(5)
memory usage: 1.7+ MB
```

Fig 7. Datatype and memory usage



	title	text	subject	date	target
count	44878	44878	44878	44878	44878
unique	38725	38642	8	2397	2
top	Factbox: Trump fills top jobs for his administ...		politicsNews	December 20, 2017	fake
freq	14	627	11272	182	23461

Fig 8. Statistical values

- We have cleaned the text by eliminating punctuation and stopwords or changed the text accordingly for the evaluation.

```
def wordopt(text):
    text = text.lower()
   text = re.sub('\[.*?\]', ' ', text)
    text = re.sub("\\W", " ", text)
    text = re.sub('https?://\S+', ' ', text)
    text = re.sub('<.*?>+', ' ', text)
    text = re.sub('[%s]' % re.escape(string.punctuation), ' ', text)
    text = re.sub('\n', ' ', text)
    text = re.sub('\w*\d\w*', ' ', text)
    return text


data['text'] = data['text'].apply( wordopt)
```

- Dropped unnecessary columns such as date and title.

```
# removing the date
data.drop(["date"],axis=1,inplace=True)
```

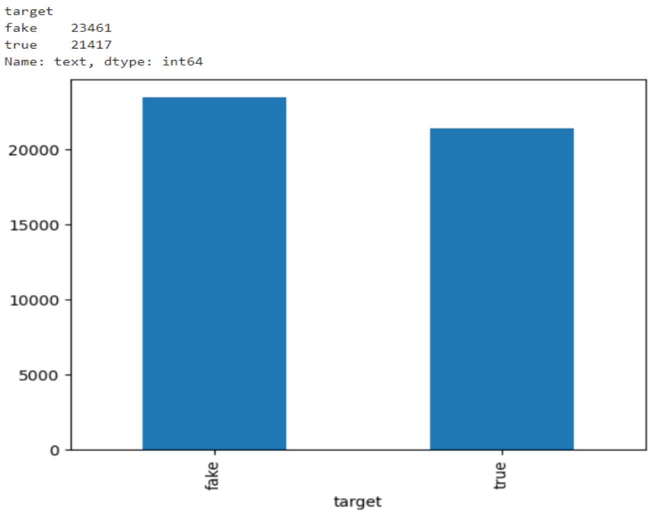
```
# removing the title

data.drop(["title"],axis=1,inplace=True)
```

4.4.Data Visualization

We have done basic visualisation task for the data-

- Created a graph representation for the fake and true datasets as shown in Fig 9.

```
print(data.groupby(['target'])['text'].count())

data.groupby(['target'])['text'].count().plot(kind = "bar")

plt.show()
```



```
target
fake    23461
true    21417
Name: text, dtype: int64
```

Fig 9. Comparison of Fake and Real news

- We also examined the news story subjects. Before stop words are removed, the topics of the news stories are not at all clear. As a result, removing stop words makes it simpler to comprehend the news reports' themes. In Fig 10., we plotted the frequencies of subject of the news:

```
print(data.groupby(['subject'])['text'].count())
data.groupby(['subject'])['text'].count().plot(kind = "bar")
plt.show()
```

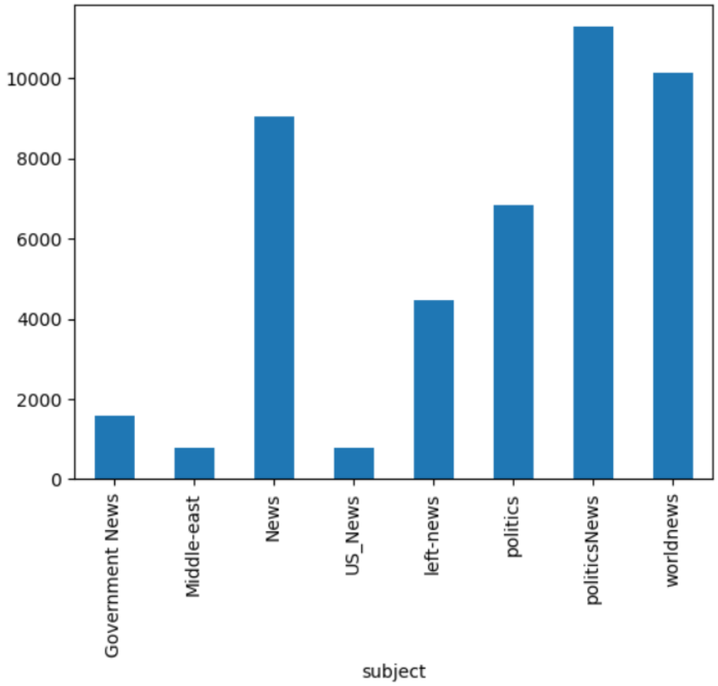


Fig 10. Frequency of subject of the news

- Created a Word Cloud that is a data visualization technique used for representing text data in which the size of each word indicates its frequency or importance. Significant textual data points can be highlighted using a word cloud. Word clouds are widely used for analysing data from social network websites.

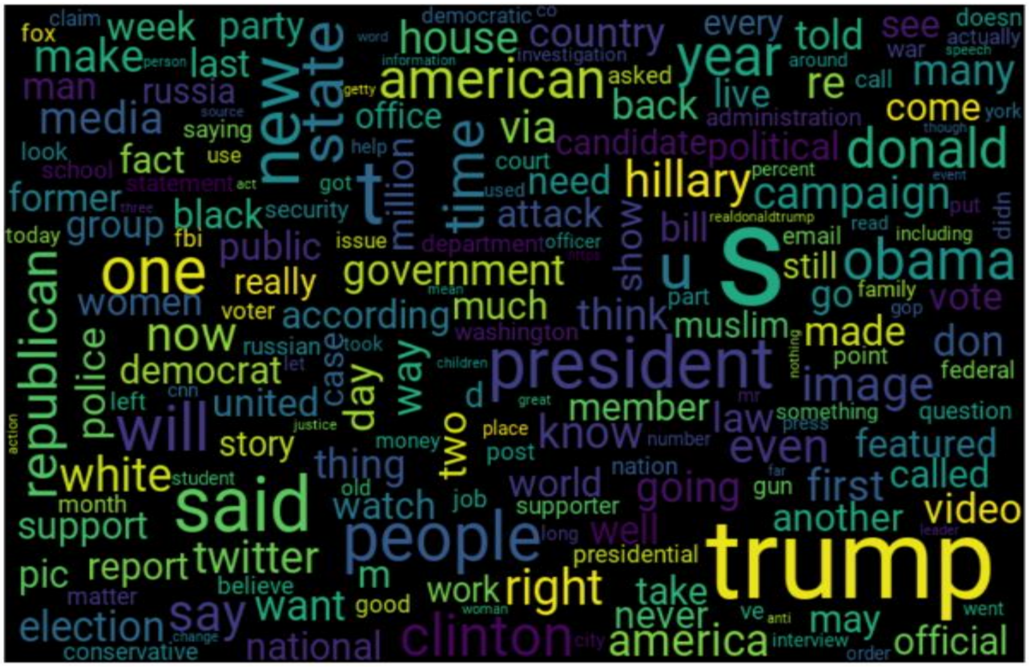


Fig 11. Word Cloud representation

Fig 11. shows the representation for the fake news dataset and Fig 12. shows the representation for the true news dataset.

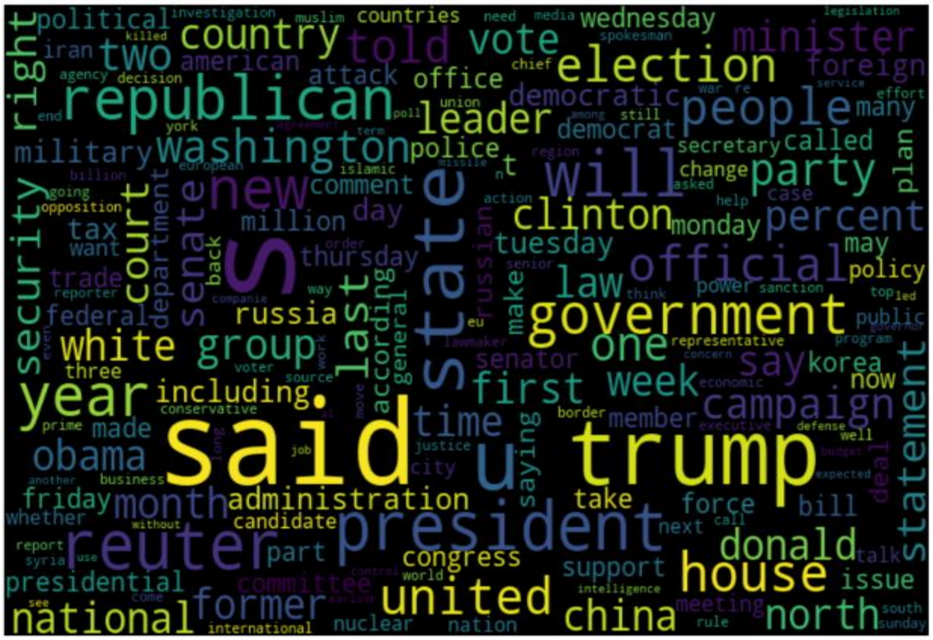


Fig 12. Word Cloud representation

4.5. Model Building

Here, we selected for training and testing dataset for the model fitting and used the Tf-idf Vectorizer in the Sklearn package to turn the text data into numerical data.

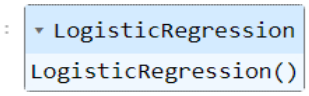
```
from sklearn.feature_extraction.text import TfidfVectorizer


vectorization = TfidfVectorizer()

xv_train = vectorization.fit_transform(x_train)

xv_test = vectorization.transform(x_test)
```

4.6. Model Fitting

Utilise the data that has been modified by Tf-idf Vectorizer to train a variety of models, including Logistic Regression, Decision Tree, Gradient Bosting, Random Forest, etc. We have Fitted the models using the training set and used the testing set to predict the news article labels, then determined each model's accuracy score using the actual and projected labels.

- **LOGISTIC REGRESSION:** The advantages of logistic regression include probability modelling, the capacity to depend on features, and the flexibility to update the model. However, for higher accuracy, logistic regression requires a big data set, but Naive Bayes may function with small datasets as well.

```
from sklearn.linear_model import LogisticRegression
LR = LogisticRegression()
LR.fit(xv_train, y_train)
```

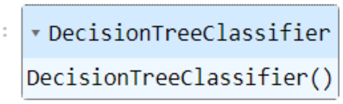


```
▾ LogisticRegression
LogisticRegression()
```

```
pred_lr = LR.predict(xv_test)
LR.score(xv_test, y_test)
0.9849598930481284
```

- **DECISION TREE:** A decision tree is made up of decision nodes that start at the top and work their way down. Dependent characteristics, no need for linear class separation, fast management of outliers, and intuitive decision tree interpretation are all advantages of employing a decision tree. When there are a significant number of sparse features, however, a decision tree will overfit and perform poorly on the testing data.

```
from sklearn.linear_model import DecisiontreeClassifier
DT = Decisiontreeclassifier()
DT.fit(xv_train, y_train)
```

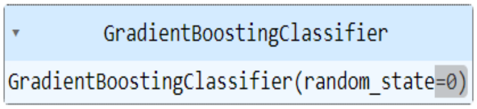


```
▾ DecisionTreeClassifier
DecisionTreeClassifier()
```

```
pred_dt = DT.predict(xv_test)
DT.score(xv_test, y_test)
0.9948752228163993
```

- **GRADIENT BOOSTING:** Gradient boosting creates an ensemble of weak prediction models, usually decision trees, as a prediction model. The resulting technique is called gradient boosted trees when a decision tree is a weak learner, and it usually outperforms random forest. It constructs the model in the same stage-by-stage manner as other boosting approaches, but it broadens the scope by allowing optimization of any differentiable loss function.

```
from sklearn.linear_model import GradientBoostingClassifier
GB = GradientBoostingClassifier(random state = 0)
GB.fit(xv_train, y_train)
```

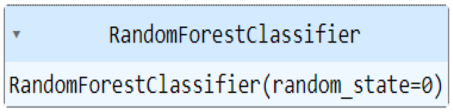


```
▾ GradientBoostingClassifier
GradientBoostingClassifier(random_state=0)
```

```
pred_gb = GB.predict(xv_test)
GB.score(xv_test, y_test)
0.9944295900178253
```

- **RANDOM FOREST:** Many decision trees are built by the random forest algorithm. Utilizing a subset of features, each decision tree is created. Each decision tree produces one class and eventually bootstraps the votes to obtain better accuracy from the Random Forest technique. A tree-shaped pattern is used to describe the plan of action in a decision tree. At any node, a decision will be made.

```
from sklearn.linear_model import RandomForestClassifier
RF = RandomForestClassifier(random state = 0)
RF.fit(xv_train, y_train)
```



```
▾ RandomForestClassifier
RandomForestClassifier(random_state=0)
```

```
pred_rf = RF.predict(xv_test)
RF.score(xv_test, y_test)
0.9862967914438503
```

4.7. Model Analysis

It is important to check the performance of multiple different machine learning algorithms consistently. The key to a fair comparison of machine learning algorithms is ensuring that each algorithm is evaluated in the same way on the same data. It's crucial to assess the false news detection model's performance on the testing set after we've trained it. By assessing its accuracy, precision, recall, and F1 score, we may do this.

Here are the classification report of the models that we used in our project for the better understanding, these includes -

- Accuracy: How often a data point is correctly classified by the algorithm.
- Precision: The number of accurately predicted positive observations divided by the total number of predicted positive observations.
- Recall: The percentage of accurately anticipated positive observations to the total number of observations in a class is known as recall.
- F1 Score: The F1 Score is the weighted average of Precision and Recall.

- **LOGISTIC REGRESSION**

```
print(classification_report(y_test, pred_lr))

                      precision    recall  f1-score   support
fake                       0.99      0.98      0.99      4662
true                       0.98      0.98      0.98      4314
accuracy                                       0.98      8976
macro avg                  0.98      0.98      0.98      8976
weighted avg               0.98      0.98      0.98      8976
```

- **DECISION TREE**

```
print(classification_report(y_test, pred_dt))
                      precision    recall  f1-score   support
fake                       0.99      1.00      1.00      4662
true                       1.00      0.99      0.99      4314
accuracy                                        0.99      8976
macro avg                0.99      0.99      0.99      8976
weighted avg           0.99      0.99      0.99      8976
```

- **GRADIENT BOOSTING**

```
print(classification_report(y_test, pred_gb))
                      precision    recall  f1-score   support
fake                       1.00      0.99      0.99      4662
true                       0.99      1.00      0.99      4314
accuracy                                        0.99      8976
macro avg                0.99      0.99      0.99      8976
weighted avg           0.99      0.99      0.99      8976
```

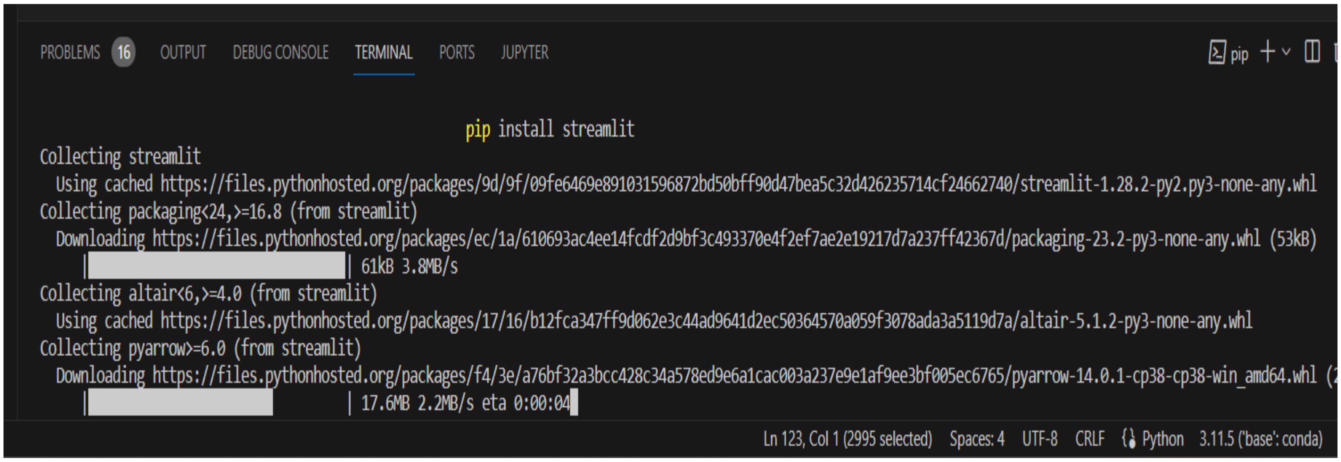
- **RANDOM FOREST**

```
print(classification_report(y_test, pred_rf))
                      precision    recall  f1-score   support
fake                       0.99      0.99      0.99      4662
true                       0.99      0.98      0.99      4314
accuracy                                        0.99      8976
macro avg                0.99      0.99      0.99      8976
weighted avg           0.99      0.99      0.99      8976
```

4.8. Representing the Output in Web Browser

- To create a web interface for the task of fake news detection. We have used streamlit library in Python to build an application for the machine learning model to detect fake news in real-time. Firstly, we installed it on your system using the pip command:

pip install streamlit



```
                                        pip install streamlit
Collecting streamlit
  Using cached https://files.pythonhosted.org/packages/9d/9f/09fe6469e891031596872bd50bff90d47bea5c32d426235714cf24662740/streamlit-1.28.2-py2.py3-none-any.whl
Collecting packaging<24,>=16.8 (from streamlit)
  Downloading https://files.pythonhosted.org/packages/ec/1a/610693ac4ee14fcdf2d9bf3c493370e4f2ef7ae2e19217d7a237ff42367d/packaging-23.2-py3-none-any.whl (53kB)
     |████████████████████████████████| 61kB 3.8MB/s
Collecting altair<6,>=4.0 (from streamlit)
  Using cached https://files.pythonhosted.org/packages/17/16/b12fca347ff9d062e3c44ad9641d2ec50364570a059f3078ada3a5119d7a/altair-5.1.2-py3-none-any.whl
Collecting pyarrow>=6.0 (from streamlit)
  Downloading https://files.pythonhosted.org/packages/f4/3e/a76bf32a3bcc428c34a578ed9e6a1cac003a237e9e1af9ee3bf005ec6765/pyarrow-14.0.1-cp38-cp38-win_amd64.whl (2
     |███████████████████████         | 17.6MB 2.2MB/s eta 0:00:04
```

Fig 13. Terminal from app.py

- We cannot run this code the same way you run your other Python programs. As we are using the streamlit library here, so we need to write a command mentioned below in our command prompt or terminal to run this code:

streamlit run filename.py

Once this command executes, it will open a link on your default web browser that will display your output as a web interface for fake news detection, as shown below.

- Now we can give input as a news article and this application will show you if the news article you gave as input is fake or real. The interface will look like the Fig 14.

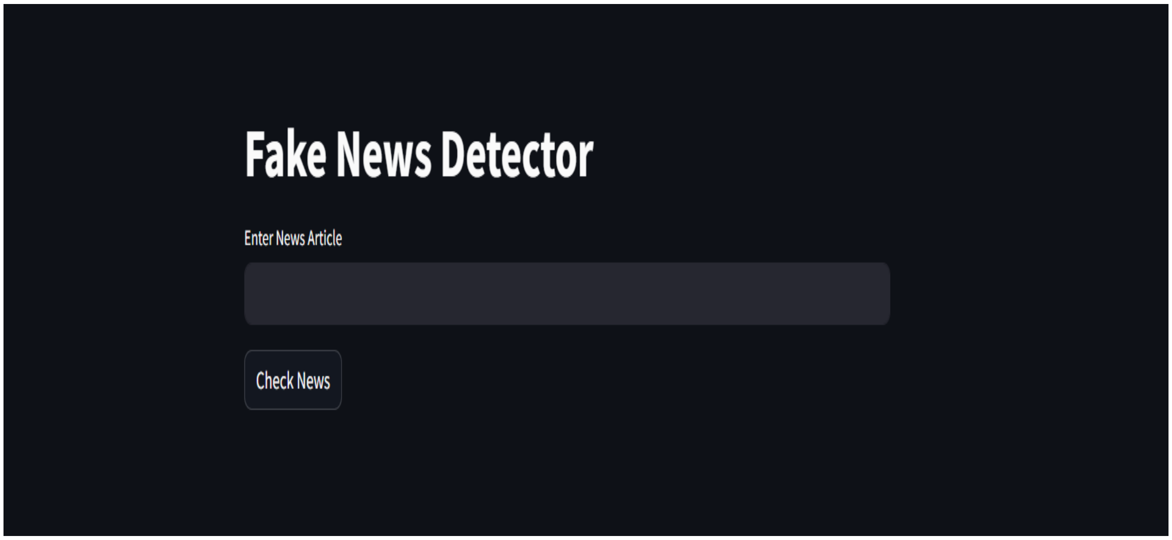


Fig 14. Web interface

Chapter 5: Output

This project can understand every sentence of the news articles from the dataset using NLP, then classify the news into real and fake news, so we can search for the news sentences in your own English. It can understand the content and give the result if it is fake or real. We have to update the dataset regularly to get the real-time experience. If we collaborate this model with any of the leading authorised news websites, we can eradicate fake news completely.

- The first look of the interface shown in Fig 15. –

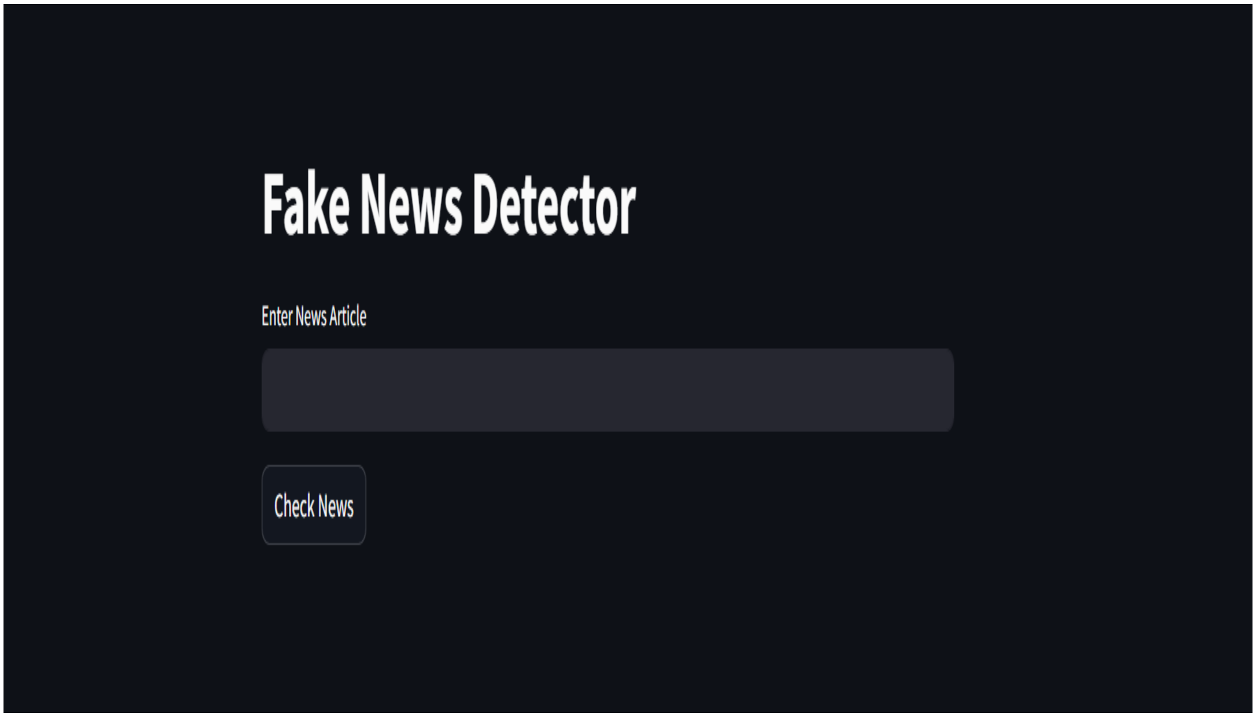
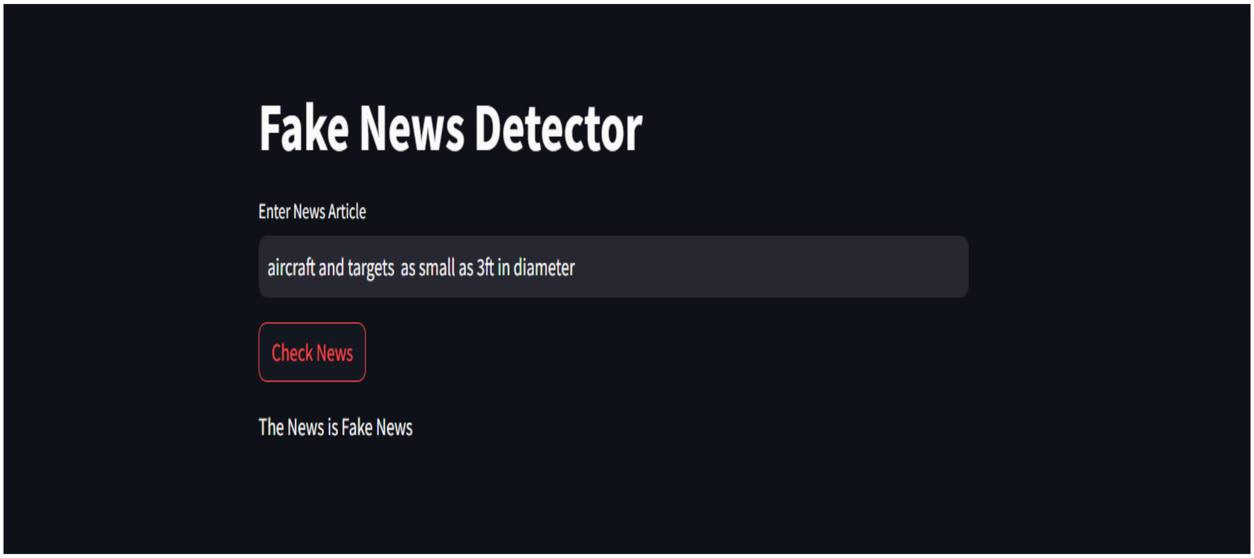


Fig 15. Web interface

- When we tested a fake news article, it will give the output as **"the news is fake news"** as shown in Fig 16. –



Fake News Detector

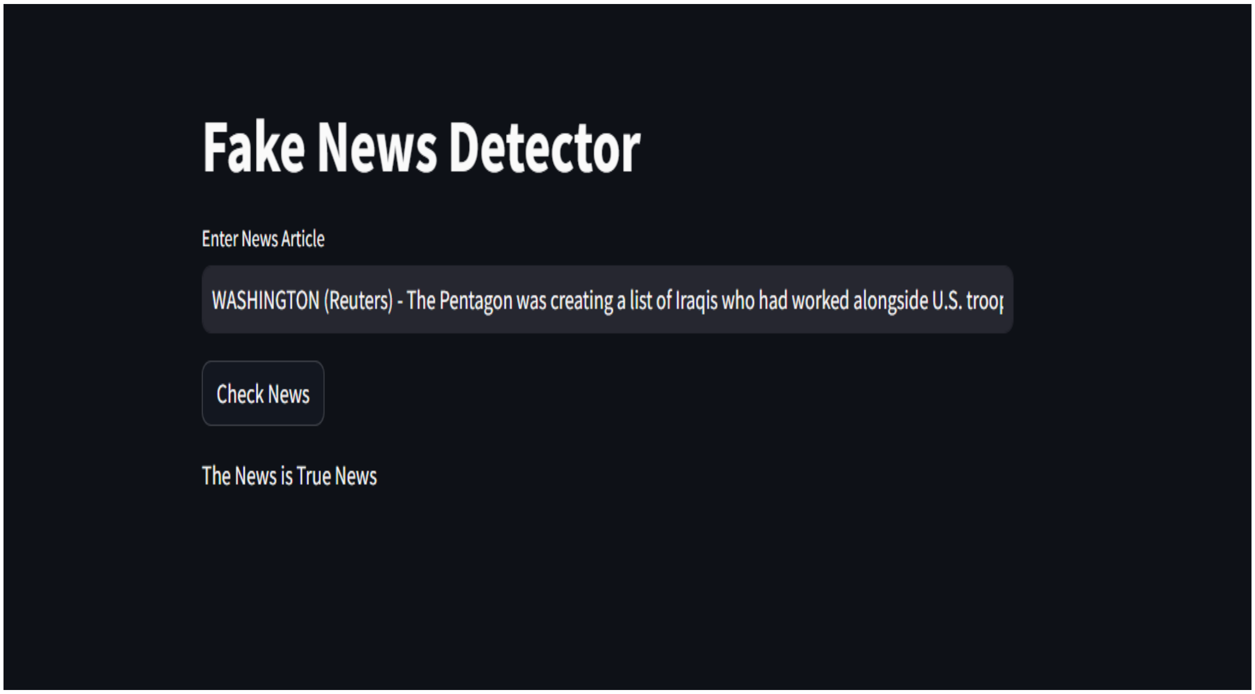
Enter News Article

aircraft and targets as small as 3ft in diameter

The News is Fake News

Fig 16. Testing of the interface

- When we tested a true news article, it will give the output as **"the news is true news"** as shown in Fig 16. –



Fake News Detector

Enter News Article

WASHINGTON (Reuters) - The Pentagon was creating a list of Iraqis who had worked alongside U.S. troo

The News is True News

Fig 17. Testing of the interface

Chapter6:

CONCLUSION

This project comes up with the applications of NLP (Natural Language Processing) techniques for detecting the 'fake news', that is misleading news stories that comes from the non-reputable sources. The result confirms that the prediction of whether news is real or fake is correct. The main scope of the project is to decrease the fake news which is spread among the people. The accuracy of this project depends upon the dataset we are using. The algorithm we are using, To make it more accurate. Considering the accuracy scores, we were able to establish for the various models, it appears that all of the models are doing a good job of identifying false news items. The Logistic Regression, Decision Tree, Gradient Boosting classifiers and Random Forest. All things considered, the results of the algorithm suggest that a range of classifiers may be used with equal success rates and that machine learning techniques may be extremely successful in spotting bogus news. It's important to keep in mind that accuracy is only one measure and that the models should be evaluated using multiple metrics including precision, recall, and F1-score in addition to factors like interpretability, scalability, and processing requirements. Investigating different feature extraction and selection methods, classifier types, and ensemble approaches may also be useful to see whether even better results may be produced. We utilised the datasets real and fake, each of which had 21417 and 23481 entries, respectively. We converted text into a numerical model using TF-IDF Vectorizer and utilised the following models: Accuracy of **98%** for Logistic Regression, **99%** for Decision Tree, **99%** for Gradient Boosting and accuracy of **99%** for the Random Forest Classifier. The interface is also giving the right values according to the news article as **“true news of fake news”.**

FUTURE WORK

The project is working well. The future work can include enhancing the model by using a better fitting algorithms or approaches as that we used cannot be very accurate at all times.

Future efforts to identify bogus news may go in the following directions:

- Including more varied and subtle aspects
- Creating more interpretable models
- Combining information from other sources
- Adapting to shifting strategies

The project is scaled for a very limited user base, so it can only detect. It can be scaled up for a greater number of users and thus can be used in other domains. Better technology for night vision can be implemented as it is not appropriate for darker regions.

REFERNCES

For making this project we have used many research papers. The links are below specified:

1. Shu, K.; Sliva, A.; Wang, S.; Tang, J.; Liu, H. Fake news detection on social media: A data mining perspective.
   ACM SIGKDD Explor. Newsl. **2017**, 19, 22–36. [CrossRef]
2. Tembhurne, J.V.; Almin, M.M.; Diwan, T. Mc-DNN: Fake News Detection Using Multi-Channel Deep Neural Networks.
   Int. J. Semant. Web Inf. Syst. IJSWIS **2022**, 18, 1–20. [CrossRef]
3. Awan, M.J.; Yasin, A.; Nobanee, H.; Ali, A.A.; Shahzad, Z.; Nabeel, M.; Zain, A.M.; Shahzad, H.M.F. Fake news data exploration and analytics.
   Electronics **2021**, 10, 2326. [CrossRef]
4. Sharma, D.K.; Garg, S. IFND: A benchmark dataset for fake news detection.
   Complex Intell. Syst. **2021**, 1–21. [CrossRef]
5. Ghayoomi, M.; Mousavian, M. Deep transfer learning for COVID-19 fake news detection in Persian.
   Expert Syst. **2022**, 39, e13008.[CrossRef]
6. Do, T.H.; Berneman, M.; Patro, J.; Bekoulis, G.; Deligiannis, N. Context-aware deep Markov random fields for fake news detection.
   IEEE Access **2021**, 9, 130042–130054. [CrossRef]
7. Kumari, R.; Ashok, N.; Ghosal, T.; Ekbal, A. What the fake? Probing misinformation detection standing on the shoulder of novelty and emotion.
   Inf. Process. Manag. **2022**, 59, 102740. [CrossRef]
8. Ying, L.; Yu, H.; Wang, J.; Ji, Y.; Qian, S. Fake news detection via multi-modal topic memory network.
   IEEE Access **2021**, 9, 132818–132829. [CrossRef]
9. Vicario, M.D.; Quattrociocchi, W.; Scala, A.; Zollo, F. Polarization and fake news: Early warning of potential misinformation targets.
   ACM Trans. Web TWEB **2019**, 13, 1–22. [CrossRef]
10. Stitini, O.; Kaloun, S.; Bencharef, O. Towards the Detection of Fake News on Social Networks Contributing to the Improvement of Trust and Transparency in Recommendation Systems: Trends and Challenges.
    Information **2022**, 13, 128. [CrossRef]

11. Fayaz, M.; Khan, A.; Bilal, M.; Khan, S.U. Machine learning for fake news classification with optimal feature selection.
Soft Comput. **2022**, 1–9. [CrossRef]

12. Islam, N.; Shaikh, A.; Qaiser, A.; Asiri, Y.; Almakdi, S.; Sulaiman, A.; Moazzam, V.; Babar, S.A. Ternion: An Autonomous Model for Fake News Detection.
Appl. Sci. **2021**, 11, 9292. [CrossRef]

13. Hansrajh, A.; Adeliyi, T.T.; Wing, J. Detection of online fake news using blending ensemble learning.
Sci. Program. **2021**, 2021, 3434458. [CrossRef]

14. Elsaeed, E.; Ouda, O.; Elmogy, M.M.; Atwan, A.; El-Daydamony, E. Detecting Fake News in Social Media Using Voting Classifier.
IEEE Access **2021**, 9, 161909–161925. [CrossRef]

15. Verma, P.K.; Agrawal, P.; Amorim, I.; Prodan, R. WELFake: Word embedding over linguistic features for fake news detection.
IEEE Trans. Comput. Soc. Syst. **2021**, 8, 881–893. [CrossRef]

16. Biradar, S.; Saumya, S.; Chauhan, A. Combating the infodemic: COVID-19 induced fake news recognition in social media networks. Complex Intell. Syst. **2022**, 1–13. [CrossRef]

17. Kanagavalli, N.; Priya, S.B. Social Networks Fake Account and Fake News Identification with Reliable Deep Learning.Intell.
Autom. Soft Comput. **2022**, 33, 191–205. [CrossRef]

18. Elhadad, M.K.; Li, K.F.; Gebali, F. Detecting misleading information on COVID-19.
IEEE Access **2020**, 8, 165201–165215. [CrossRef

19. Galli, A.; Masciari, E.; Moscato, V.; Sperlí, G. A comprehensive Benchmark for fake news detection.
J. Intell. Inf. Syst. **2022**, 59, 237–261. [CrossRef] [PubMed]

20. Sadeghi, F.; Bidgoly, A.J.; Amirkhani, H. Fake news detection on social media using a natural language inference approach.
Multimed. Tools Appl. **2022**, 81, 33801–33821. [CrossRef]

21. Fouad, K.M.; Sabbeh, S.F.; Medhat, W. Arabic fake news detection using deep learning. CMC-Comput.
Mater. Contin. **2022**, 71, 3647–3665. [CrossRef]

22. Nassif, A.B.; Elnagar, A.; Elgendy, O.; Afadar, Y. Arabic fake news detection based on deep contextualized embedding models.
Neural Comput.
Appl. **2022**, 34, 16019–16032. [CrossRef] [PubMed]

23. Lee, J.-W.; Kim, J.-H. Fake Sentence Detection Based on Transfer Learning: Applying to Korean COVID-19 Fake News.
Appl. Sci. **2022**, 12, 6402. [CrossRef]

24. Truică, C.-O.; Apostol, E.-S. MisRoBÆRTa: Transformers versus Misinformation.
Mathematics **2022**, 10, 569. [CrossRef]

25. Jang, Y.; Park, C.-H.; Lee, D.-G.; Seo, Y.-S. Fake News Detection on Social Media: A Temporal-Based Approach. CMC-Comput.Mater. Contin. **2021**, 69, 3563–3579. [CrossRef]

26. Mikolajewicz, N.; Komarova, S.V. Meta-analytic methodology for basic research: A practical guide.
Front. Physiol. **2019**, 10,203[CrossRef]

27. Collins, B.; Hoang, D.T.; Nguyen, N.T.; Hwang, D. Trends in combating fake news on social media–a survey.
J. Inf. Telecommun. **2021**, 5, 247-266. [CrossRef]

28. Chora´s, M.; Demestichas, K.; Giełczyk, A.; Herrero, Á.; Ksieniewicz, P.; Remoundou, K.; Urda, D.; Wo´zniak, M. Advanced Machine Learning techniques for fake news (online disinformation) detection: A systematic mapping study.
Appl. Soft Comput. **2021**, 101, 107050. [CrossRef]

29. Varlamis, I.; Michail, D.; Glykou, F.; Tsantilas, P. A Survey on the Use of Graph Convolutional Networks for Combating FakeNews. Future Internet **2022**, 14, 70. [CrossRef]

30. Shahid, W.; Jamshidi, B.; Hakak, S.; Isah, H.; Khan, W.Z.; Khan, M.K.; Choo, K.-K.R. Detecting and Mitigating the Dissemination of Fake News: Challenges and Future Research Opportunities.

IEEE Trans. Comput. Soc. Syst. **2022**, 1–14. [CrossRef]

31. Khan, S.; Hakak, S.; Deepa, N.; Dev, K.; Trelova, S. Detecting COVID-19 related Fake News using feature extraction.
Front. Public Health **2022**, 1967. [CrossRef]

32. Lozano, M.G.; Brynielsson, J.; Franke, U.; Rosell, M.; Tjörnhammar, E.; Varga, S.; Vlassov, V. Veracity assessment of online data.Decis.
Support Syst. **2020**, 129, 113132. [CrossRef]

33. https://aws.amazon.com/what-is/nlp/#:~:text=Natural%20language%20processing%20(NLP)%20is,manipulate%2C%20and%20comprehend%20human%20language.

34. J. D'Souza, "An Introduction to Bag-of-Words in NLP," 03 04 2018. [Online]. Available:https://medium.com/greyatom/an-introduction-tobag-of-words-in-nlp-ac 967d43b428.

35. A. Douglas, “News consumption and the new electronic media,” *The International Journal of Press/Politics*, vol. 11, no. 1, pp. 29–52, 2006.
View at: Publisher Site | Google Scholar

36. N. K. Conroy, V. L. Rubin, and Y. Chen, “Automatic deception detection: methods for finding fake news,” *Proceedings of the Association for Information Science and Technology*, vol. 52, no. 1, pp. 1–4, 2015.
View at: Publisher Site | Google Scholar

37. F. T. Asr and M. Taboada, “Misinfotext: a collection of news articles, with false and true labels,” 2019.
View at: Google Scholar

For making this project we have used many websites and research papers. The links are below specified:

- https://realpython.com/nltk-nlp-python/
- https://www.datacamp.com/tutorial/wordcloud-python
- https://www.simplilearn.com/tutorials/machine-learning-tutorial/how-to-create-a-fake-news-detection-system
- https://www.datacamp.com/tutorial/streamlit

We have also used so many other youtube channels and google to solve our errors.

We used Official python documentations to code in python

ANNEXURE I:

RESPONSIBILITY CHART

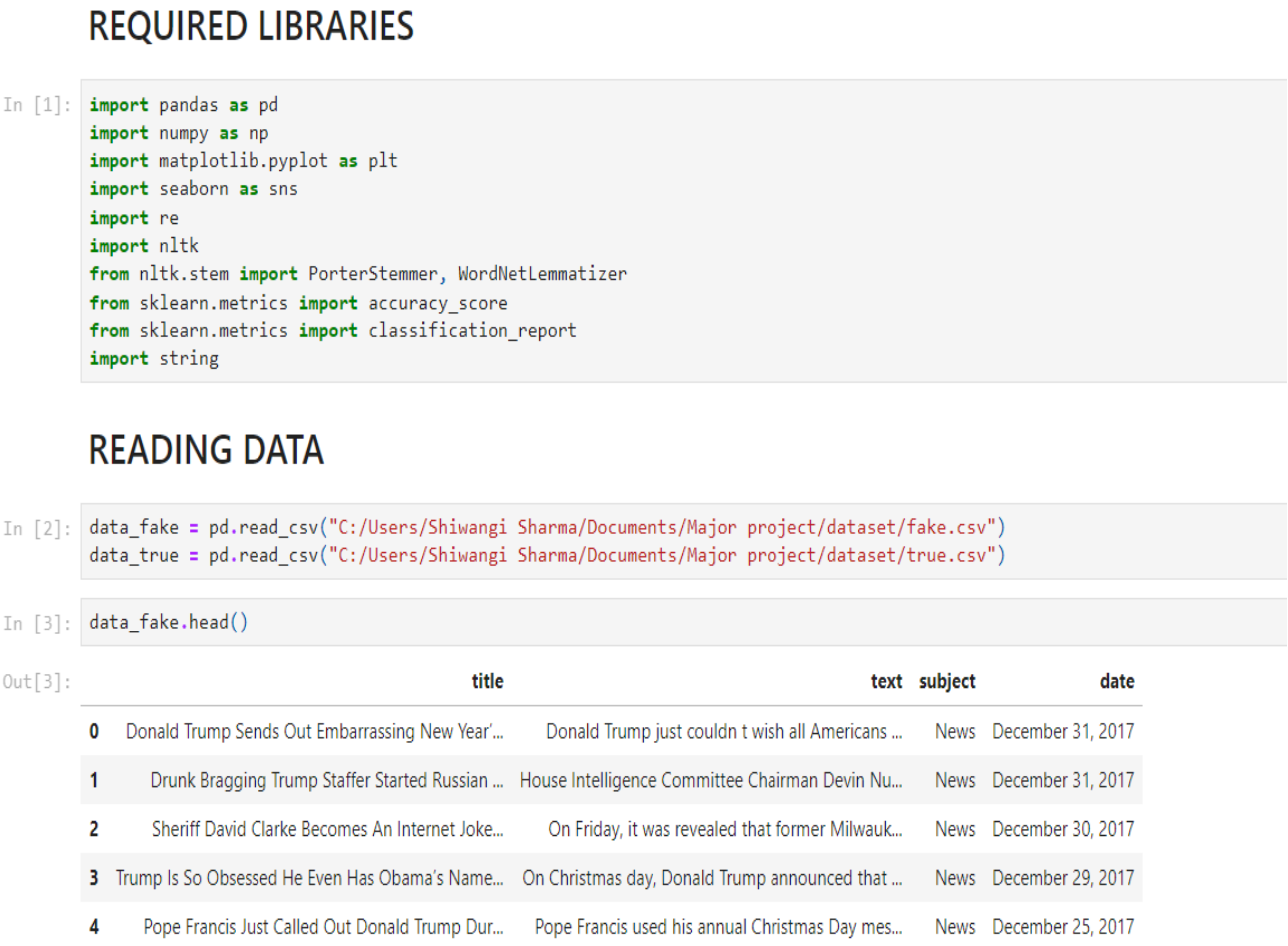
ROLL NUMBER	*NAME*	*RESPONSIBILITIES*
2201062040	**AAYUSH UJJWAL**	**ALL THE CODING PART – DATA COLLECTION, DATA PRE-PROCESSING, DATA VISUALIZATION, MODEL FITTING, WEB INTERFACE, POWER POINT WORK AND THE FINAL REPORT.**

Annexure II:

COMPLETE IMPLEMENTATION

The coding part in below with specific modules discussed above.

fake news detetction.py



REQUIRED LIBRARIES

```
In [1]: import pandas as pd
        import numpy as np
        import matplotlib.pyplot as plt
        import seaborn as sns
        import re
        import nltk
        from nltk.stem import PorterStemmer, WordNetLemmatizer
        from sklearn.metrics import accuracy_score
        from sklearn.metrics import classification_report
        import string
```

READING DATA

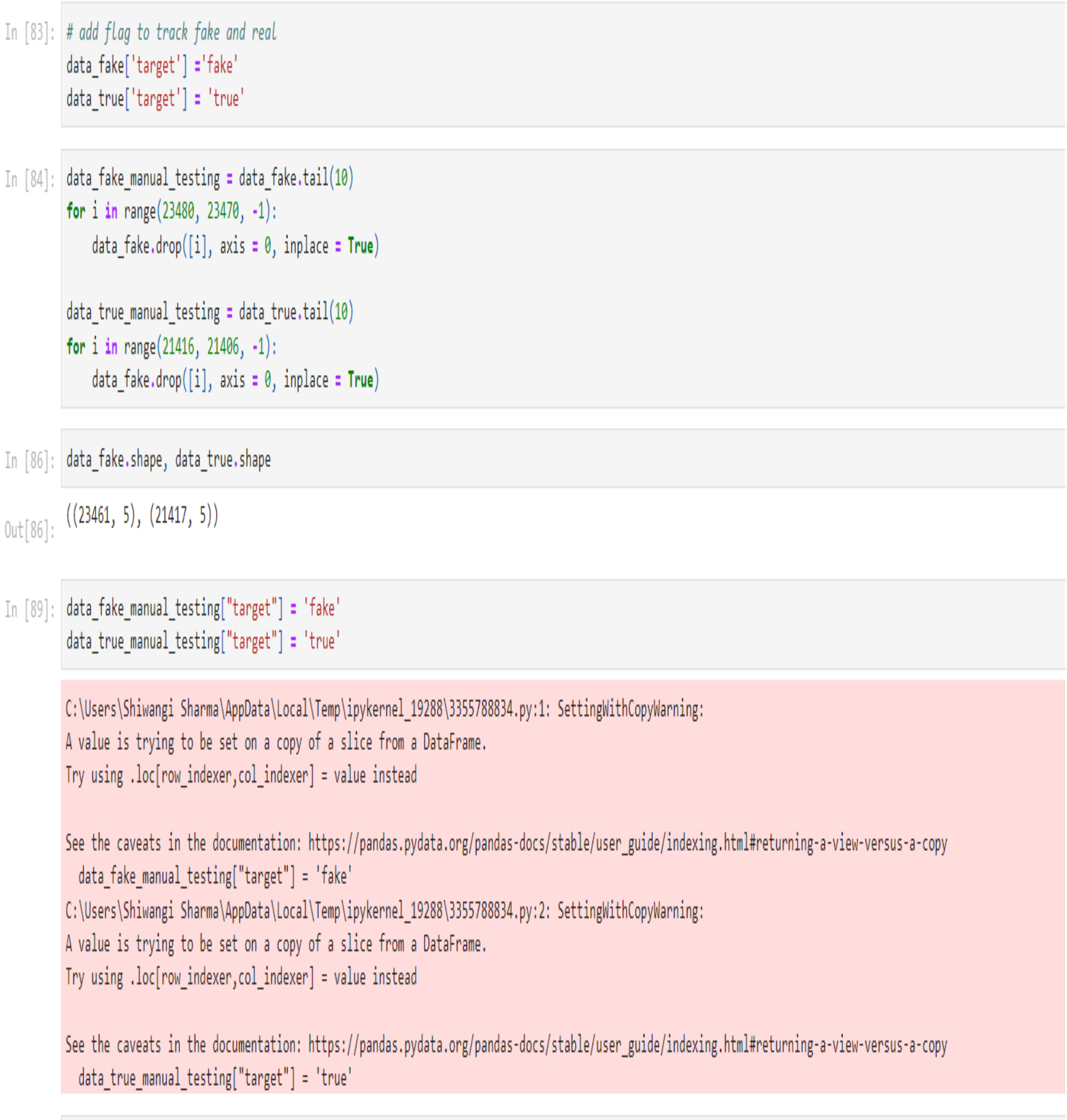
```
In [2]: data_fake = pd.read_csv("C:/Users/Shiwangi Sharma/Documents/Major project/dataset/fake.csv")
        data_true = pd.read_csv("C:/Users/Shiwangi Sharma/Documents/Major project/dataset/true.csv")
```

```
In [3]: data_fake.head()
```

Out[3]:

	title	text	subject	date
0	Donald Trump Sends Out Embarrassing New Year'...	Donald Trump just couldn t wish all Americans ...	News	December 31, 2017
1	Drunk Bragging Trump Staffer Started Russian ...	House Intelligence Committee Chairman Devin Nu...	News	December 31, 2017
2	Sheriff David Clarke Becomes An Internet Joke...	On Friday, it was revealed that former Milwauk...	News	December 30, 2017
3	Trump Is So Obsessed He Even Has Obama's Name...	On Christmas day, Donald Trump announced that ...	News	December 29, 2017
4	Pope Francis Just Called Out Donald Trump Dur...	Pope Francis used his annual Christmas Day mes...	News	December 25, 2017

DATA CLEANING & PREPROCESSING



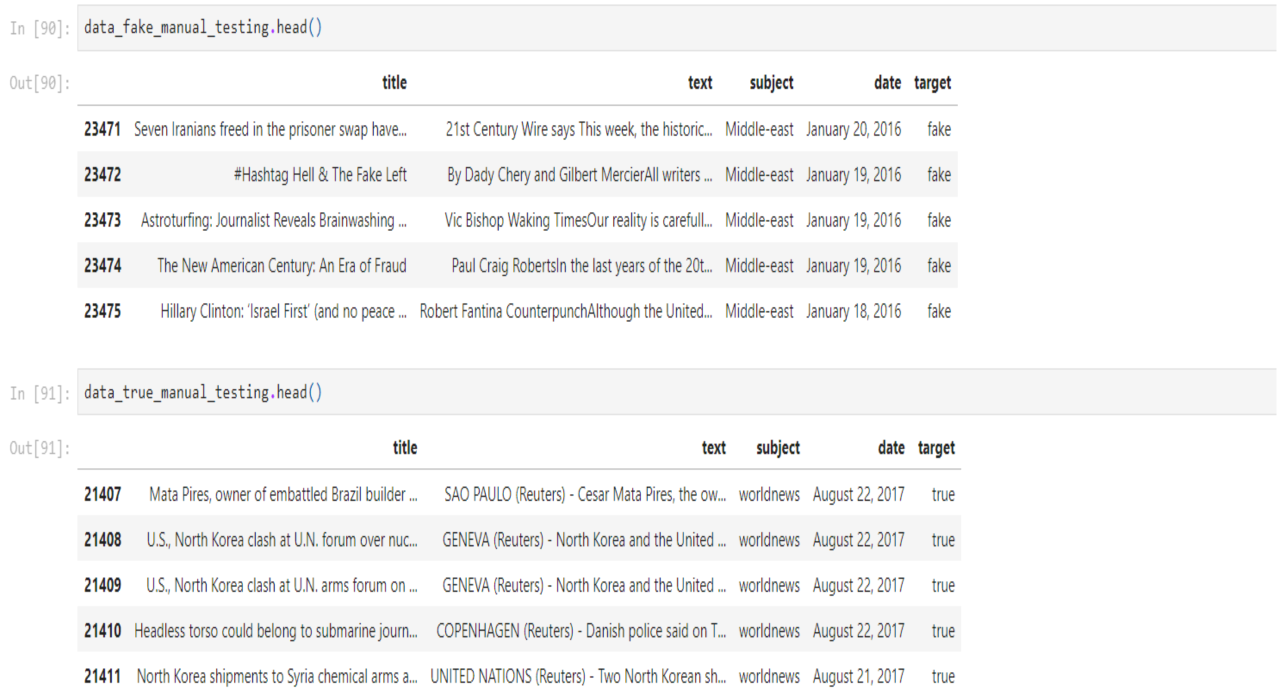
```
In [83]: # add flag to track fake and real
         data_fake['target'] ='fake'
         data_true['target'] = 'true'
```

```
In [84]: data_fake_manual_testing = data_fake.tail(10)
         for i in range(23480, 23470, -1):
             data_fake.drop([i], axis = 0, inplace = True)

         data_true_manual_testing = data_true.tail(10)
         for i in range(21416, 21406, -1):
             data_fake.drop([i], axis = 0, inplace = True)
```

```
In [86]: data_fake.shape, data_true.shape

Out[86]: ((23461, 5), (21417, 5))
```

```
In [89]: data_fake_manual_testing["target"] = 'fake'
         data_true_manual_testing["target"] = 'true'
```

```
C:\Users\Shiwangi Sharma\AppData\Local\Temp\ipykernel_19288\3355788834.py:1: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  data_fake_manual_testing["target"] = 'fake'
C:\Users\Shiwangi Sharma\AppData\Local\Temp\ipykernel_19288\3355788834.py:2: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  data_true_manual_testing["target"] = 'true'
```



```
In [90]: data_fake_manual_testing.head()
```

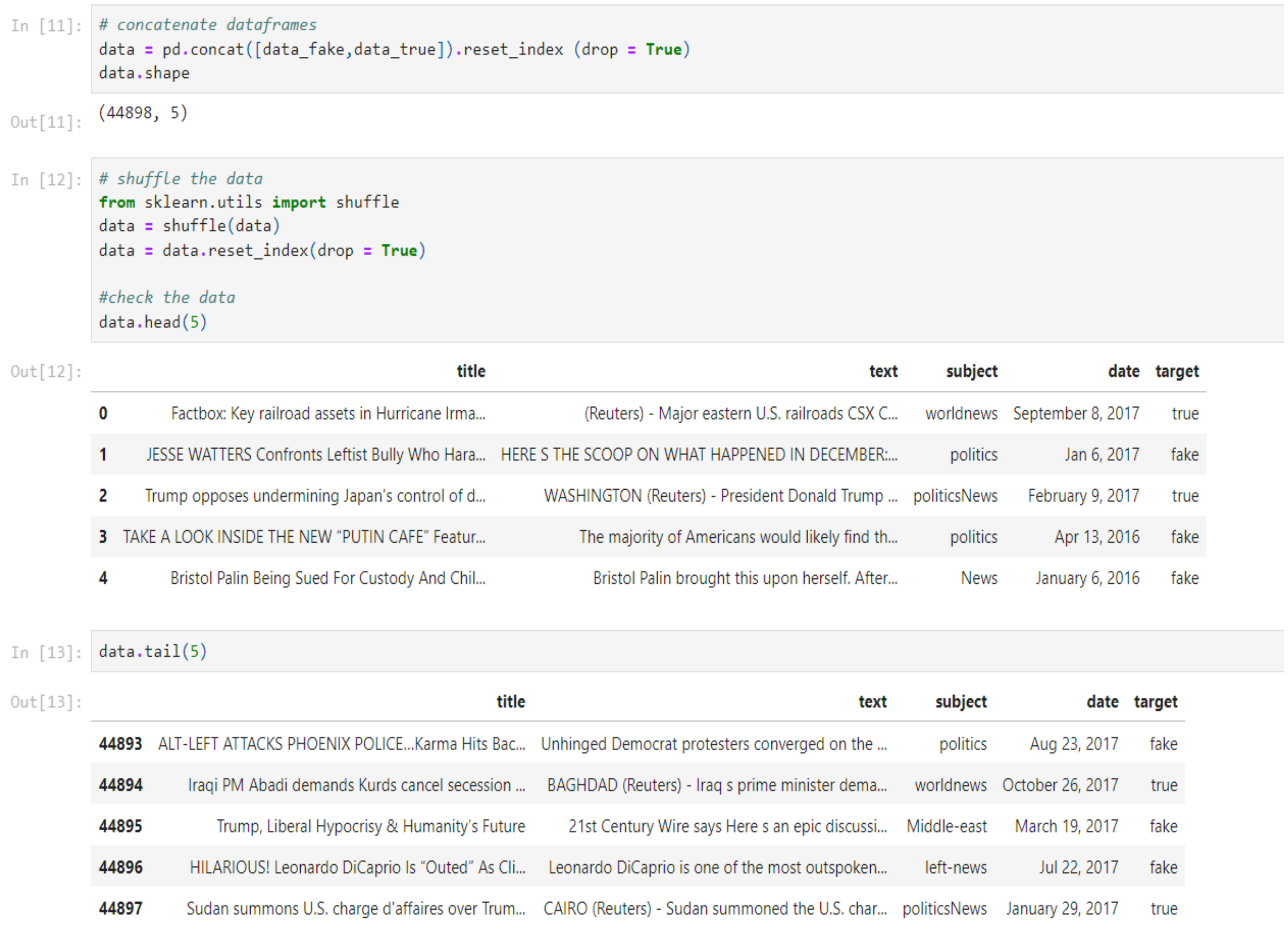
Out[90]:

	title	text	subject	date	target
23471	Seven Iranians freed in the prisoner swap have...	21st Century Wire says This week, the historic...	Middle-east	January 20, 2016	fake
23472	#Hashtag Hell & The Fake Left	By Dady Chery and Gilbert MercierAll writers ...	Middle-east	January 19, 2016	fake
23473	Astroturfing: Journalist Reveals Brainwashing ...	Vic Bishop Waking TimesOur reality is carefull...	Middle-east	January 19, 2016	fake
23474	The New American Century: An Era of Fraud	Paul Craig RobertsIn the last years of the 20t...	Middle-east	January 19, 2016	fake
23475	Hillary Clinton: 'Israel First' (and no peace ...	Robert Fantina CounterpunchAlthough the United...	Middle-east	January 18, 2016	fake

```
In [91]: data_true_manual_testing.head()
```

Out[91]:

	title	text	subject	date	target
21407	Mata Pires, owner of embattled Brazil builder ...	SAO PAULO (Reuters) - Cesar Mata Pires, the ow...	worldnews	August 22, 2017	true
21408	U.S., North Korea clash at U.N. forum over nuc...	GENEVA (Reuters) - North Korea and the United ...	worldnews	August 22, 2017	true
21409	U.S., North Korea clash at U.N. arms forum on ...	GENEVA (Reuters) - North Korea and the United ...	worldnews	August 22, 2017	true
21410	Headless torso could belong to submarine journ...	COPENHAGEN (Reuters) - Danish police said on T...	worldnews	August 22, 2017	true
21411	North Korea shipments to Syria chemical arms a...	UNITED NATIONS (Reuters) - Two North Korean sh...	worldnews	August 21, 2017	true



```
In [11]: # concatenate dataframes
         data = pd.concat([data_fake,data_true]).reset_index (drop = True)
         data.shape

Out[11]: (44898, 5)
```

```
In [12]: # shuffle the data
         from sklearn.utils import shuffle
         data = shuffle(data)
         data = data.reset_index(drop = True)

         #check the data
         data.head(5)
```

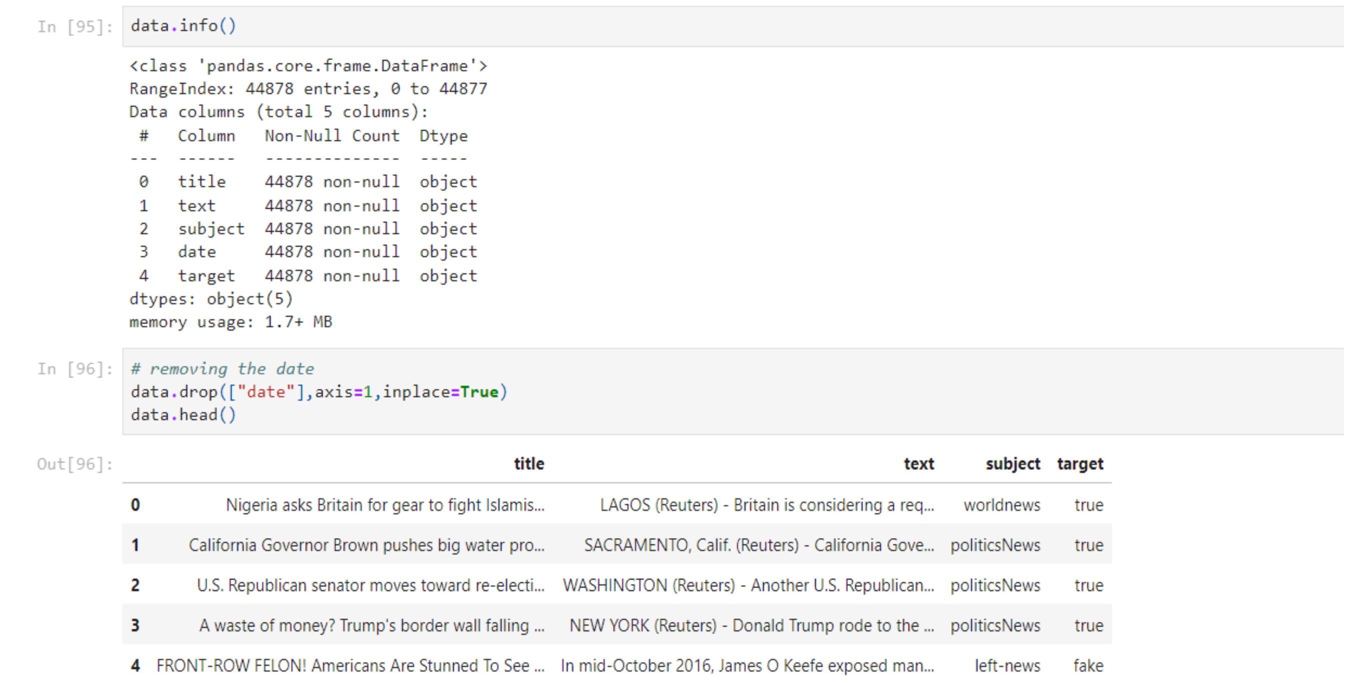
Out[12]:

	title	text	subject	date	target
0	Factbox: Key railroad assets in Hurricane Irma...	(Reuters) - Major eastern U.S. railroads CSX C...	worldnews	September 8, 2017	true
1	JESSE WATTERS Confronts Leftist Bully Who Hara...	HERE S THE SCOOP ON WHAT HAPPENED IN DECEMBER:...	politics	Jan 6, 2017	fake
2	Trump opposes undermining Japan's control of d...	WASHINGTON (Reuters) - President Donald Trump ...	politicsNews	February 9, 2017	true
3	TAKE A LOOK INSIDE THE NEW "PUTIN CAFE" Featur...	The majority of Americans would likely find th...	politics	Apr 13, 2016	fake
4	Bristol Palin Being Sued For Custody And Chil...	Bristol Palin brought this upon herself. After...	News	January 6, 2016	fake

```
In [13]: data.tail(5)
```

Out[13]:

	title	text	subject	date	target
44893	ALT-LEFT ATTACKS PHOENIX POLICE...Karma Hits Bac...	Unhinged Democrat protesters converged on the ...	politics	Aug 23, 2017	fake
44894	Iraqi PM Abadi demands Kurds cancel secession ...	BAGHDAD (Reuters) - Iraq s prime minister dema...	worldnews	October 26, 2017	true
44895	Trump, Liberal Hypocrisy & Humanity's Future	21st Century Wire says Here s an epic discussi...	Middle-east	March 19, 2017	fake
44896	HILARIOUS! Leonardo DiCaprio Is "Outed" As Cli...	Leonardo DiCaprio is one of the most outspoken...	left-news	Jul 22, 2017	fake
44897	Sudan summons U.S. charge d'affaires over Trum...	CAIRO (Reuters) - Sudan summoned the U.S. char...	politicsNews	January 29, 2017	true



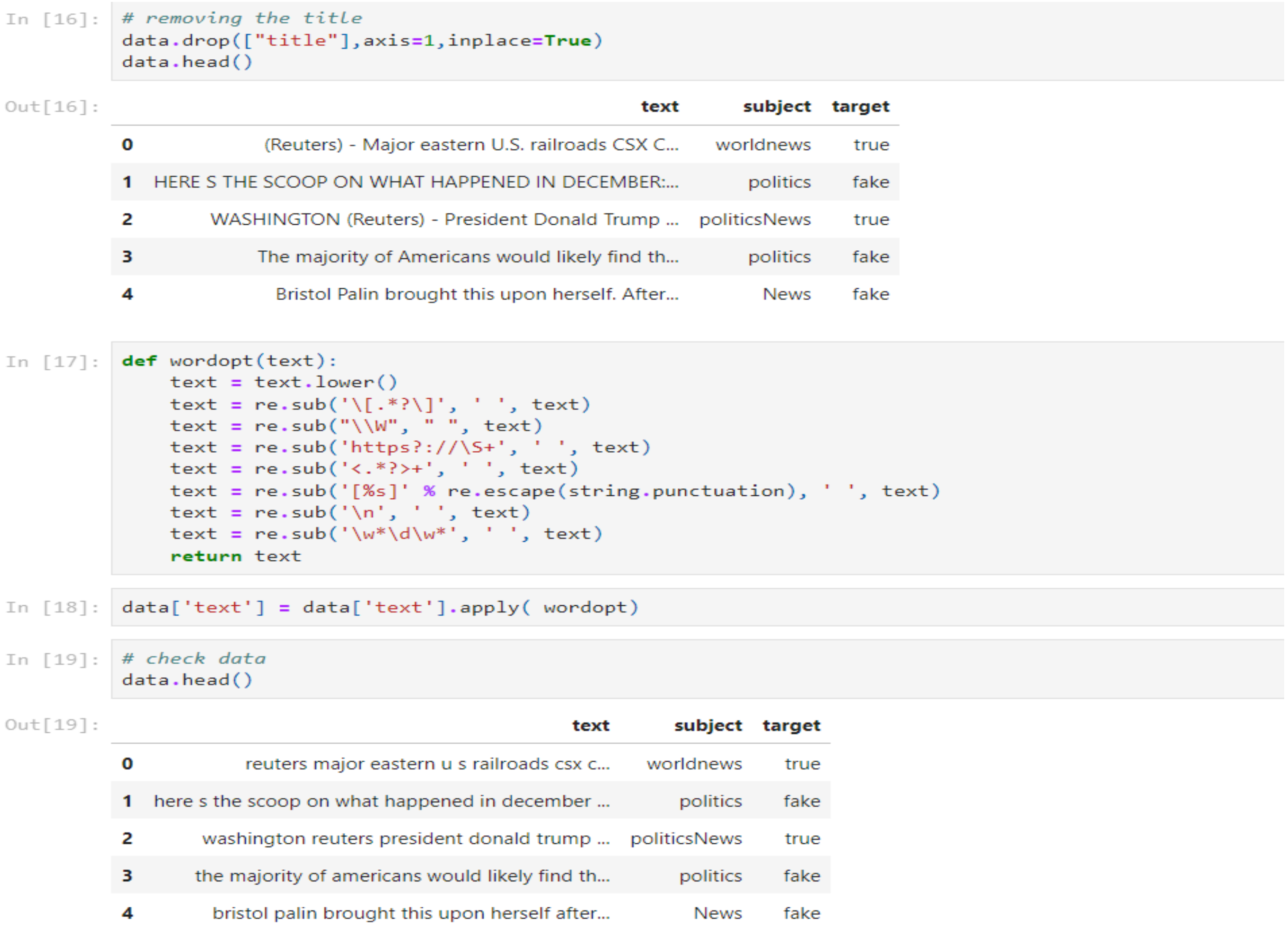
In [95]:
```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 44878 entries, 0 to 44877
Data columns (total 5 columns):
 #   Column   Non-Null Count  Dtype 
---  ------   --------------  ----- 
 0   title    44878 non-null  object
 1   text     44878 non-null  object
 2   subject  44878 non-null  object
 3   date     44878 non-null  object
 4   target   44878 non-null  object
dtypes: object(5)
memory usage: 1.7+ MB
```

In [96]:
```
# removing the date
data.drop(["date"],axis=1,inplace=True)
data.head()
```

Out[96]:

	title	text	subject	target
0	Nigeria asks Britain for gear to fight Islamis...	LAGOS (Reuters) - Britain is considering a req...	worldnews	true
1	California Governor Brown pushes big water pro...	SACRAMENTO, Calif. (Reuters) - California Gove...	politicsNews	true
2	U.S. Republican senator moves toward re-electi...	WASHINGTON (Reuters) - Another U.S. Republican...	politicsNews	true
3	A waste of money? Trump's border wall falling ...	NEW YORK (Reuters) - Donald Trump rode to the ...	politicsNews	true
4	FRONT-ROW FELON! Americans Are Stunned To See ...	In mid-October 2016, James O Keefe exposed man...	left-news	fake



In [16]:
```
# removing the title
data.drop(["title"],axis=1,inplace=True)
data.head()
```

Out[16]:

	text	subject	target
0	(Reuters) - Major eastern U.S. railroads CSX C...	worldnews	true
1	HERE S THE SCOOP ON WHAT HAPPENED IN DECEMBER:...	politics	fake
2	WASHINGTON (Reuters) - President Donald Trump ...	politicsNews	true
3	The majority of Americans would likely find th...	politics	fake
4	Bristol Palin brought this upon herself. After...	News	fake

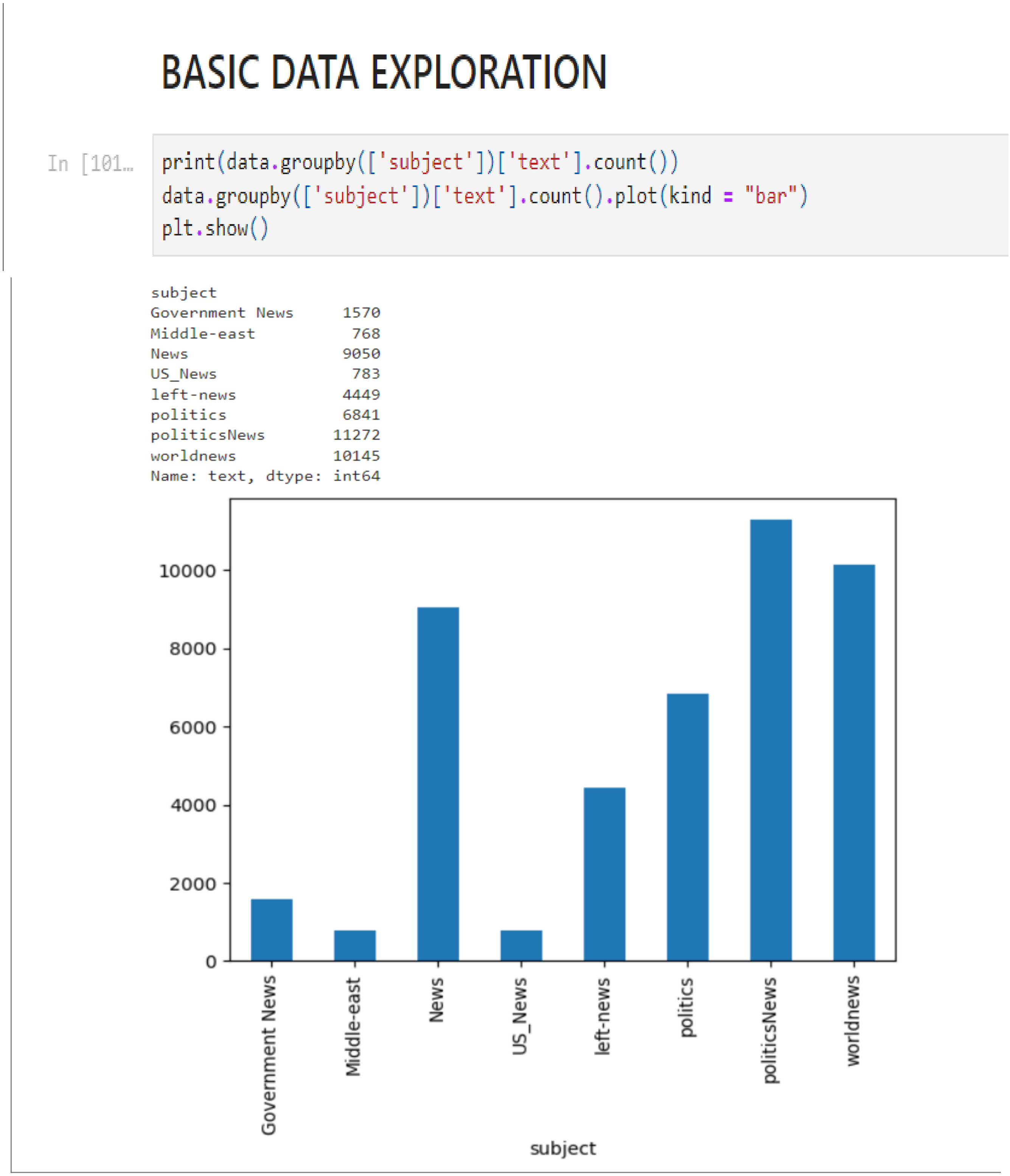
In [17]:
```
def wordopt(text):
    text = text.lower()
    text = re.sub('\[.*?\]', ' ', text)
    text = re.sub("\\W", " ", text)
    text = re.sub('https?://\S+', ' ', text)
    text = re.sub('<.*?>+', ' ', text)
    text = re.sub('[%s]' % re.escape(string.punctuation), ' ', text)
    text = re.sub('\n', ' ', text)
    text = re.sub('\w*\d\w*', ' ', text)
    return text
```

In [18]:
```
data['text'] = data['text'].apply( wordopt)
```

In [19]:
```
# check data
data.head()
```

Out[19]:

	text	subject	target
0	reuters major eastern u s railroads csx c...	worldnews	true
1	here s the scoop on what happened in december ...	politics	fake
2	washington reuters president donald trump ...	politicsNews	true
3	the majority of americans would likely find th...	politics	fake
4	bristol palin brought this upon herself after...	News	fake

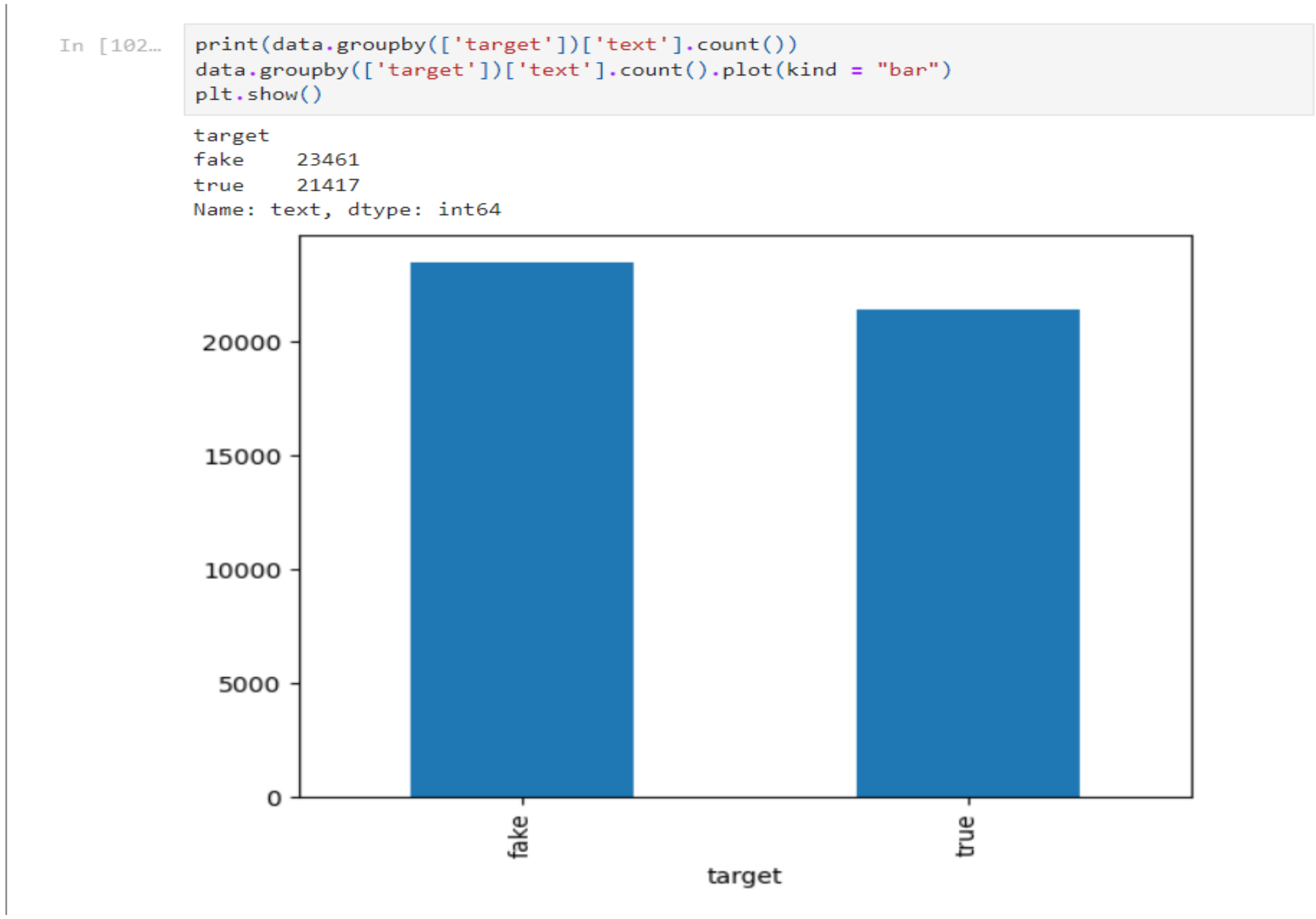


BASIC DATA EXPLORATION

In [101…

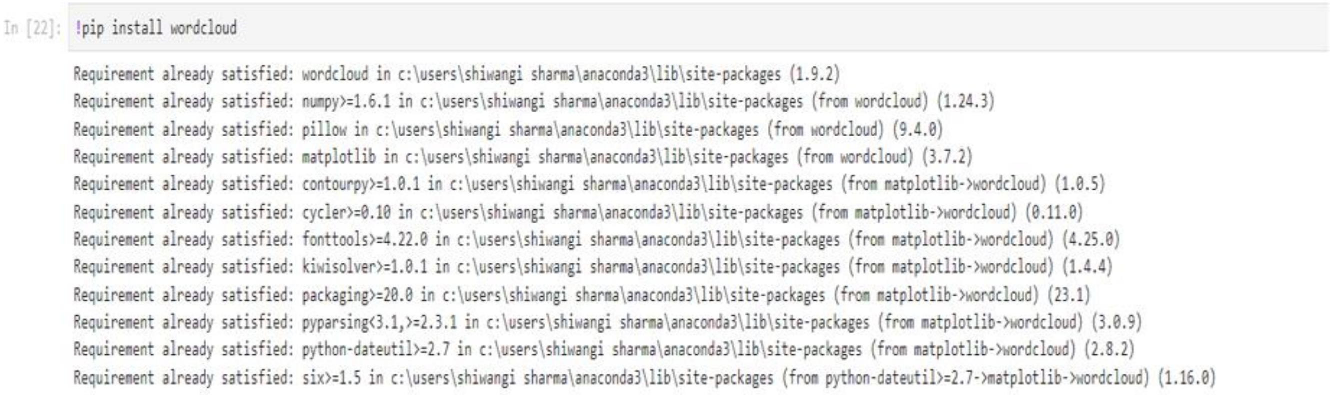
```
print(data.groupby(['subject'])['text'].count())
data.groupby(['subject'])['text'].count().plot(kind = "bar")
plt.show()
```

```
subject
Government News     1570
Middle-east          768
News                9050
US_News              783
left-news           4449
politics            6841
politicsNews       11272
worldnews          10145
Name: text, dtype: int64
```



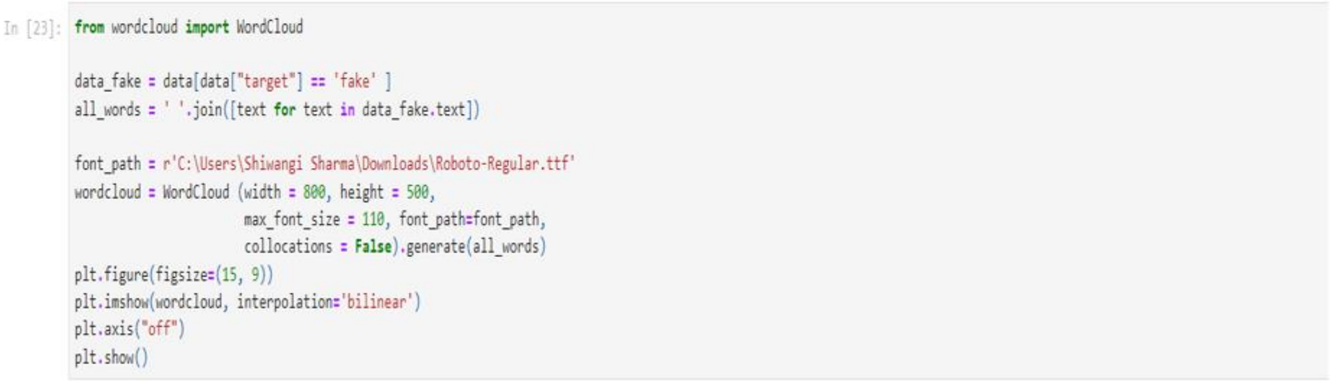
```
In [102…  print(data.groupby(['target'])['text'].count())
          data.groupby(['target'])['text'].count().plot(kind = "bar")
          plt.show()
```

```
target
fake     23461
true     21417
Name: text, dtype: int64
```

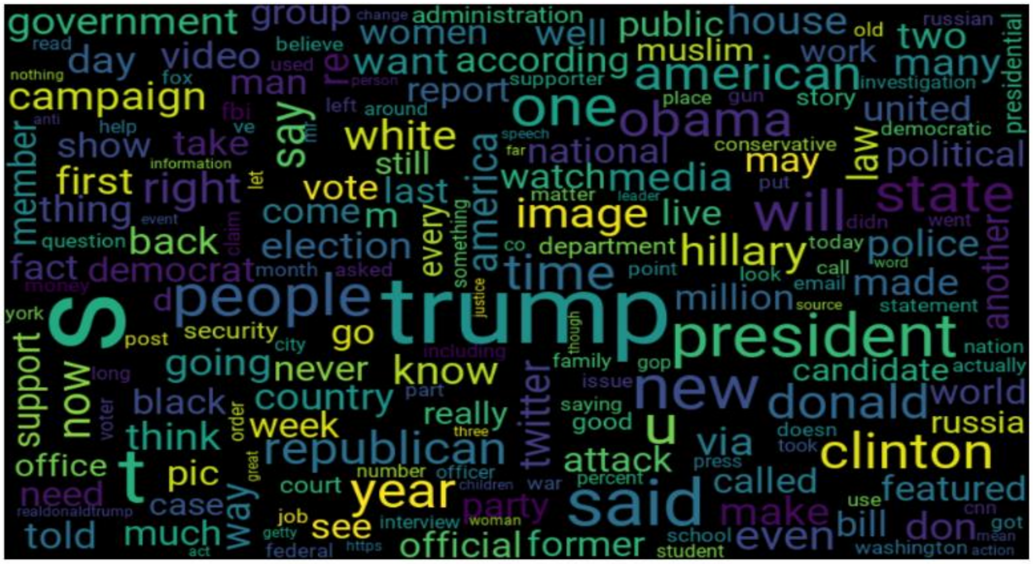


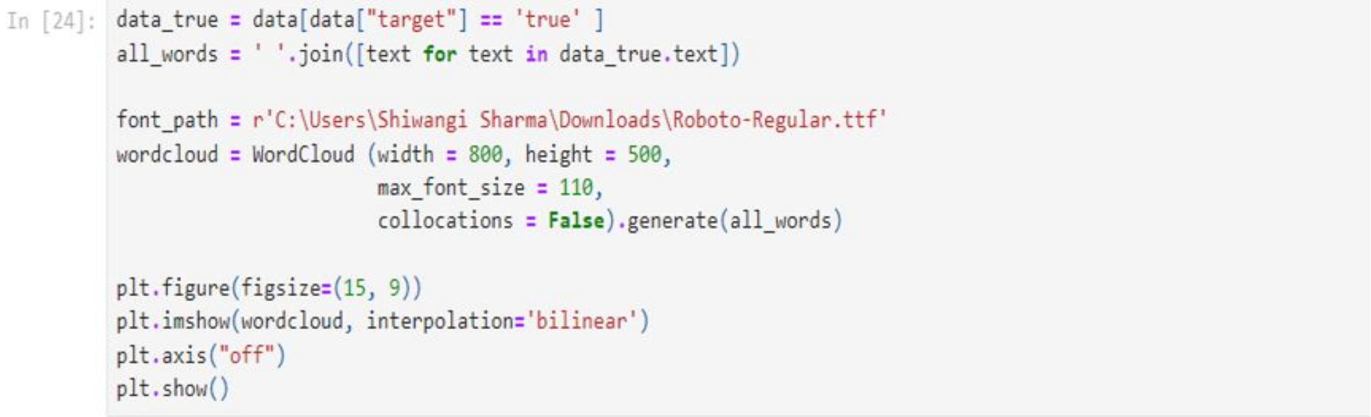
```
In [22]: !pip install wordcloud
```

```
Requirement already satisfied: wordcloud in c:\users\shiwangi sharma\anaconda3\lib\site-packages (1.9.2)
Requirement already satisfied: numpy>=1.6.1 in c:\users\shiwangi sharma\anaconda3\lib\site-packages (from wordcloud) (1.24.3)
Requirement already satisfied: pillow in c:\users\shiwangi sharma\anaconda3\lib\site-packages (from wordcloud) (9.4.0)
Requirement already satisfied: matplotlib in c:\users\shiwangi sharma\anaconda3\lib\site-packages (from wordcloud) (3.7.2)
Requirement already satisfied: contourpy>=1.0.1 in c:\users\shiwangi sharma\anaconda3\lib\site-packages (from matplotlib->wordcloud) (1.0.5)
Requirement already satisfied: cycler>=0.10 in c:\users\shiwangi sharma\anaconda3\lib\site-packages (from matplotlib->wordcloud) (0.11.0)
Requirement already satisfied: fonttools>=4.22.0 in c:\users\shiwangi sharma\anaconda3\lib\site-packages (from matplotlib->wordcloud) (4.25.0)
Requirement already satisfied: kiwisolver>=1.0.1 in c:\users\shiwangi sharma\anaconda3\lib\site-packages (from matplotlib->wordcloud) (1.4.4)
Requirement already satisfied: packaging>=20.0 in c:\users\shiwangi sharma\anaconda3\lib\site-packages (from matplotlib->wordcloud) (23.1)
Requirement already satisfied: pyparsing<3.1,>=2.3.1 in c:\users\shiwangi sharma\anaconda3\lib\site-packages (from matplotlib->wordcloud) (3.0.9)
Requirement already satisfied: python-dateutil>=2.7 in c:\users\shiwangi sharma\anaconda3\lib\site-packages (from matplotlib->wordcloud) (2.8.2)
Requirement already satisfied: six>=1.5 in c:\users\shiwangi sharma\anaconda3\lib\site-packages (from python-dateutil>=2.7->matplotlib->wordcloud) (1.16.0)
```

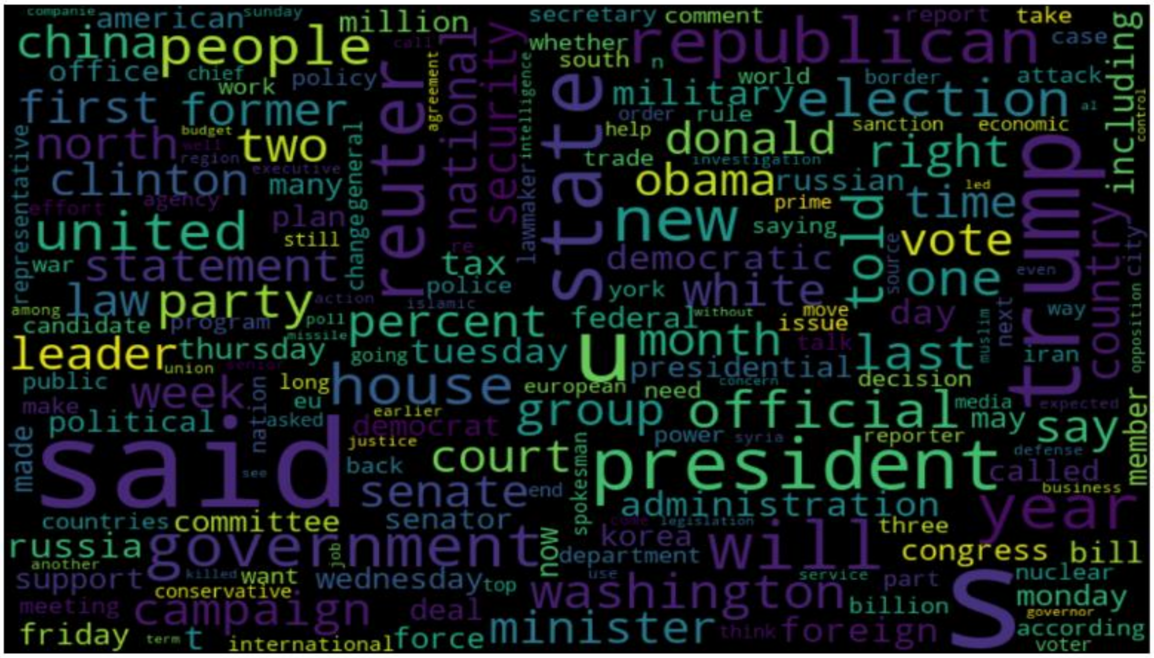


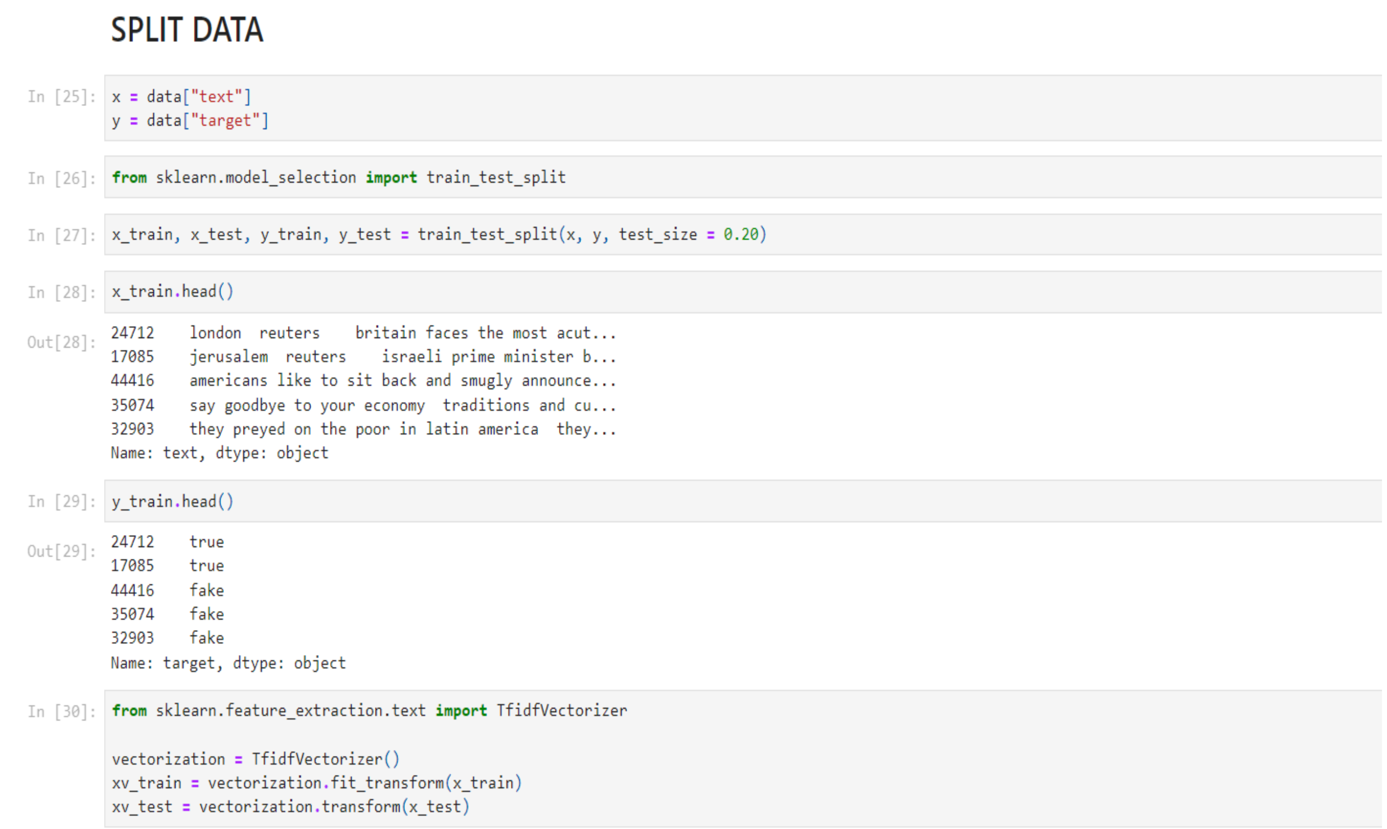
```
In [23]: from wordcloud import WordCloud

         data_fake = data[data["target"] == 'fake' ]
         all_words = ' '.join([text for text in data_fake.text])

         font_path = r'C:\Users\Shiwangi Sharma\Downloads\Roboto-Regular.ttf'
         wordcloud = WordCloud (width = 800, height = 500,
                                max_font_size = 110, font_path=font_path,
                                collocations = False).generate(all_words)
         plt.figure(figsize=(15, 9))
         plt.imshow(wordcloud, interpolation='bilinear')
         plt.axis("off")
         plt.show()
```




In [24]:

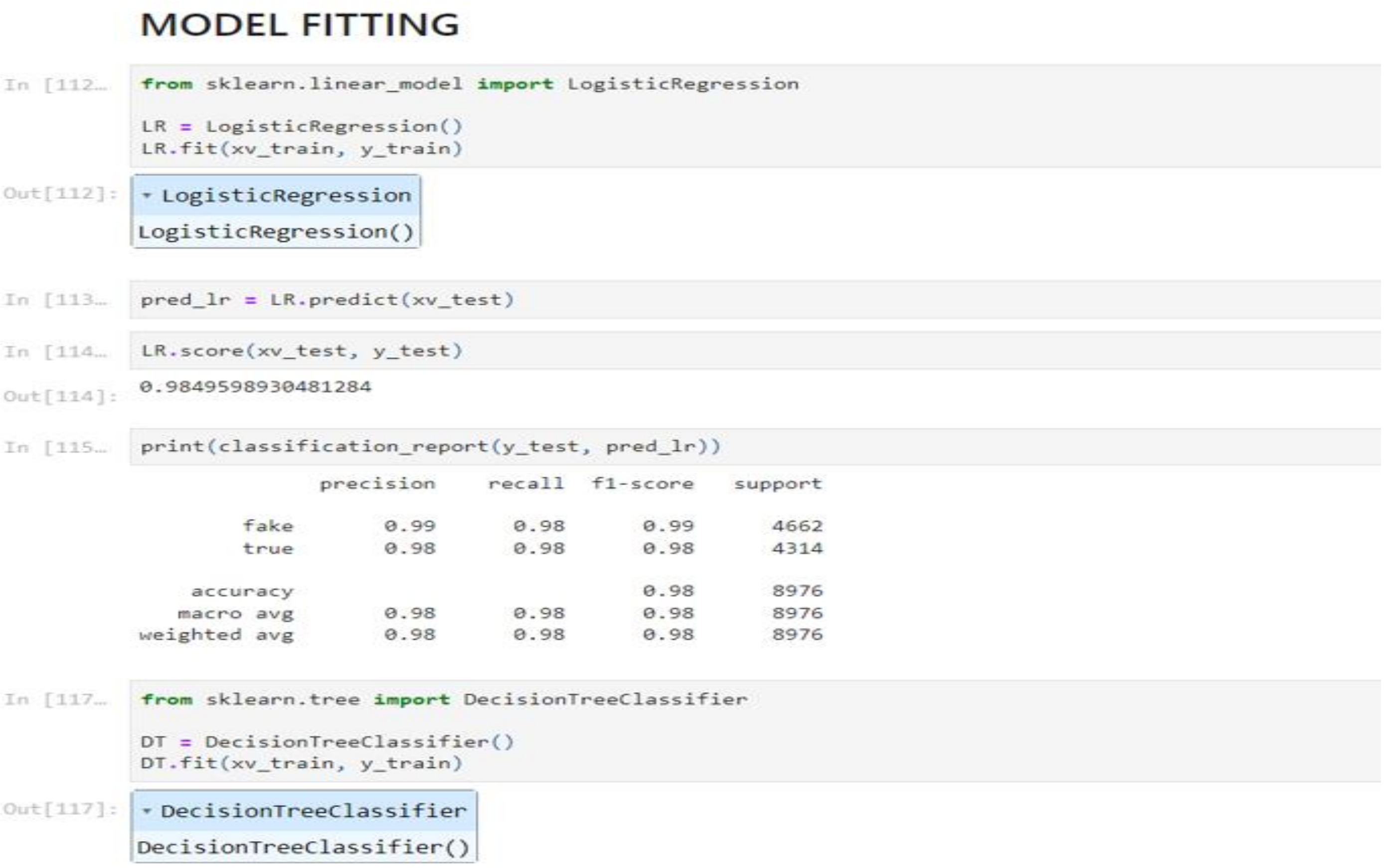
```
data_true = data[data["target"] == 'true' ]
all_words = ' '.join([text for text in data_true.text])

font_path = r'C:\Users\Shiwangi Sharma\Downloads\Roboto-Regular.ttf'
wordcloud = WordCloud (width = 800, height = 500,
                      max_font_size = 110,
                      collocations = False).generate(all_words)

plt.figure(figsize=(15, 9))
plt.imshow(wordcloud, interpolation='bilinear')
plt.axis("off")
plt.show()
```





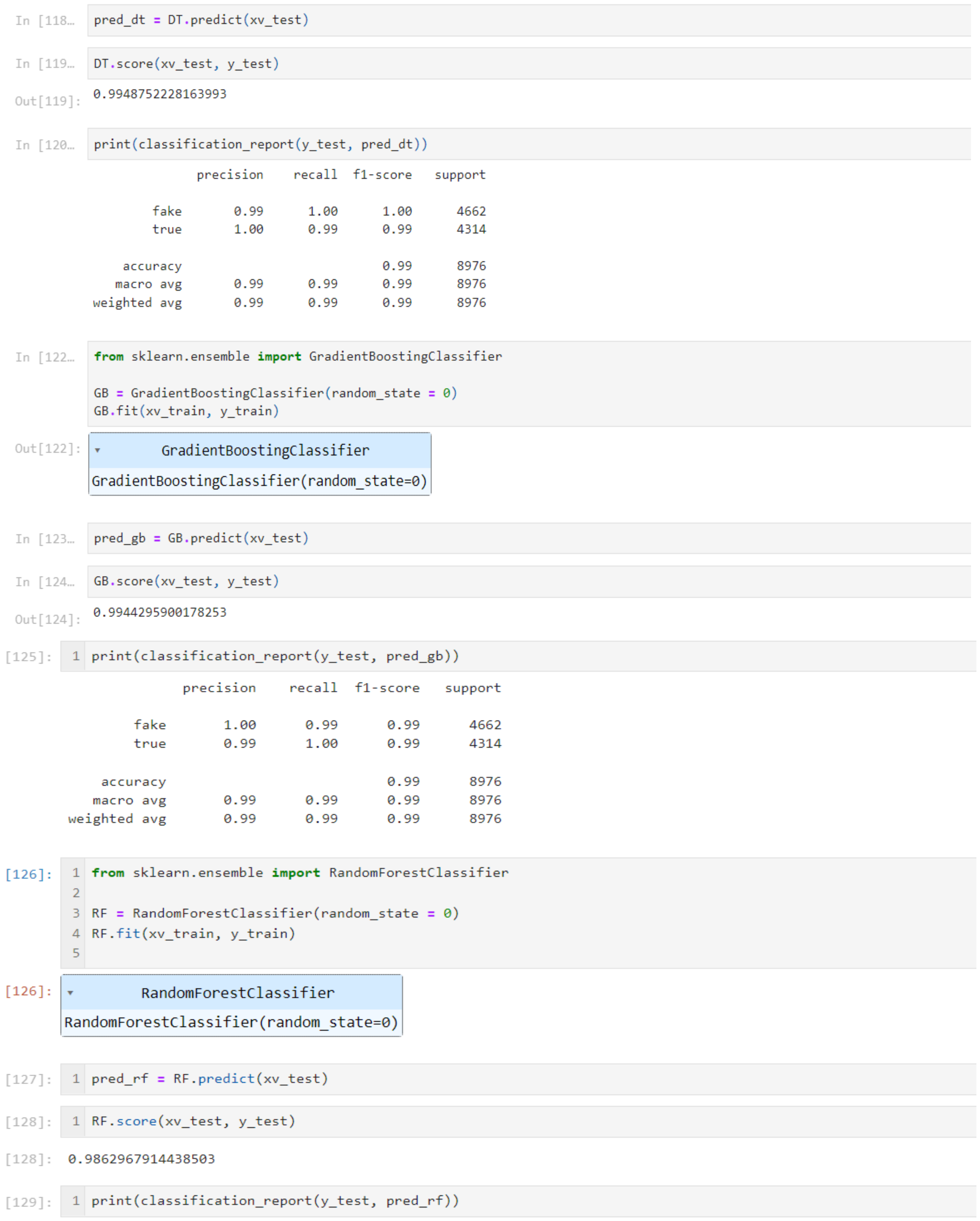
SPLIT DATA

```
In [25]: x = data["text"]
         y = data["target"]

In [26]: from sklearn.model_selection import train_test_split

In [27]: x_train, x_test, y_train, y_test = train_test_split(x, y, test_size = 0.20)

In [28]: x_train.head()

Out[28]: 24712    london  reuters    britain faces the most acut...
         17085    jerusalem  reuters    israeli prime minister b...
         44416    americans like to sit back and smugly announce...
         35074    say goodbye to your economy  traditions and cu...
         32903    they preyed on the poor in latin america  they...
         Name: text, dtype: object

In [29]: y_train.head()

Out[29]: 24712    true
         17085    true
         44416    fake
         35074    fake
         32903    fake
         Name: target, dtype: object

In [30]: from sklearn.feature_extraction.text import TfidfVectorizer

         vectorization = TfidfVectorizer()
         xv_train = vectorization.fit_transform(x_train)
         xv_test = vectorization.transform(x_test)
```



MODEL FITTING

```
In [112…  from sklearn.linear_model import LogisticRegression

          LR = LogisticRegression()
          LR.fit(xv_train, y_train)

Out[112]: ▾ LogisticRegression
          LogisticRegression()

In [113…  pred_lr = LR.predict(xv_test)

In [114…  LR.score(xv_test, y_test)

Out[114]: 0.9849598930481284

In [115…  print(classification_report(y_test, pred_lr))

                        precision    recall  f1-score   support

                fake       0.99      0.98      0.99      4662
                true       0.98      0.98      0.98      4314

            accuracy                           0.98      8976
           macro avg       0.98      0.98      0.98      8976
        weighted avg       0.98      0.98      0.98      8976

In [117…  from sklearn.tree import DecisionTreeClassifier

          DT = DecisionTreeClassifier()
          DT.fit(xv_train, y_train)

Out[117]: ▾ DecisionTreeClassifier
          DecisionTreeClassifier()
```



```
In [118…  pred_dt = DT.predict(xv_test)
```

```
In [119…  DT.score(xv_test, y_test)
```

```
Out[119]: 0.9948752228163993
```

```
In [120…  print(classification_report(y_test, pred_dt))
```

```
              precision    recall  f1-score   support

        fake       0.99      1.00      1.00      4662
        true       1.00      0.99      0.99      4314

    accuracy                           0.99      8976
   macro avg       0.99      0.99      0.99      8976
weighted avg       0.99      0.99      0.99      8976
```

```
In [122…  from sklearn.ensemble import GradientBoostingClassifier

          GB = GradientBoostingClassifier(random_state = 0)
          GB.fit(xv_train, y_train)
```

```
Out[122]: ▾ GradientBoostingClassifier
          GradientBoostingClassifier(random_state=0)
```

```
In [123…  pred_gb = GB.predict(xv_test)
```

```
In [124…  GB.score(xv_test, y_test)
```

```
Out[124]: 0.9944295900178253
```

```
[125]:  print(classification_report(y_test, pred_gb))
```

```
              precision    recall  f1-score   support

        fake       1.00      0.99      0.99      4662
        true       0.99      1.00      0.99      4314

    accuracy                           0.99      8976
   macro avg       0.99      0.99      0.99      8976
weighted avg       0.99      0.99      0.99      8976
```

```
[126]:  from sklearn.ensemble import RandomForestClassifier

        RF = RandomForestClassifier(random_state = 0)
        RF.fit(xv_train, y_train)

```

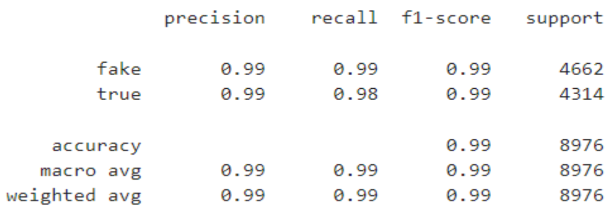
```
[126]:  ▾ RandomForestClassifier
        RandomForestClassifier(random_state=0)
```

```
[127]:  pred_rf = RF.predict(xv_test)
```

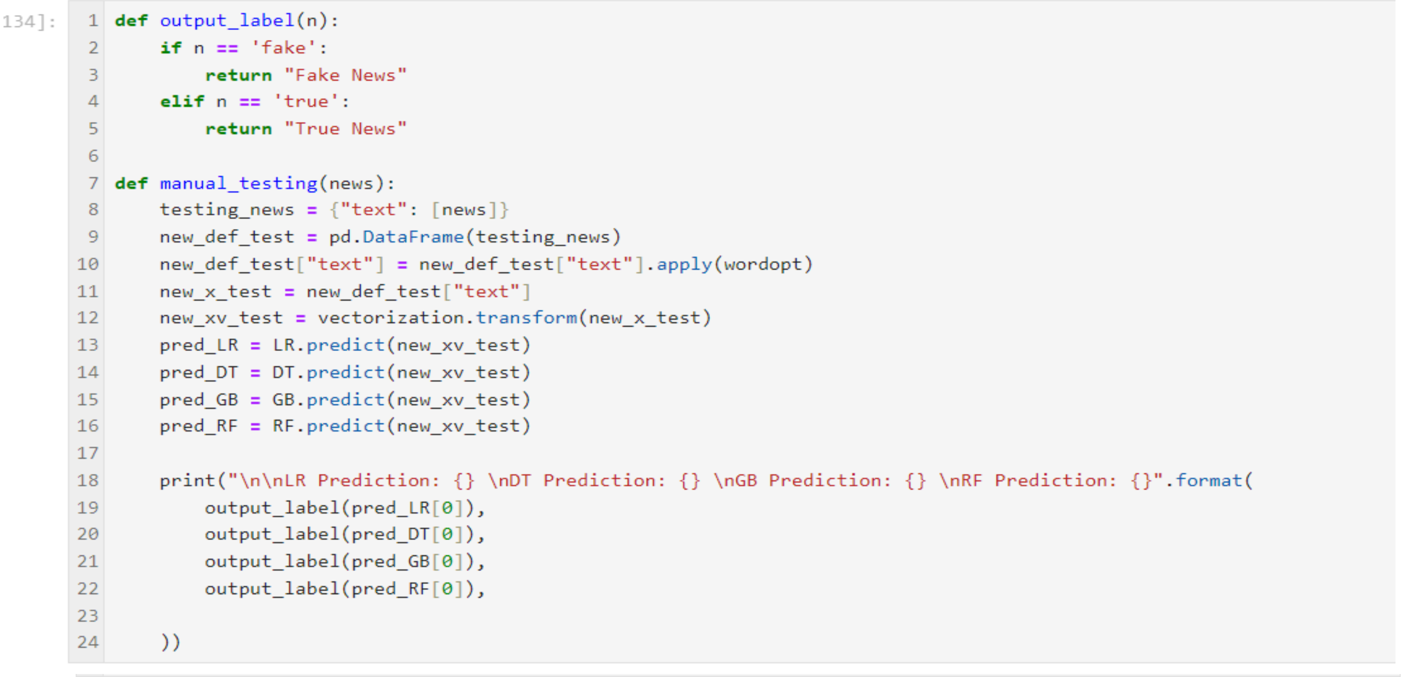
```
[128]:  RF.score(xv_test, y_test)
```

```
[128]:  0.9862967914438503
```

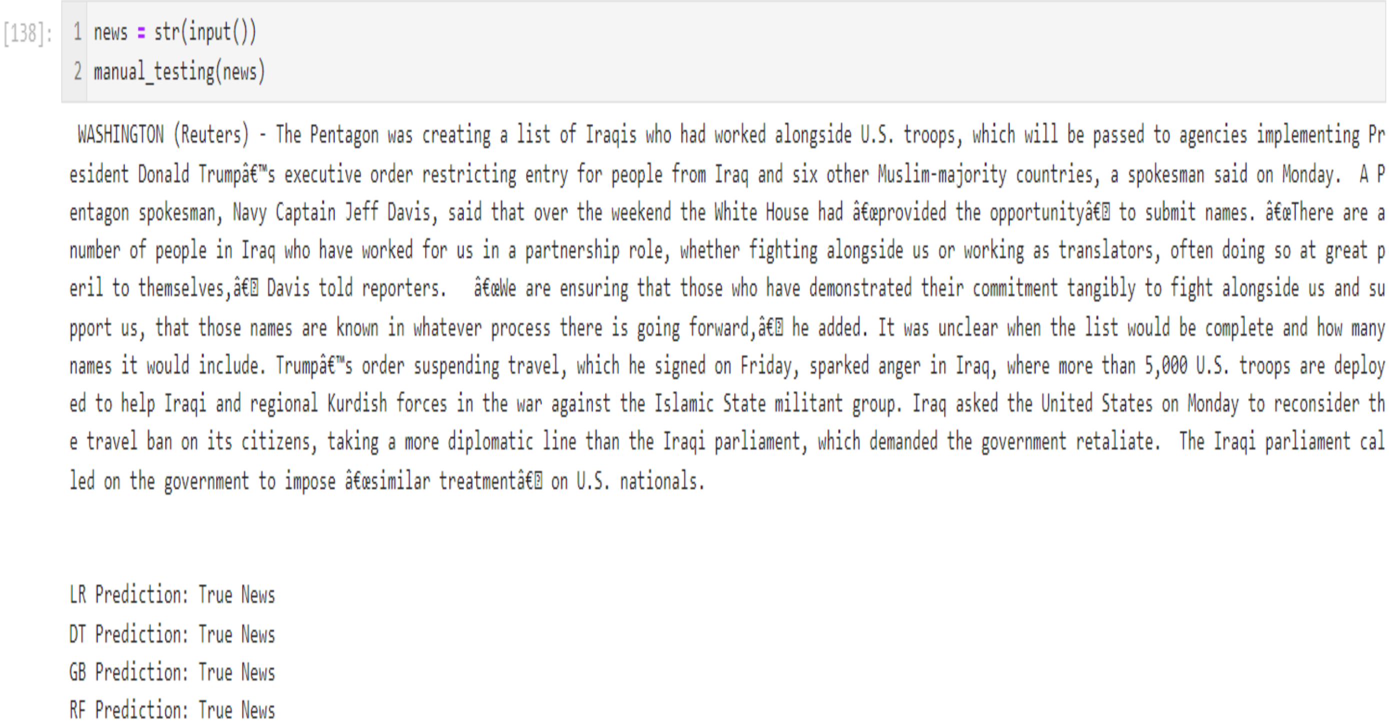
```
[129]:  print(classification_report(y_test, pred_rf))
```



```
              precision    recall  f1-score   support

        fake       0.99      0.99      0.99      4662
        true       0.99      0.98      0.99      4314

    accuracy                           0.99      8976
   macro avg       0.99      0.99      0.99      8976
weighted avg       0.99      0.99      0.99      8976
```

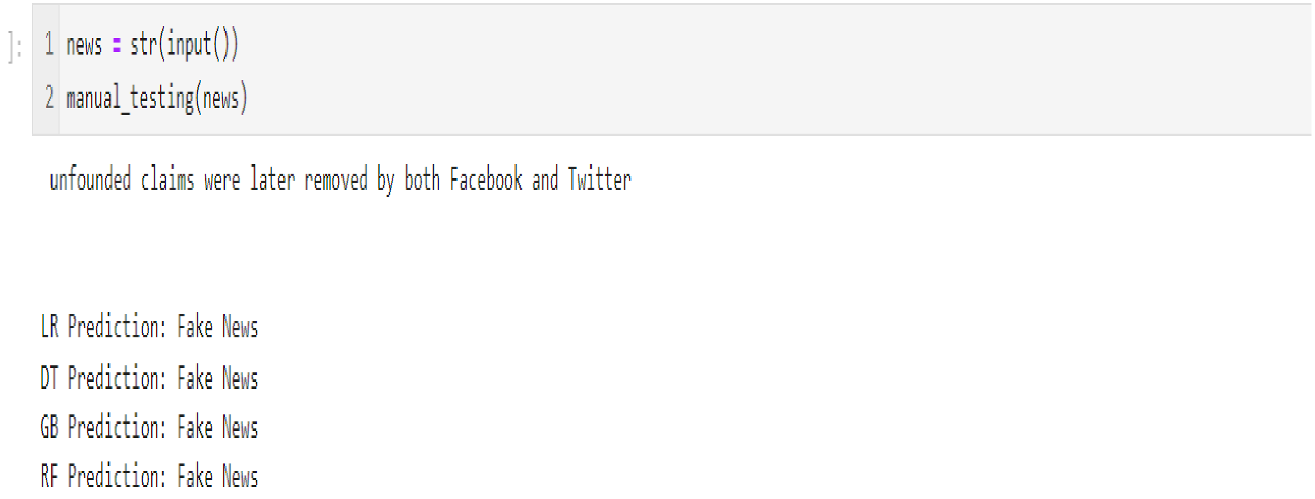


```
134]:
def output_label(n):
    if n == 'fake':
        return "Fake News"
    elif n == 'true':
        return "True News"

def manual_testing(news):
    testing_news = {"text": [news]}
    new_def_test = pd.DataFrame(testing_news)
    new_def_test["text"] = new_def_test["text"].apply(wordopt)
    new_x_test = new_def_test["text"]
    new_xv_test = vectorization.transform(new_x_test)
    pred_LR = LR.predict(new_xv_test)
    pred_DT = DT.predict(new_xv_test)
    pred_GB = GB.predict(new_xv_test)
    pred_RF = RF.predict(new_xv_test)

    print("\n\nLR Prediction: {} \nDT Prediction: {} \nGB Prediction: {} \nRF Prediction: {}".format(
        output_label(pred_LR[0]),
        output_label(pred_DT[0]),
        output_label(pred_GB[0]),
        output_label(pred_RF[0]),

    ))
```



```
[138]:
news = str(input())
manual_testing(news)
```

```
 WASHINGTON (Reuters) - The Pentagon was creating a list of Iraqis who had worked alongside U.S. troops, which will be passed to agencies implementing Pr
esident Donald Trumpâ€™s executive order restricting entry for people from Iraq and six other Muslim-majority countries, a spokesman said on Monday.  A P
entagon spokesman, Navy Captain Jeff Davis, said that over the weekend the White House had â€œprovided the opportunityâ€ to submit names. â€œThere are a
number of people in Iraq who have worked for us in a partnership role, whether fighting alongside us or working as translators, often doing so at great p
eril to themselves,â€ Davis told reporters.  â€œWe are ensuring that those who have demonstrated their commitment tangibly to fight alongside us and su
pport us, that those names are known in whatever process there is going forward,â€ he added. It was unclear when the list would be complete and how many
names it would include. Trumpâ€™s order suspending travel, which he signed on Friday, sparked anger in Iraq, where more than 5,000 U.S. troops are deploy
ed to help Iraqi and regional Kurdish forces in the war against the Islamic State militant group. Iraq asked the United States on Monday to reconsider th
e travel ban on its citizens, taking a more diplomatic line than the Iraqi parliament, which demanded the government retaliate.  The Iraqi parliament cal
led on the government to impose â€œsimilar treatmentâ€ on U.S. nationals.


LR Prediction: True News
DT Prediction: True News
GB Prediction: True News
RF Prediction: True News
```



```
news = str(input())
manual_testing(news)
```

```
unfounded claims were later removed by both Facebook and Twitter



LR Prediction: Fake News
DT Prediction: Fake News
GB Prediction: Fake News
RF Prediction: Fake News
```

app.py

```
# REQUIRED LIBRARIES
import streamlit as st
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import re
import nltk
from nltk.stem import PorterStemmer, WordNetLemmatizer
from sklearn.metrics import accuracy_score
from sklearn.metrics import classification_report
import string

## READING DATA

data_fake = pd.read_csv("C:/Users/Shiwangi Sharma/Documents/Major project/dataset/fake.csv")
data_true = pd.read_csv("C:/Users/Shiwangi Sharma/Documents/Major project/dataset/true.csv")

## DATA CLEANING & PREPROCESSING
```

```
data_fake['target'] ='fake'
data_true['target'] = 'true'


data_fake_manual_testing = data_fake.tail(10)
for i in range(23480, 23470, -1):
    data_fake.drop([i], axis = 0, inplace = True)


data_true_manual_testing = data_true.tail(10)
for i in range(21416, 21406, -1):
    data_fake.drop([i], axis = 0, inplace = True)


data_fake_manual_testing["target"] = 'fake'
data_true_manual_testing["target"] = 'true'

# concatenate dataframes
data = pd.concat([data_fake,data_true]).reset_index (drop = True)

# shuffle the data
from sklearn.utils import shuffle
data = shuffle(data)
data = data.reset_index(drop = True)



# removing the date
```

```
data.drop(["date"],axis=1,inplace=True)


# removing the title
data.drop(["title"],axis=1,inplace=True)


def wordopt(text):
    text = text.lower()
    text = re.sub('\[.*?\]', ' ', text)
    text = re.sub("\\W", " ", text)
    text = re.sub('https?://\S+', ' ', text)
    text = re.sub('<.*?>+', ' ', text)
    text = re.sub('[%s]' % re.escape(string.punctuation), ' ', text)
    text = re.sub('\n', ' ', text)
    text = re.sub('\w*\d\w*', ' ', text)
    return text


data['text'] = data['text'].apply( wordopt)


## SPLIT DATA


x = data["text"]
y = data["target"]


from sklearn.model_selection import train_test_split
```

```
x_train, x_test, y_train, y_test = train_test_split(x, y, test_size = 0.20)

from sklearn.feature_extraction.text import TfidfVectorizer

vectorization = TfidfVectorizer()
xv_train = vectorization.fit_transform(x_train)
xv_test = vectorization.transform(x_test)


## MODEL FITTING

# Logistic Regression
from sklearn.linear_model import LogisticRegression
LR = LogisticRegression()
LR.fit(xv_train, y_train)

# Decision Tree
from sklearn.tree import DecisionTreeClassifier
DT = DecisionTreeClassifier()
DT.fit(xv_train, y_train)


# Gradient Boosting
```

```
from sklearn.ensemble import GradientBoostingClassifier

GB = GradientBoostingClassifier(random_state = 0)

GB.fit(xv_train, y_train)


# Random Forest

from sklearn.ensemble import RandomForestClassifier

RF = RandomForestClassifier(random_state = 0)

RF.fit(xv_train, y_train)




# Function to convert model output to human-readable label

def output_label(n):

    if n == 'fake':

        return "Fake News"

    elif n == 'true':

        return "True News"


# Function for manual testing

def manual_testing(news):

    testing_news = {"text": [news]}

    new_def_test = pd.DataFrame(testing_news)

    new_def_test["text"] = new_def_test["text"].apply(wordopt)

    new_x_test = new_def_test["text"]

    new_xv_test = vectorization.transform(new_x_test)
    pred_LR = LR.predict(new_xv_test)
    pred_DT = DT.predict(new_xv_test)
    pred_GB = GB.predict(new_xv_test)
    pred_RF = RF.predict(new_xv_test)


    print("\n\nLR Prediction: {} \nDT Prediction: {} \nGB Prediction: {} \nRF Prediction: {}".format(
        output_label(pred_LR[0]),
        output_label(pred_DT[0]),
        output_label(pred_GB[0]),
        output_label(pred_RF[0]),


    ))



# WEBSITE


st.title('Fake News Detector')
input_text = st.text_input("Enter News Article")


# Function for model prediction
def prediction(input_text):
    input_data = vectorization.transform([input_text])
    prediction = LR.predict(input_data)
    prediction = DT.predict(input_data)
    prediction = GB.predict(input_data)
    prediction = RF.predict(input_data)
    return prediction[0]

# Display prediction result on the web app
if st.button("Check News"):
    if input_text:
        pred = prediction(input_text)
        result_label = output_label(pred)
        st.write(f'The News is {result_label}')
```

Annexure III:

RESEARCH PAPER

Submission is in progress….

FAKE NEWS DETECTION

Aayush Ujjwal[1], Jyoti Kataria[2]
Student, Department of BCA, K.R. Mangalam University, Sohna Road,
Gurugram, Haryana, India

ABSTRACT

With the recent social media boom, the spread of fake news has become a great concern for everybody. It has been used to manipulate public opinions, influence the election. A 2018 MIT study found that fake news spreads six times faster on Twitter than real news. The credibility and trust in the news media are at an all-time low. It is becoming increasingly difficult to determine which news is real and which is fake. Various machine learning methods have been used to separate real news from fake ones. In this study, we tried to accomplish that using Natural Language Processing. There are lots of machine learning models that we can use to have better results.

Now there is some confusion present in the authenticity of the correctness. But it definitely opens the window for further research. There are some of the aspects that has to be kept in mind considering the fact that fake news detection is not only a simple web interface but also a quite complex thing that includes a lot of backend work.

1. INTRODUCTION

Fake news is untrue information presented as news. It often has the aim of damaging the reputation of a person or entity or making money through advertising revenue. Once common in print, the prevalence of fake news has increased with the rise of social media, especially the Facebook News Feed. During the 2016 US presidential election, various kinds of fake news about the candidates widely spread in the online social networks, which may have a significant effect on the election results. According to a post-election statistical report, online social networks account for more than 41.8% of the fake news data traffic in the election, which is much greater than the data traffic shares of both traditional TV/radio/print medium and online search engines respectively. Fake news detection is becoming increasingly difficult because people who have ill intentions are writing the fake pieces so convincingly that it is difficult to separate from real news. What we have done is a simplistic approach that looks at the news text and tries to predict whether they may be fake or not.
Fake news can be intimidating as they attract more audience than normal. People use them because this can be a very good marketing strategy. But the money earned might not live upto fact that it can harm people.