ARTIFICIAL INTELLIGENCE

FAKE NEWS DETECTION USING NLP

PHASE-V DOCUMENTATION

AMBALAVANAN N

820421104006

Problem Statement:

In the era of digital information, the proliferation of fake news has become a significant concern, impacting public opinion, social harmony, and political stability. The challenge lies in accurately identifying and combating misinformation in the vast sea of online content. The problem to be addressed is developing a robust and efficient Natural Language Processing (NLP) based system for Fake News Detection. This system should be capable of analysing textual information from various sources and distinguishing between authentic news and fabricated or misleading content. The goal is to create a reliable solution that can contribute to the promotion of accurate information, safeguarding the integrity of online discourse, and fostering a more informed society.

Objective:

The objective of this document is to explore and implement Natural Language Processing (NLP) techniques for detecting fake news.

Explanation:

Explore NLP Techniques:

The objective involves exploring various techniques and methods within the field of Natural Language Processing. This exploration may include understanding concepts like tokenization, stemming, lemmatization, sentiment analysis, and other text processing techniques used in NLP.

Implement NLP for Fake News Detection:

The objective emphasizes the practical application of NLP techniques specifically for the purpose of identifying fake news. This implementation can involve using algorithms, models, and tools within the realm of NLP to process textual data and distinguish between genuine and fake news articles.

Focus on Detection:

The primary focus of the objective is on the detection aspect. This means developing methods or models that can analyse textual information, extract relevant features, and determine whether a given piece of news is authentic or fake. The objective doesn't just involve understanding the theoretical concepts but also applying them effectively to solve the real-world problem of identifying misinformation.

Addressing the Problem:

By stating the objective as detecting fake news, the document aims to contribute to addressing the growing concern of misinformation in the digital age. Fake news can have significant social, political, and economic consequences. The objective underscores the importance of using advanced NLP techniques to combat this issue and promote accurate information dissemination.

Introduction to NLP:

Natural Language Processing (NLP) is a branch of artificial intelligence that focuses on enabling computers to understand, interpret, and generate human language in a way that is both valuable and meaningful. In essence, NLP seeks to bridge the gap between human communication and computer understanding. By employing algorithms and computational linguistics, NLP allows machines to process and analyze vast amounts of natural language data, thereby facilitating tasks that involve language comprehension and generation.

Applications of NLP:

NLP finds applications in a wide range of fields, including but not limited to:

Sentiment Analysis: Determining the sentiment or emotional tone of a piece of text, whether it's positive, negative, or neutral. Sentiment analysis is crucial for understanding public opinion, customer feedback, and social media interactions.

Machine Translation: Translating text or speech from one language to another. NLP powers machine translation services like Google Translate, making multilingual communication more accessible.

Named Entity Recognition (NER): Identifying and classifying named entities (such as names of people, organizations, locations, etc.) within textual data. NER is vital for information retrieval and knowledge extraction.

Speech Recognition: Converting spoken language into text. NLP enables voice assistants like Siri and Alexa to understand spoken commands and respond appropriately.

Question Answering Systems: Building systems that can comprehend complex questions posed in natural language and provide accurate and relevant answers. Such systems are valuable in customer support and educational applications.

Fake News Detection: Identifying and flagging misinformation or fake news articles by analyzing the language patterns and context. NLP techniques play a significant role in distinguishing between reliable and unreliable sources of information.

NLP Techniques Relevant to Fake News Detection:

Tokenization:

Definition: Tokenization is the process of breaking down text into smaller units, usually words or subwords (tokens). These tokens serve as the fundamental units for further NLP analysis.

Importance in Fake News Detection: Tokenization allows analyzing the frequency of specific words, identifying unique vocabulary, and understanding the structure of sentences. By breaking down the text into tokens, NLP models can process and understand the language patterns present in news articles.

Stemming:

Definition: Stemming is the process of reducing words to their root or base form. It involves removing prefixes or suffixes to obtain the word's core meaning.

Importance in Fake News Detection: Stemming helps in reducing words to their base forms, ensuring that variations of words are treated as the same entity. This consistency aids in comparing and matching words during the analysis, enhancing the accuracy of fake news detection models.

Lemmatization:

Definition: Lemmatization is the process of reducing words to their base or dictionary form (lemma). Unlike stemming, lemmatization considers the word's meaning and context before transforming it.

Importance in Fake News Detection: Lemmatization ensures that words are transformed into their canonical forms, facilitating a more accurate analysis of the text. It helps in identifying the root meaning of words, which is crucial for understanding the context of news articles and detecting subtle nuances in language.

Sentiment Analysis:

Definition: Sentiment analysis, also known as opinion mining, involves determining the emotional tone expressed in a piece of text, such as positive, negative, or neutral.

Importance in Fake News Detection: Sentiment analysis can be used to assess the sentiment associated with specific news articles or social media posts. By understanding the sentiment, analysts can gauge the emotional tone of the information, helping in identifying potentially misleading or biased content.

exploring advanced techniques like deep learning models (e.g., LSTM, BERT) for improved fake news detection accuracy

There are a lot of Machine Learning and Deep Learning Models present outside, few of them are Decision Tree, Random Forest, k-Nearest Neighbors classifiers, SVM (Support Vector Machines),RNN(Recurrent neural Network),LSTM(Long Short Term Memory),CSI(CAPTURE,SCORE,INTEGRATE) which is a hybrid deep model especially built for fake news detection. Since our problem is to determine whether the news is true or fake basically binary classification problem.

Some other models are TI-CNN (Text and Image convolutional neural network) is trained with both images and text information simultaneously. The convolutional neural network makes the model to see the entire input at once, and it can be trained much faster than LSTM and many other RNN models.

Naive Bayes (NB) Classifier is a deterministic algorithm that uses the Bayes theorem to classify data. Naive Bayes classifiers work by correlating the use of tokens (typically words, or sometimes other constructions, syntactic or not), with spam and non-spam e-mails and then using Bayes theorem to calculate a probability that an email is or is not a spam message.

From the anlyses, we can say that the machine learning models such as Random Forest, Decision tree, SVM, NB are performing good as neural networks for small datasets but they are less performing in large datasets. At the same time, CNNs and RNNs embedded deep learning models are performing well in both small and large sized datasets. Comparatively, the hybrid models perform better than ordinary neural networks. Now, we will see some of the hybrid models in detailed manner which are CNN-LSTM,CNN-BILSTM,BI-LSTM,LSTM.

LSTM(Long Term Short Memory) :

The architecture of LSTM

LSTMs deal with both Long Term Memory (LTM) and Short Term Memory (STM) and for making the calculations simple and effective it uses the concept of gates.

- **Forget Gate:** LTM goes to forget gate and it forgets information that is not useful.

- **Learn Gate:** Event (current input) and STM are combined together so that necessary information that we have recently learned from STM can be applied to the current input.

- **Remember Gate:** LTM information that we haven't forget and STM and Event are combined together in Remember gate which works as updated LTM.

- **Use Gate:** This gate also uses LTM, STM, and Event to predict the output of the current event which works as an updated STM.

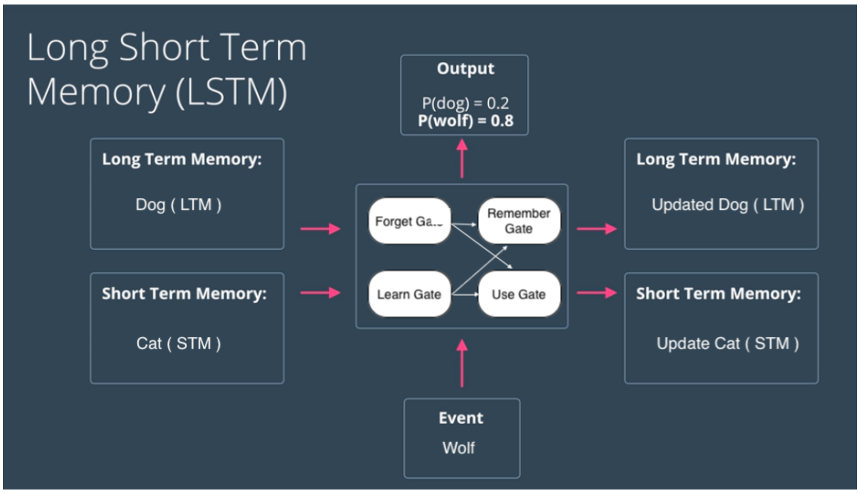
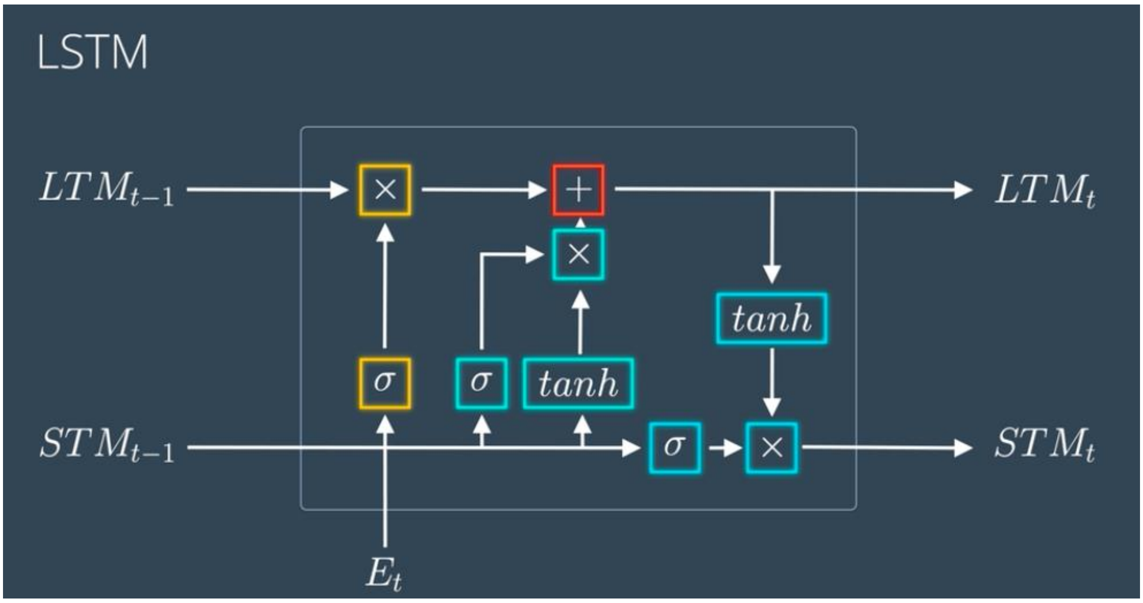
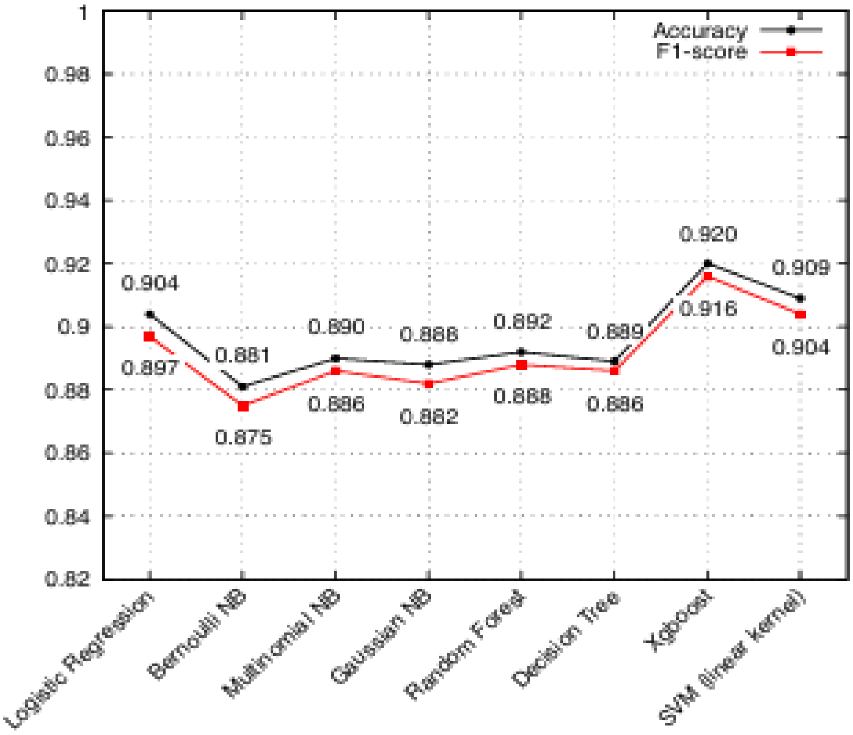


Figure : Structure of the gates

Mathematical Structure



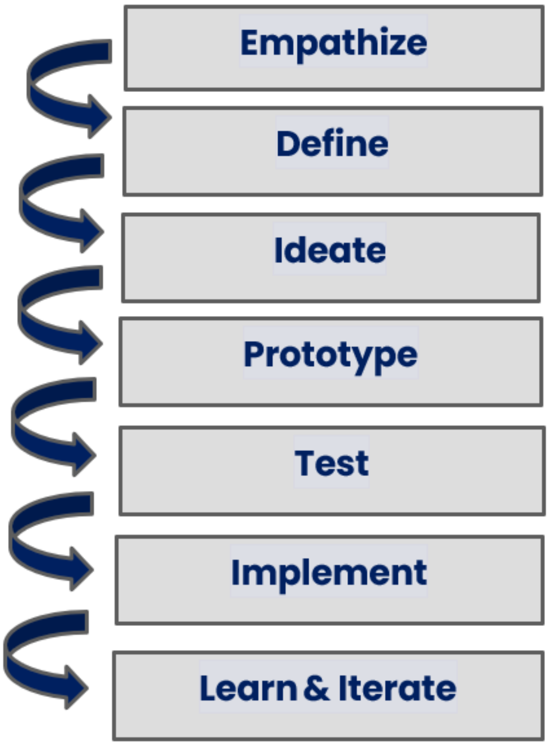
Performance of ML models using bag of words feature:



Design Thinking Process:

- To better understand the impact of news, on individuals and communities we conducted interviews and surveys with users. We also explored existing methods for detecting news and their limitations.
- We framed the problem as follows; "How can we effectively detect news and prevent its spread using NLP techniques?" In doing we aimed to improve the accuracy of identifying news and enhance user awareness of reliable sources.
- During the ideation phase we brainstormed solutions that involved NLP algorithms, user interfaces and educational components. We created prototypes of user interfaces and interaction flows for our news detection system.
- Moving to the prototype stage we developed low fidelity versions of user interfaces to visualize the product. Additionally, we built NLP algorithms of analysing linguistic patterns and inconsistencies in news articles. To support data storage and retrieval a backend system was implemented.
- To gather feedback on our interface designs usability testing sessions were conducted with users. Simultaneously we tested our NLP algorithms using a dataset containing both fake news articles. We refined these algorithms based on accuracy metrics as valuable input from users.
- Finally, after incorporating all improvements from testing phases, we developed the version of our news detection system by integrating the user interface, with the NLP algorithms.
- We introduced a mechanism, for learning enabling the system to adjust and learn from forms of misinformation as time goes on.
- Learning and Iteration.
- We closely monitored the systems performance in real life scenarios gathering feedback from users and making necessary updates to enhance accuracy and improve the user experience.
- Dataset, Data Preprocessing and Feature Extraction.
- For our dataset we utilized a curated collection of news articles that were labelled encompassing both news stories and examples of fake news.

To preprocess the data, we tokenized the text by removing characters, stop words (common words, with little meaning) and irrelevant information.



Dataset, Data Pre-processing, and Feature Extraction:

- **Dataset:**

  The **Fake and Real News Dataset** available at the provided Kaggle link is a comprehensive collection of textual data designed for research and analysis in the domain of fake news detection. The dataset contains two main categories of news articles: **"Fake News"** and **"Real News."** Each category provides valuable insights into the characteristics of both genuine and deceptive news content, making it a valuable resource for natural language processing (NLP) and machine learning projects focused on misinformation detection.

  Dataset Link: https://www.kaggle.com/datasets/clmentbisaillon/fake-and-real-news-dataset

- **Data Preprocessing:** Cleaned and tokenized the text data, removing special characters, stop words, and irrelevant information.
- **Feature Extraction:** Used techniques like TF-IDF (Term Frequency-Inverse Document Frequency) to convert text data into numerical features, enabling the LSTM model to process the input data effectively.

Phases of development:

Phase 1:

We collected and explored a given dataset and explain about the project. Choose the fake news dataset available.

Phase 2:

we can explore innovative techniques such as ensemble methods and deep learning architectures to improve the prediction system's accuracy and robustness. Consider exploring advanced techniques like deep learning models (e.g., LSTM, BERT) for improved fake news detection accuracy.

Phase 3:

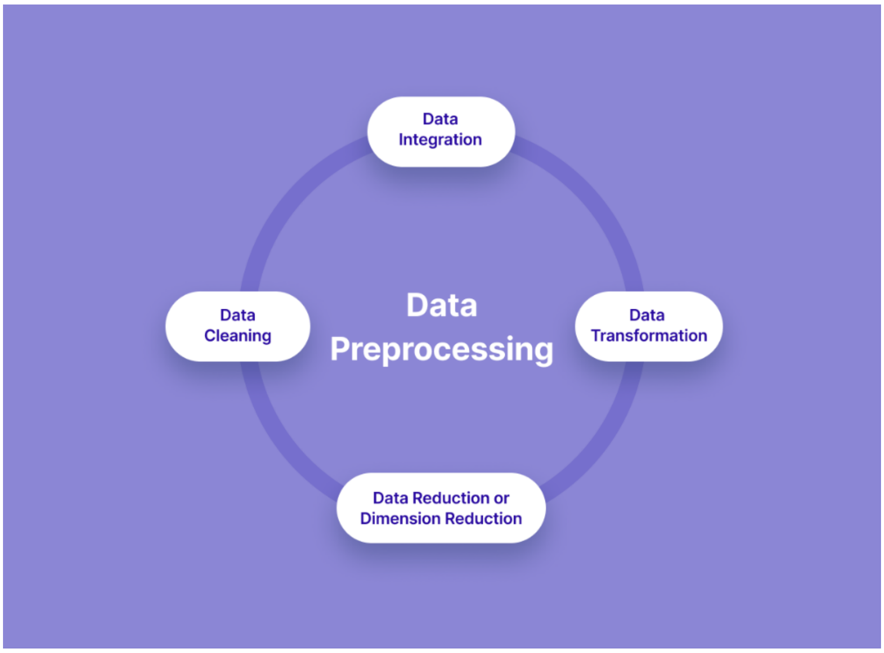
Begin building your project by loading and preprocessing the dataset. Begin building the fake news detection model by loading and preprocessing the dataset. Load the fake news dataset and preprocess the textual data.

Phase 4:

Continue building the fake news detection model by applying NLP techniques and training a classification Model. Text Preprocessing and Feature Extraction Model training and evaluation.

Data Preprocessing:

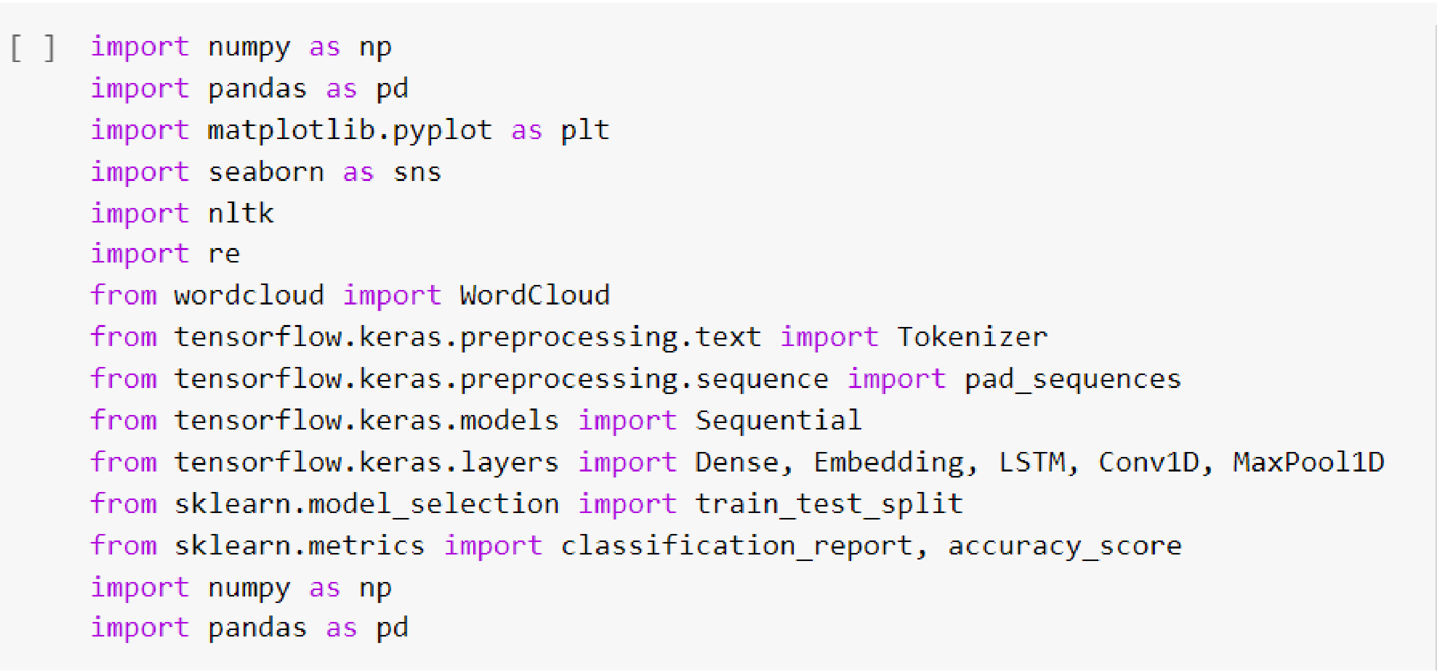
Data preprocessing is a fundamental step in the data analysis and machine learning process, involving the transformation and cleaning of raw data into a format suitable for modeling. In the realm of data preprocessing, various techniques are applied to enhance the quality and usability of the data. One common task is handling missing values, where strategies like removal or imputation are employed to deal with incomplete data points. Categorical variables, which contain labels and not numerical values, require encoding methods like one-hot encoding or label encoding to convert them into a numerical format understandable by machine learning algorithms. Scaling features is another crucial aspect, ensuring that all features are on a similar scale to prevent certain variables from dominating others. In the context of Natural Language Processing (NLP), specific preprocessing steps include tokenization, lowercasing, removing special characters, stopwords, and performing stemming or lemmatization. Additionally, techniques like vectorization using word embeddings and handling out-of-vocabulary words are essential for processing textual data. Proper data preprocessing not only ensures the data is ready for analysis but also significantly impacts the performance and accuracy of machine learning models.



Data cleaning:

Data cleaning is a process of removing inconsistencies in the dataset and incorrect values .It also in involves handling missing values either by removing them or assigning them average values. It helps to improve the efficiency of the model.

In the first step, we will only remove the unnecessary data points from the dataset which does not helps in improving the model performance.

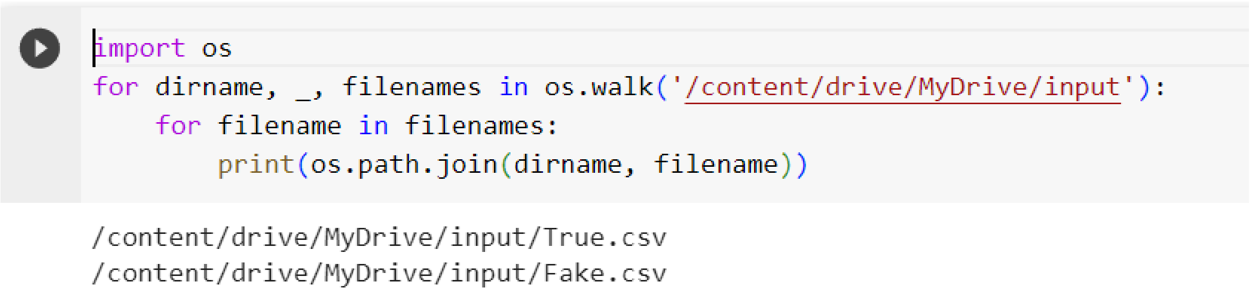


```
[ ]  import numpy as np
     import pandas as pd
     import matplotlib.pyplot as plt
     import seaborn as sns
     import nltk
     import re
     from wordcloud import WordCloud
     from tensorflow.keras.preprocessing.text import Tokenizer
     from tensorflow.keras.preprocessing.sequence import pad_sequences
     from tensorflow.keras.models import Sequential
     from tensorflow.keras.layers import Dense, Embedding, LSTM, Conv1D, MaxPool1D
     from sklearn.model_selection import train_test_split
     from sklearn.metrics import classification_report, accuracy_score
     import numpy as np
     import pandas as pd
```

Initially we import the necessary packages for our data cleaning process and also in the future purposes

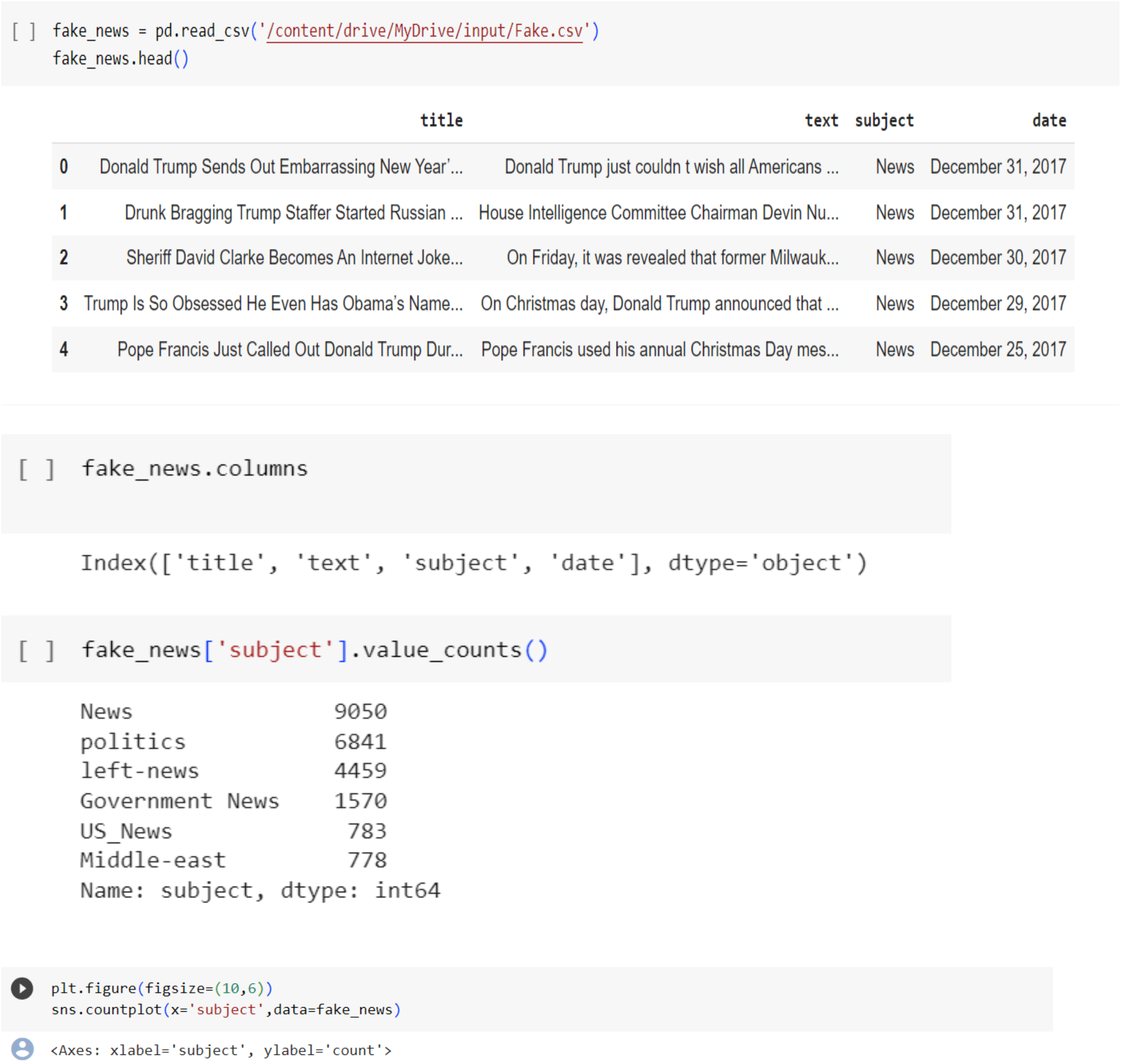
we use these packages in various stages of our cleaning process and also in the future in which we need to build models.

Here, we read the .csv files of true and fake news and then explore the count values of their subjects



```
import os
for dirname, _, filenames in os.walk('/content/drive/MyDrive/input'):
    for filename in filenames:
        print(os.path.join(dirname, filename))
```

```
/content/drive/MyDrive/input/True.csv
/content/drive/MyDrive/input/Fake.csv
```



```
[ ] fake_news = pd.read_csv('/content/drive/MyDrive/input/Fake.csv')
    fake_news.head()
```

	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

```
[ ] fake_news.columns
```

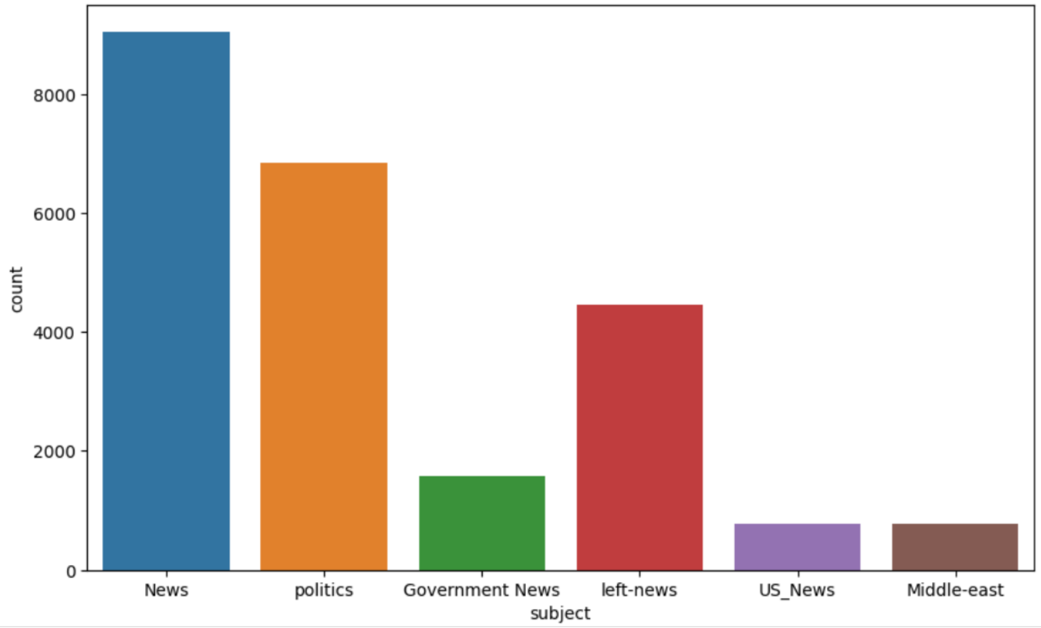
```
Index(['title', 'text', 'subject', 'date'], dtype='object')
```

```
[ ] fake_news['subject'].value_counts()
```

```
News               9050
politics           6841
left-news          4459
Government News    1570
US_News             783
Middle-east         778
Name: subject, dtype: int64
```

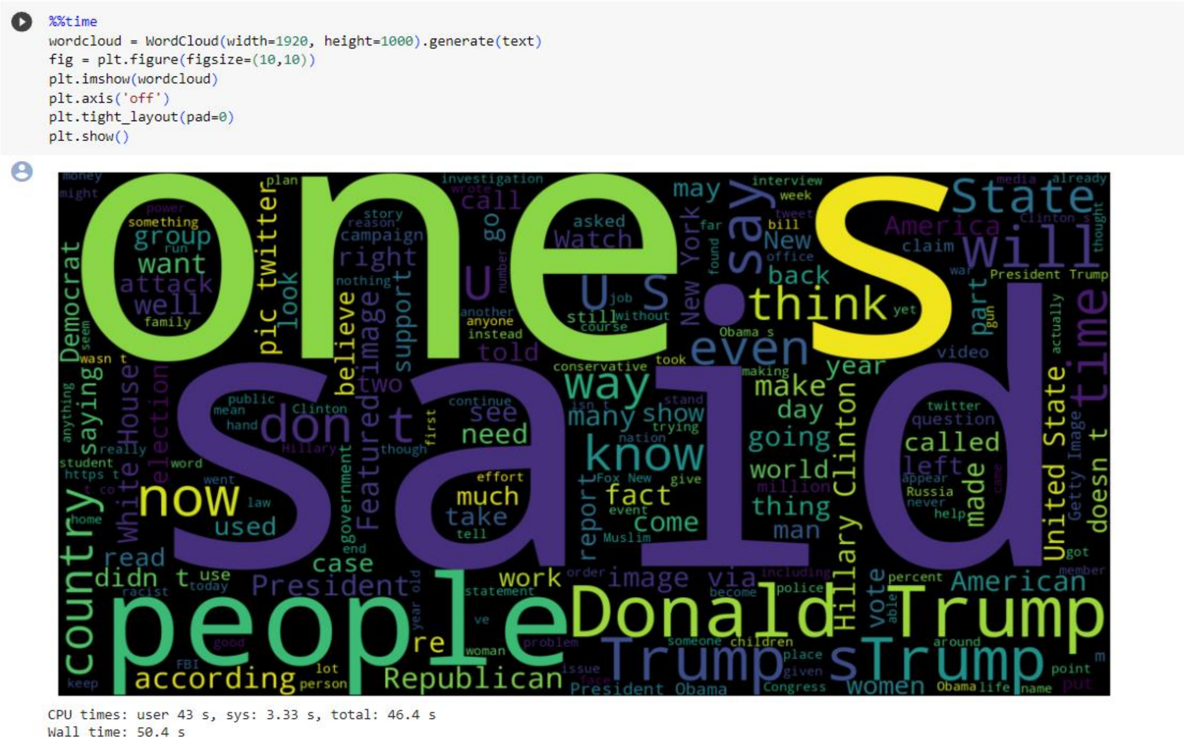
```
plt.figure(figsize=(10,6))
sns.countplot(x='subject',data=fake_news)
```

```
<Axes: xlabel='subject', ylabel='count'>
```



Here , we have used wordcloud to see that which word has mostly used for the fake news. By seeing that we can make a conclusion that which topic(about a person, event or anything) is mostly contains fake news).We also do the same for true news.

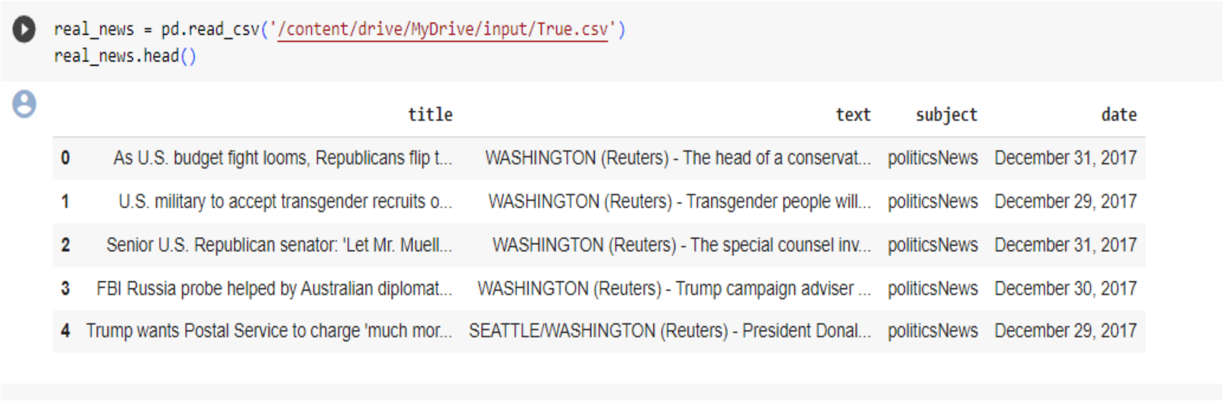
Word Cloud for Fake News:



```
%%time
wordcloud = WordCloud(width=1920, height=1000).generate(text)
fig = plt.figure(figsize=(10,10))
plt.imshow(wordcloud)
plt.axis('off')
plt.tight_layout(pad=0)
plt.show()
```

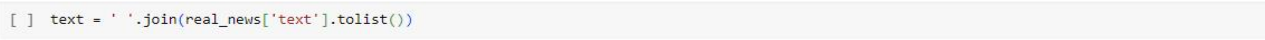
```
CPU times: user 43 s, sys: 3.33 s, total: 46.4 s
Wall time: 50.4 s
```

Word cloud for True News:

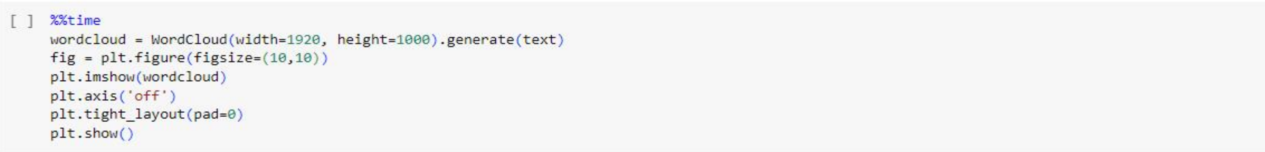


```
real_news = pd.read_csv('/content/drive/MyDrive/input/True.csv')
real_news.head()
```

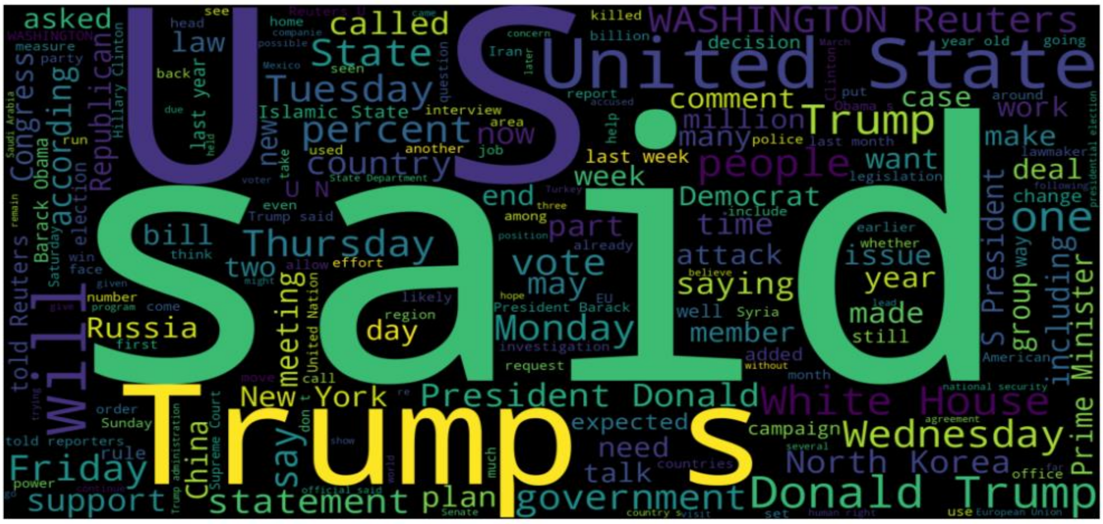
	title	text	subject	date
0	As U.S. budget fight looms, Republicans flip t...	WASHINGTON (Reuters) - The head of a conservat...	politicsNews	December 31, 2017
1	U.S. military to accept transgender recruits o...	WASHINGTON (Reuters) - Transgender people will...	politicsNews	December 29, 2017
2	Senior U.S. Republican senator: 'Let Mr. Muell...	WASHINGTON (Reuters) - The special counsel inv...	politicsNews	December 31, 2017
3	FBI Russia probe helped by Australian diplomat...	WASHINGTON (Reuters) - Trump campaign adviser ...	politicsNews	December 30, 2017
4	Trump wants Postal Service to charge 'much mor...	SEATTLE/WASHINGTON (Reuters) - President Donal...	politicsNews	December 29, 2017



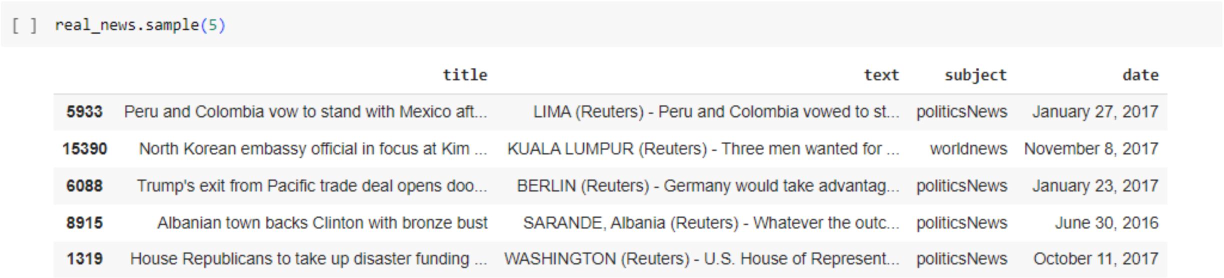
```
[ ]  text = ' '.join(real_news['text'].tolist())
```



```
[ ]  %%time
     wordcloud = WordCloud(width=1920, height=1000).generate(text)
     fig = plt.figure(figsize=(10,10))
     plt.imshow(wordcloud)
     plt.axis('off')
     plt.tight_layout(pad=0)
     plt.show()
```



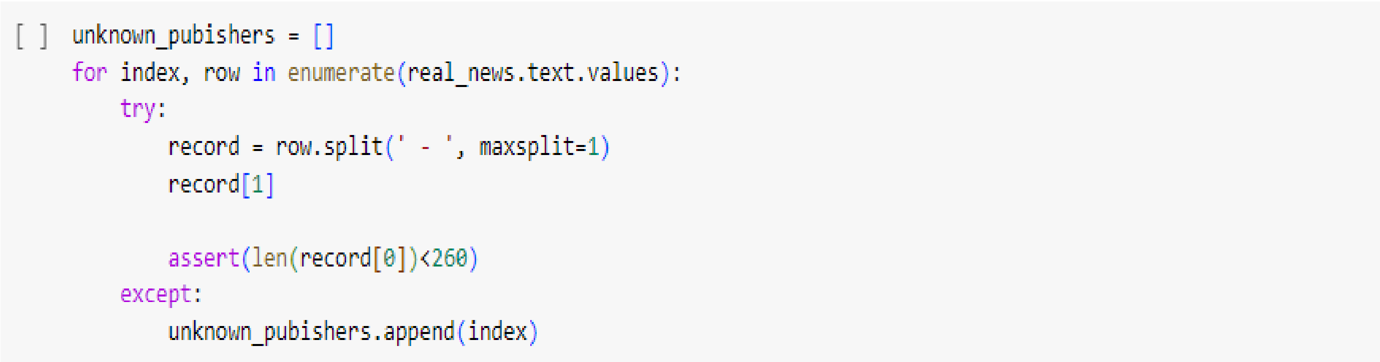
```
CPU times: user 38.5 s, sys: 2.83 s, total: 41.3 s
Wall time: 41.4 s
```



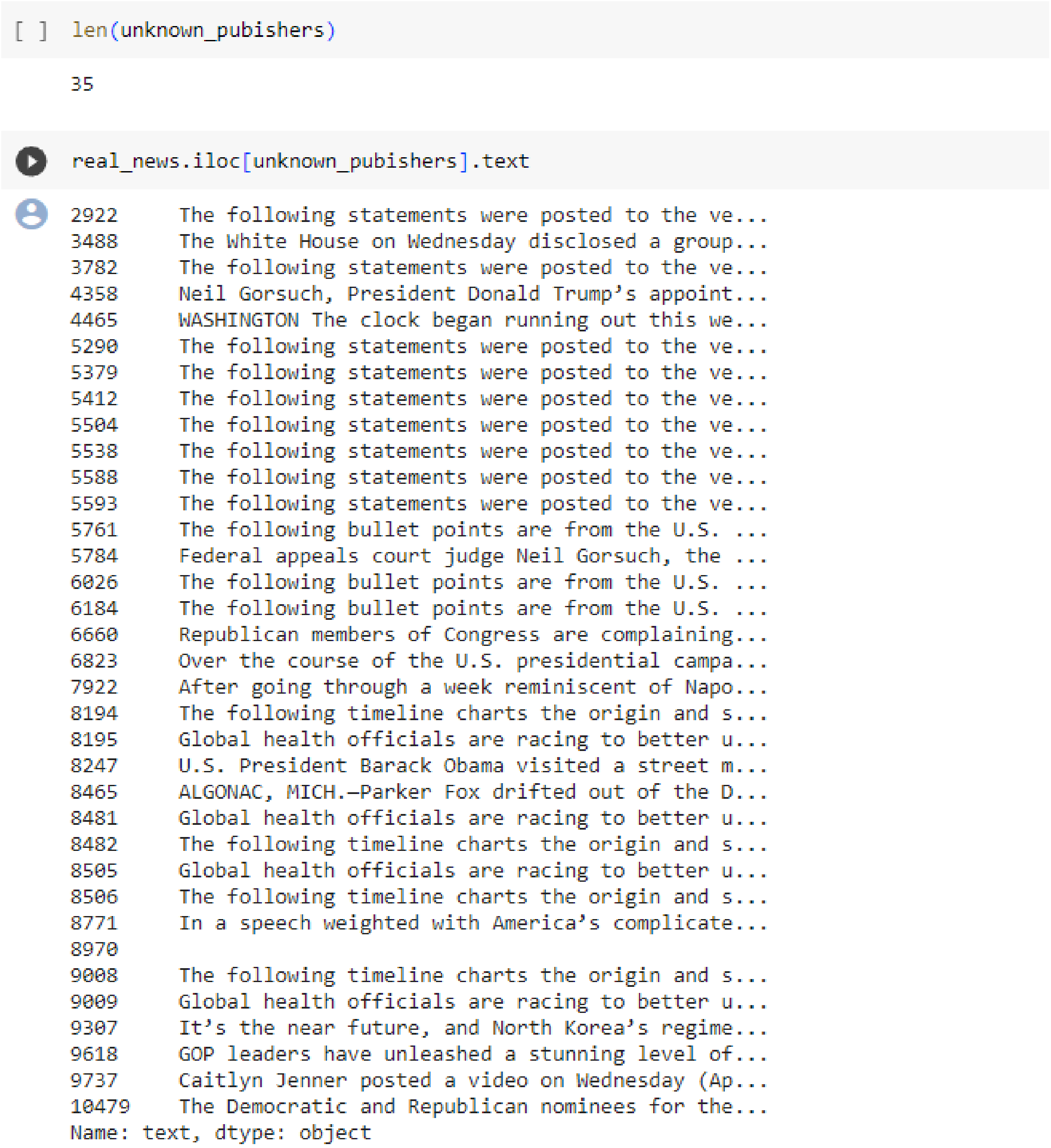
```
[ ]  real_news.sample(5)
```

	title	text	subject	date
5933	Peru and Colombia vow to stand with Mexico aft...	LIMA (Reuters) - Peru and Colombia vowed to st...	politicsNews	January 27, 2017
15390	North Korean embassy official in focus at Kim ...	KUALA LUMPUR (Reuters) - Three men wanted for ...	worldnews	November 8, 2017
6088	Trump's exit from Pacific trade deal opens doo...	BERLIN (Reuters) - Germany would take advantag...	politicsNews	January 23, 2017
8915	Albanian town backs Clinton with bronze bust	SARANDE, Albania (Reuters) - Whatever the outc...	politicsNews	June 30, 2016
1319	House Republicans to take up disaster funding ...	WASHINGTON (Reuters) - U.S. House of Represent...	politicsNews	October 11, 2017

Let’s create a list of news lists in real_news.csv with unknown publishers by using the following code snippets



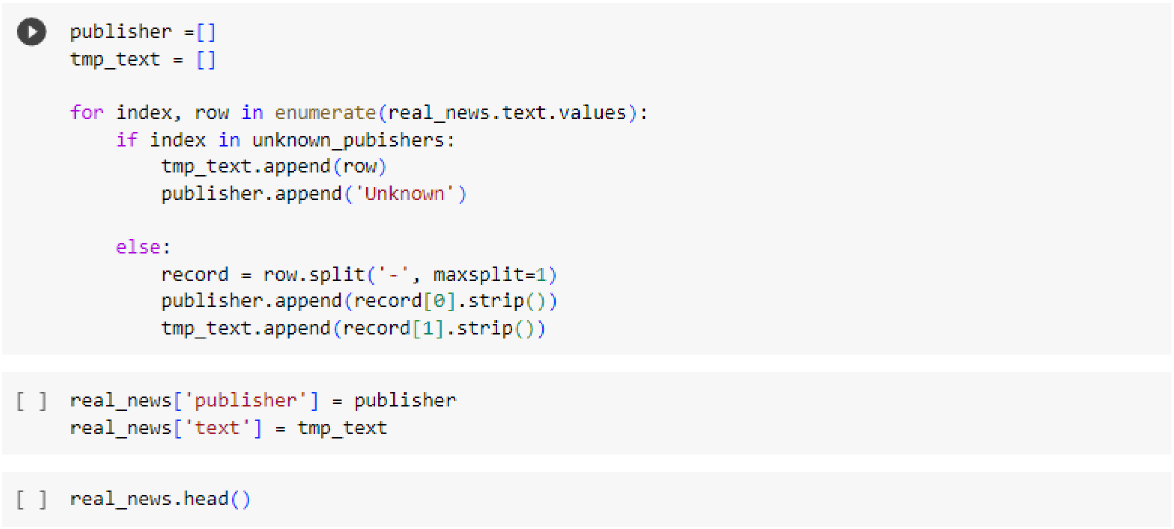
```
[ ]  unknown_pubishers = []
     for index, row in enumerate(real_news.text.values):
         try:
             record = row.split(' - ', maxsplit=1)
             record[1]

             assert(len(record[0])<260)
         except:
             unknown_pubishers.append(index)
```



```
[ ]  len(unknown_pubishers)

     35
```

```
real_news.iloc[unknown_pubishers].text
```

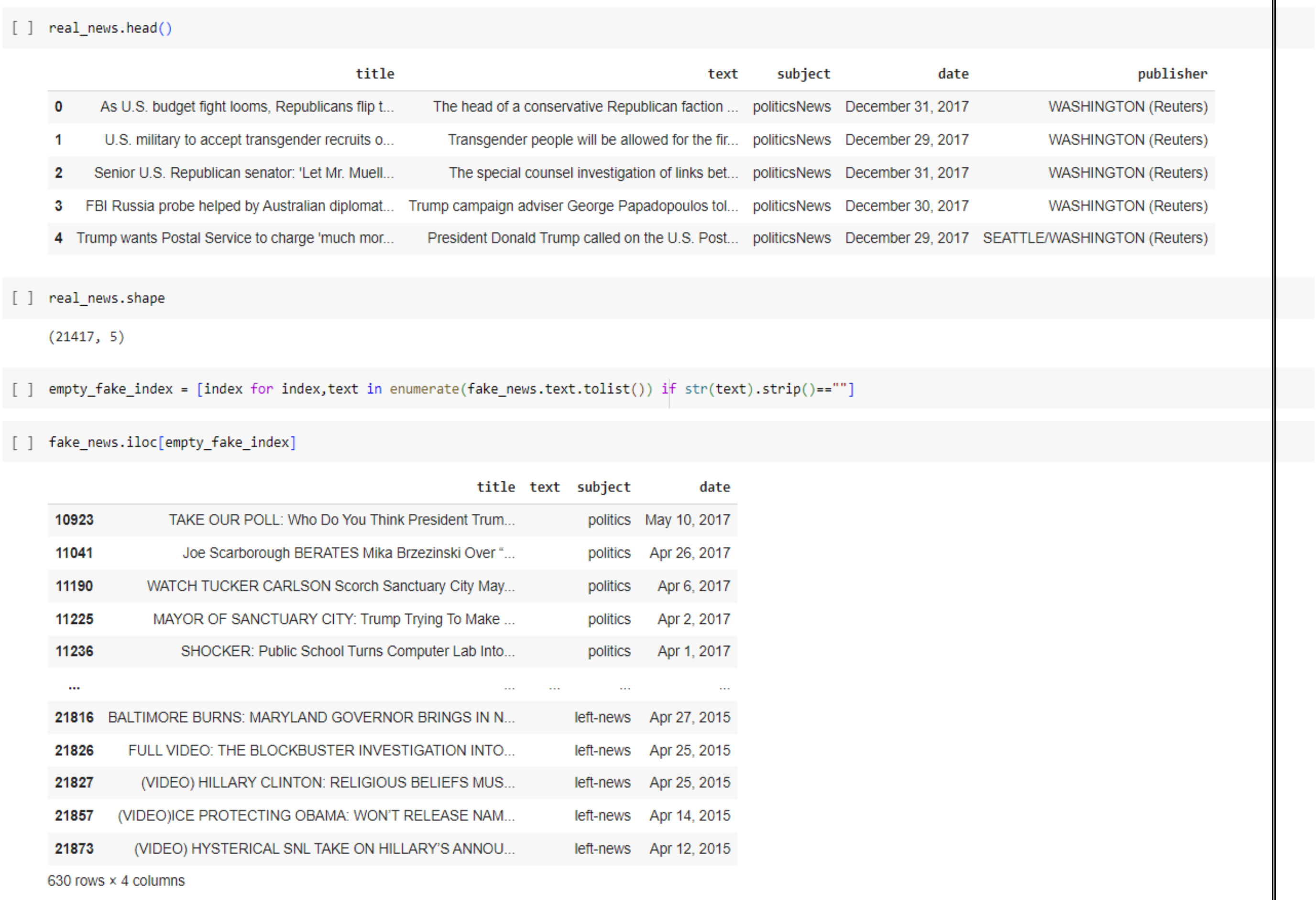
```
2922     The following statements were posted to the ve...
3488     The White House on Wednesday disclosed a group...
3782     The following statements were posted to the ve...
4358     Neil Gorsuch, President Donald Trump’s appoint...
4465     WASHINGTON The clock began running out this we...
5290     The following statements were posted to the ve...
5379     The following statements were posted to the ve...
5412     The following statements were posted to the ve...
5504     The following statements were posted to the ve...
5538     The following statements were posted to the ve...
5588     The following statements were posted to the ve...
5593     The following statements were posted to the ve...
5761     The following bullet points are from the U.S. ...
5784     Federal appeals court judge Neil Gorsuch, the ...
6026     The following bullet points are from the U.S. ...
6184     The following bullet points are from the U.S. ...
6660     Republican members of Congress are complaining...
6823     Over the course of the U.S. presidential campa...
7922     After going through a week reminiscent of Napo...
8194     The following timeline charts the origin and s...
8195     Global health officials are racing to better u...
8247     U.S. President Barack Obama visited a street m...
8465     ALGONAC, MICH.—Parker Fox drifted out of the D...
8481     Global health officials are racing to better u...
8482     The following timeline charts the origin and s...
8505     Global health officials are racing to better u...
8506     The following timeline charts the origin and s...
8771     In a speech weighted with America’s complicate...
8970
9008     The following timeline charts the origin and s...
9009     Global health officials are racing to better u...
9307     It’s the near future, and North Korea’s regime...
9618     GOP leaders have unleashed a stunning level of...
9737     Caitlyn Jenner posted a video on Wednesday (Ap...
10479    The Democratic and Republican nominees for the...
Name: text, dtype: object
```



```
publisher =[]
tmp_text = []

for index, row in enumerate(real_news.text.values):
    if index in unknown_pubishers:
        tmp_text.append(row)
        publisher.append('Unknown')

    else:
        record = row.split('-', maxsplit=1)
        publisher.append(record[0].strip())
        tmp_text.append(record[1].strip())
```

```
[ ] real_news['publisher'] = publisher
    real_news['text'] = tmp_text
```

```
[ ] real_news.head()
```



```
[ ] real_news.head()
```

	title	text	subject	date	publisher
0	As U.S. budget fight looms, Republicans flip t...	The head of a conservative Republican faction ...	politicsNews	December 31, 2017	WASHINGTON (Reuters)
1	U.S. military to accept transgender recruits o...	Transgender people will be allowed for the fir...	politicsNews	December 29, 2017	WASHINGTON (Reuters)
2	Senior U.S. Republican senator: 'Let Mr. Muell...	The special counsel investigation of links bet...	politicsNews	December 31, 2017	WASHINGTON (Reuters)
3	FBI Russia probe helped by Australian diplomat...	Trump campaign adviser George Papadopoulos tol...	politicsNews	December 30, 2017	WASHINGTON (Reuters)
4	Trump wants Postal Service to charge 'much mor...	President Donald Trump called on the U.S. Post...	politicsNews	December 29, 2017	SEATTLE/WASHINGTON (Reuters)

```
[ ] real_news.shape
```

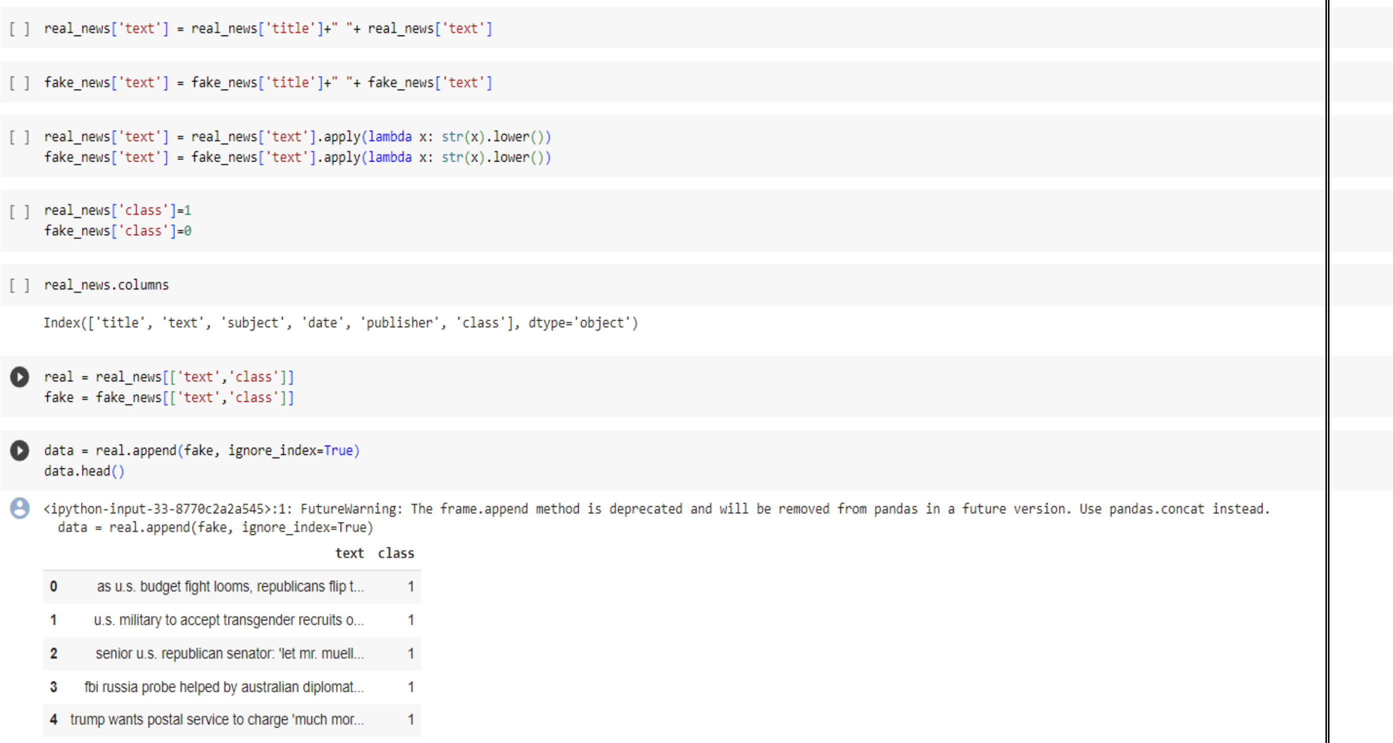
```
(21417, 5)
```

```
[ ] empty_fake_index = [index for index,text in enumerate(fake_news.text.tolist()) if str(text).strip()==""]
```

```
[ ] fake_news.iloc[empty_fake_index]
```

	title	text	subject	date
10923	TAKE OUR POLL: Who Do You Think President Trum...		politics	May 10, 2017
11041	Joe Scarborough BERATES Mika Brzezinski Over "...		politics	Apr 26, 2017
11190	WATCH TUCKER CARLSON Scorch Sanctuary City May...		politics	Apr 6, 2017
11225	MAYOR OF SANCTUARY CITY: Trump Trying To Make ...		politics	Apr 2, 2017
11236	SHOCKER: Public School Turns Computer Lab Into...		politics	Apr 1, 2017
...	...	...	...	...
21816	BALTIMORE BURNS: MARYLAND GOVERNOR BRINGS IN N...		left-news	Apr 27, 2015
21826	FULL VIDEO: THE BLOCKBUSTER INVESTIGATION INTO...		left-news	Apr 25, 2015
21827	(VIDEO) HILLARY CLINTON: RELIGIOUS BELIEFS MUS...		left-news	Apr 25, 2015
21857	(VIDEO)ICE PROTECTING OBAMA: WON'T RELEASE NAM...		left-news	Apr 14, 2015
21873	(VIDEO) HYSTERICAL SNL TAKE ON HILLARY'S ANNOU...		left-news	Apr 12, 2015

630 rows × 4 columns



```
[ ] real_news['text'] = real_news['title']+" "+ real_news['text']
```

```
[ ] fake_news['text'] = fake_news['title']+" "+ fake_news['text']
```

```
[ ] real_news['text'] = real_news['text'].apply(lambda x: str(x).lower())
    fake_news['text'] = fake_news['text'].apply(lambda x: str(x).lower())
```

```
[ ] real_news['class']=1
    fake_news['class']=0
```

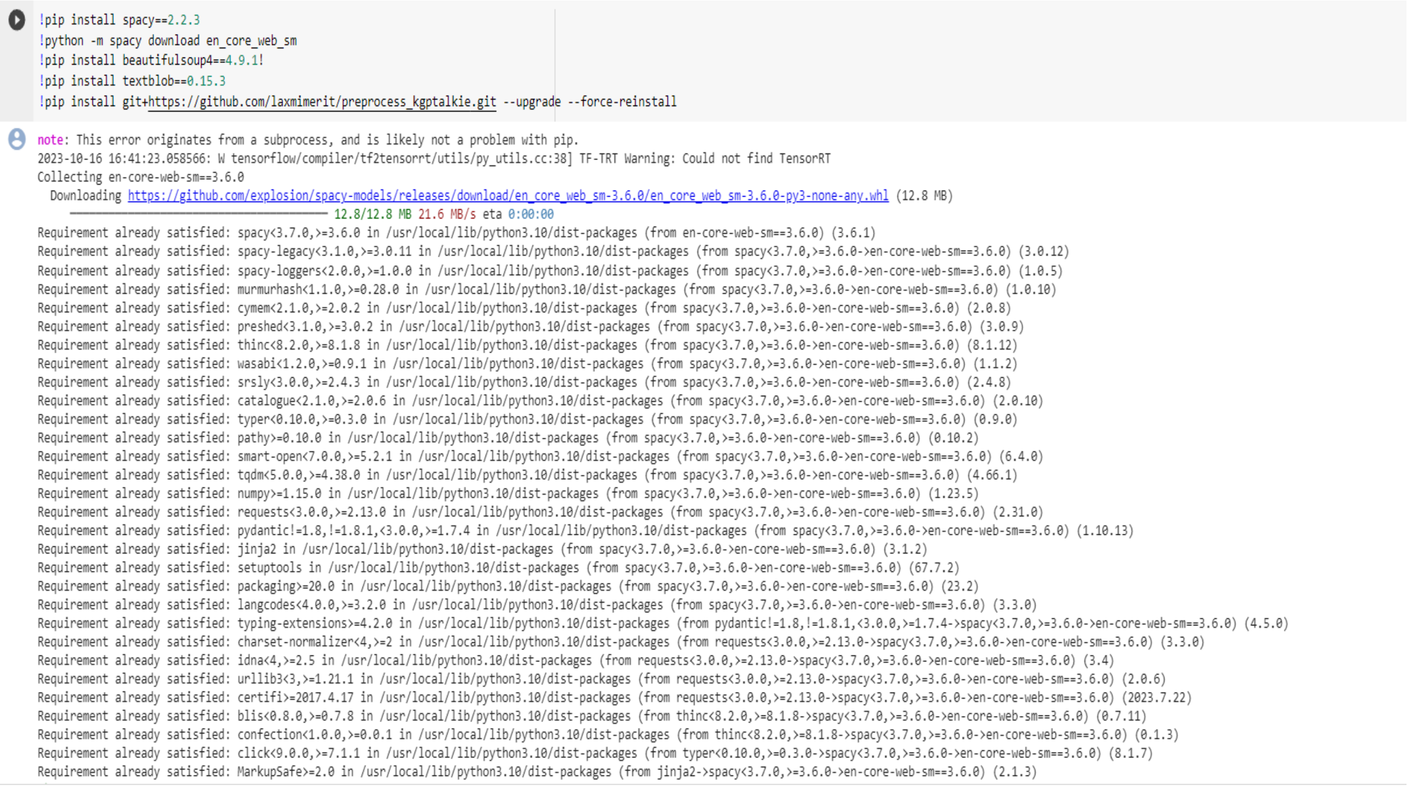
```
[ ] real_news.columns

    Index(['title', 'text', 'subject', 'date', 'publisher', 'class'], dtype='object')
```

```
real = real_news[['text','class']]
fake = fake_news[['text','class']]
```

```
data = real.append(fake, ignore_index=True)
data.head()
```

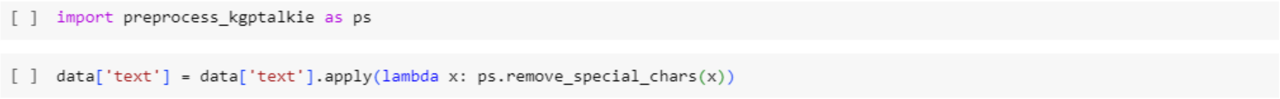
```
<ipython-input-33-8770c2a2a545>:1: FutureWarning: The frame.append method is deprecated and will be removed from pandas in a future version. Use pandas.concat instead.
  data = real.append(fake, ignore_index=True)
```

	text	class
0	as u.s. budget fight looms, republicans flip t...	1
1	u.s. military to accept transgender recruits o...	1
2	senior u.s. republican senator: 'let mr. muell...	1
3	fbi russia probe helped by australian diplomat...	1
4	trump wants postal service to charge 'much mor...	1

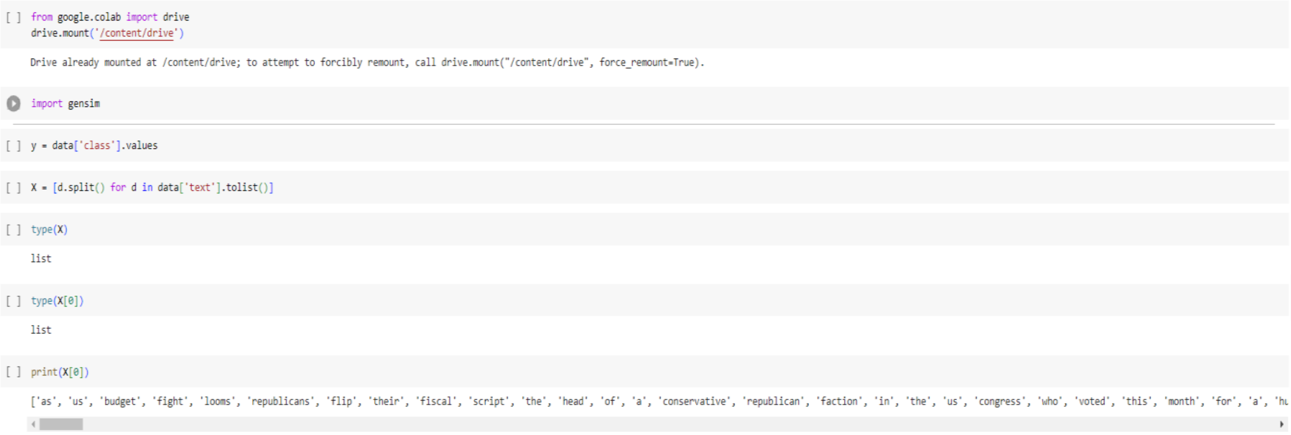


```
!pip install spacy==2.2.3
!python -m spacy download en_core_web_sm
!pip install beautifulsoup4==4.9.1!
!pip install textblob==0.15.3
!pip install git+https://github.com/laxmimerit/preprocess_kgptalkie.git --upgrade --force-reinstall
```

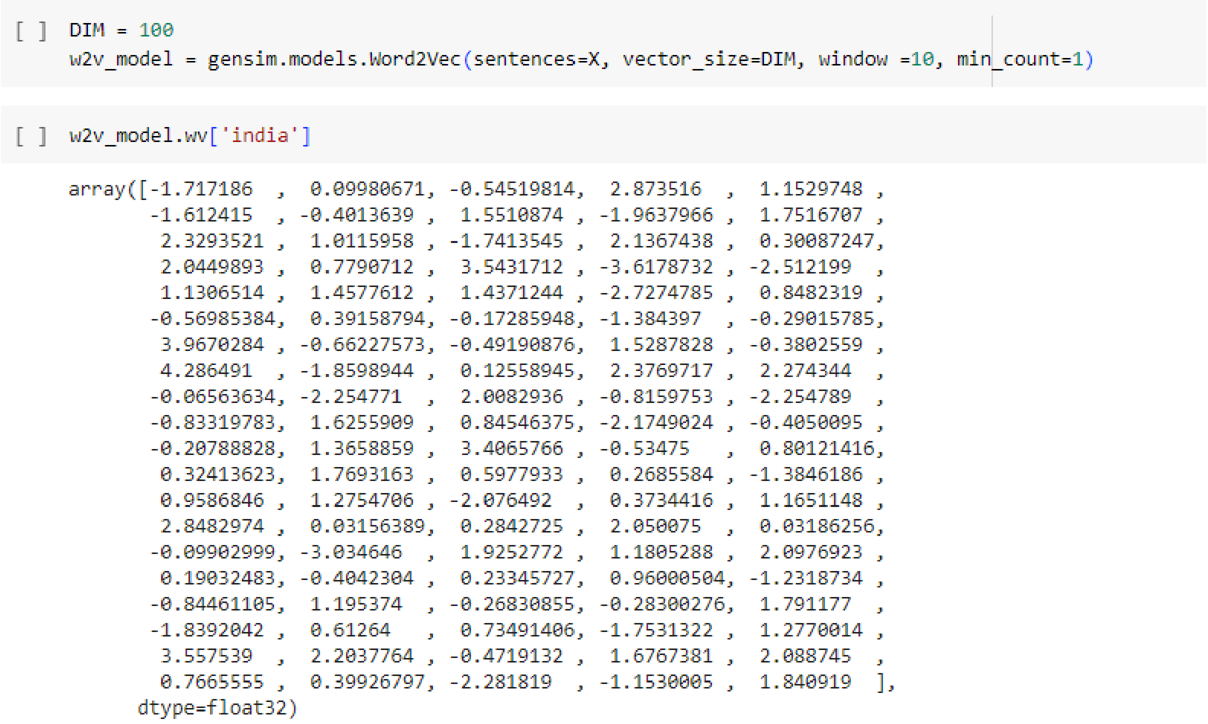
```
note: This error originates from a subprocess, and is likely not a problem with pip.
2023-10-16 16:41:23.058566: W tensorflow/compiler/tf2tensorrt/utils/py_utils.cc:38] TF-TRT Warning: Could not find TensorRT
Collecting en-core-web-sm==3.6.0
  Downloading https://github.com/explosion/spacy-models/releases/download/en_core_web_sm-3.6.0/en_core_web_sm-3.6.0-py3-none-any.whl (12.8 MB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 12.8/12.8 MB 21.6 MB/s eta 0:00:00
Requirement already satisfied: spacy<3.7.0,>=3.6.0 in /usr/local/lib/python3.10/dist-packages (from en-core-web-sm==3.6.0) (3.6.1)
Requirement already satisfied: spacy-legacy<3.1.0,>=3.0.11 in /usr/local/lib/python3.10/dist-packages (from spacy<3.7.0,>=3.6.0->en-core-web-sm==3.6.0) (3.0.12)
Requirement already satisfied: spacy-loggers<2.0.0,>=1.0.0 in /usr/local/lib/python3.10/dist-packages (from spacy<3.7.0,>=3.6.0->en-core-web-sm==3.6.0) (1.0.5)
Requirement already satisfied: murmurhash<1.1.0,>=0.28.0 in /usr/local/lib/python3.10/dist-packages (from spacy<3.7.0,>=3.6.0->en-core-web-sm==3.6.0) (1.0.10)
Requirement already satisfied: cymem<2.1.0,>=2.0.2 in /usr/local/lib/python3.10/dist-packages (from spacy<3.7.0,>=3.6.0->en-core-web-sm==3.6.0) (2.0.8)
Requirement already satisfied: preshed<3.1.0,>=3.0.2 in /usr/local/lib/python3.10/dist-packages (from spacy<3.7.0,>=3.6.0->en-core-web-sm==3.6.0) (3.0.9)
Requirement already satisfied: thinc<8.2.0,>=8.1.8 in /usr/local/lib/python3.10/dist-packages (from spacy<3.7.0,>=3.6.0->en-core-web-sm==3.6.0) (8.1.12)
Requirement already satisfied: wasabi<1.2.0,>=0.9.1 in /usr/local/lib/python3.10/dist-packages (from spacy<3.7.0,>=3.6.0->en-core-web-sm==3.6.0) (1.1.2)
Requirement already satisfied: srsly<3.0.0,>=2.4.3 in /usr/local/lib/python3.10/dist-packages (from spacy<3.7.0,>=3.6.0->en-core-web-sm==3.6.0) (2.4.8)
Requirement already satisfied: catalogue<2.1.0,>=2.0.6 in /usr/local/lib/python3.10/dist-packages (from spacy<3.7.0,>=3.6.0->en-core-web-sm==3.6.0) (2.0.10)
Requirement already satisfied: typer<0.10.0,>=0.3.0 in /usr/local/lib/python3.10/dist-packages (from spacy<3.7.0,>=3.6.0->en-core-web-sm==3.6.0) (0.9.0)
Requirement already satisfied: pathy>=0.10.0 in /usr/local/lib/python3.10/dist-packages (from spacy<3.7.0,>=3.6.0->en-core-web-sm==3.6.0) (0.10.2)
Requirement already satisfied: smart-open<7.0.0,>=5.2.1 in /usr/local/lib/python3.10/dist-packages (from spacy<3.7.0,>=3.6.0->en-core-web-sm==3.6.0) (6.4.0)
Requirement already satisfied: tqdm<5.0.0,>=4.38.0 in /usr/local/lib/python3.10/dist-packages (from spacy<3.7.0,>=3.6.0->en-core-web-sm==3.6.0) (4.66.1)
Requirement already satisfied: numpy>=1.15.0 in /usr/local/lib/python3.10/dist-packages (from spacy<3.7.0,>=3.6.0->en-core-web-sm==3.6.0) (1.23.5)
Requirement already satisfied: requests<3.0.0,>=2.13.0 in /usr/local/lib/python3.10/dist-packages (from spacy<3.7.0,>=3.6.0->en-core-web-sm==3.6.0) (2.31.0)
Requirement already satisfied: pydantic!=1.8,!=1.8.1,<3.0.0,>=1.7.4 in /usr/local/lib/python3.10/dist-packages (from spacy<3.7.0,>=3.6.0->en-core-web-sm==3.6.0) (1.10.13)
Requirement already satisfied: jinja2 in /usr/local/lib/python3.10/dist-packages (from spacy<3.7.0,>=3.6.0->en-core-web-sm==3.6.0) (3.1.2)
Requirement already satisfied: setuptools in /usr/local/lib/python3.10/dist-packages (from spacy<3.7.0,>=3.6.0->en-core-web-sm==3.6.0) (67.7.2)
Requirement already satisfied: packaging>=20.0 in /usr/local/lib/python3.10/dist-packages (from spacy<3.7.0,>=3.6.0->en-core-web-sm==3.6.0) (23.2)
Requirement already satisfied: langcodes<4.0.0,>=3.2.0 in /usr/local/lib/python3.10/dist-packages (from spacy<3.7.0,>=3.6.0->en-core-web-sm==3.6.0) (3.3.0)
Requirement already satisfied: typing-extensions>=4.2.0 in /usr/local/lib/python3.10/dist-packages (from pydantic!=1.8,!=1.8.1,<3.0.0,>=1.7.4->spacy<3.7.0,>=3.6.0->en-core-web-sm==3.6.0) (4.5.0)
Requirement already satisfied: charset-normalizer<4,>=2 in /usr/local/lib/python3.10/dist-packages (from requests<3.0.0,>=2.13.0->spacy<3.7.0,>=3.6.0->en-core-web-sm==3.6.0) (3.3.0)
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.10/dist-packages (from requests<3.0.0,>=2.13.0->spacy<3.7.0,>=3.6.0->en-core-web-sm==3.6.0) (3.4)
Requirement already satisfied: urllib3<3,>=1.21.1 in /usr/local/lib/python3.10/dist-packages (from requests<3.0.0,>=2.13.0->spacy<3.7.0,>=3.6.0->en-core-web-sm==3.6.0) (2.0.6)
Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.10/dist-packages (from requests<3.0.0,>=2.13.0->spacy<3.7.0,>=3.6.0->en-core-web-sm==3.6.0) (2023.7.22)
Requirement already satisfied: blis<0.8.0,>=0.7.8 in /usr/local/lib/python3.10/dist-packages (from thinc<8.2.0,>=8.1.8->spacy<3.7.0,>=3.6.0->en-core-web-sm==3.6.0) (0.7.11)
Requirement already satisfied: confection<1.0.0,>=0.0.1 in /usr/local/lib/python3.10/dist-packages (from thinc<8.2.0,>=8.1.8->spacy<3.7.0,>=3.6.0->en-core-web-sm==3.6.0) (0.1.3)
Requirement already satisfied: click<9.0.0,>=7.1.1 in /usr/local/lib/python3.10/dist-packages (from typer<0.10.0,>=0.3.0->spacy<3.7.0,>=3.6.0->en-core-web-sm==3.6.0) (8.1.7)
Requirement already satisfied: MarkupSafe>=2.0 in /usr/local/lib/python3.10/dist-packages (from jinja2->spacy<3.7.0,>=3.6.0->en-core-web-sm==3.6.0) (2.1.3)
```



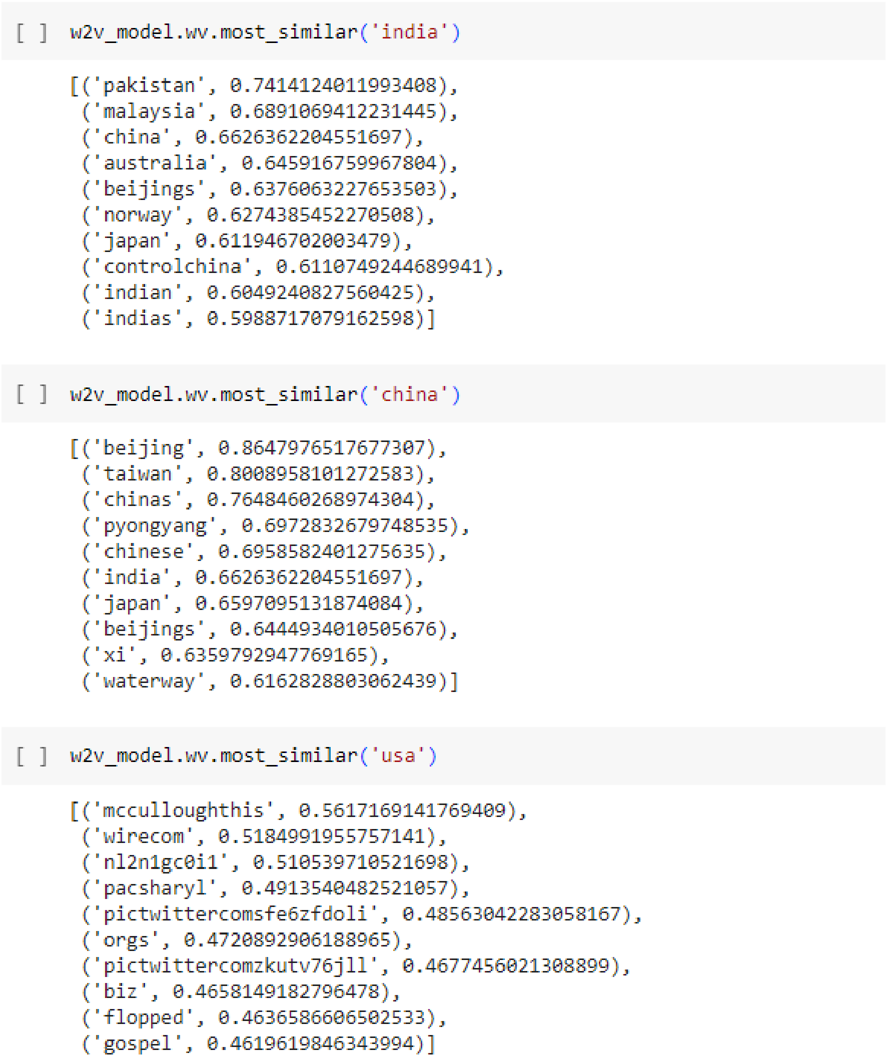
```
[ ]  import preprocess_kgptalkie as ps

[ ]  data['text'] = data['text'].apply(lambda x: ps.remove_special_chars(x))
```



```
[ ]  from google.colab import drive
     drive.mount('/content/drive')

     Drive already mounted at /content/drive; to attempt to forcibly remount, call drive.mount("/content/drive", force_remount=True).

     import gensim

[ ]  y = data['class'].values

[ ]  X = [d.split() for d in data['text'].tolist()]

[ ]  type(X)

     list

[ ]  type(X[0])

     list

[ ]  print(X[0])

     ['as', 'us', 'budget', 'fight', 'looms', 'republicans', 'flip', 'their', 'fiscal', 'script', 'the', 'head', 'of', 'a', 'conservative', 'republican', 'faction', 'in', 'the', 'us', 'congress', 'who', 'voted', 'this', 'month', 'for', 'a', 'hu
```



```
[ ]  DIM = 100
     w2v_model = gensim.models.Word2Vec(sentences=X, vector_size=DIM, window =10, min_count=1)

[ ]  w2v_model.wv['india']

     array([-1.717186  ,  0.09980671, -0.54519814,  2.873516  ,  1.1529748 ,
            -1.612415  , -0.4013639 ,  1.5510874 , -1.9637966 ,  1.7516707 ,
             2.3293521 ,  1.0115958 , -1.7413545 ,  2.1367438 ,  0.30087247,
             2.0449893 ,  0.7790712 ,  3.5431712 , -3.6178732 , -2.512199  ,
             1.1306514 ,  1.4577612 ,  1.4371244 , -2.7274785 ,  0.8482319 ,
            -0.56985384,  0.39158794, -0.17285948, -1.384397  , -0.29015785,
             3.9670284 , -0.66227573, -0.49190876,  1.5287828 , -0.3802559 ,
             4.286491  , -1.8598944 ,  0.12558945,  2.3769717 ,  2.274344  ,
            -0.06563634, -2.254771  ,  2.0082936 , -0.8159753 , -2.254789  ,
            -0.83319783,  1.6255909 ,  0.84546375, -2.1749024 , -0.4050095 ,
            -0.20788828,  1.3658859 ,  3.4065766 , -0.53475   ,  0.80121416,
             0.32413623,  1.7693163 ,  0.5977933 ,  0.2685584 , -1.3846186 ,
             0.9586846 ,  1.2754706 , -2.076492  ,  0.3734416 ,  1.1651148 ,
             2.8482974 ,  0.03156389,  0.2842725 ,  2.050075  ,  0.03186256,
            -0.09902999, -3.034646  ,  1.9252772 ,  1.1805288 ,  2.0976923 ,
             0.19032483, -0.4042304 ,  0.23345727,  0.96000504, -1.2318734 ,
            -0.84461105,  1.195374  , -0.26830855, -0.28300276,  1.791177  ,
            -1.8392042 ,  0.61264   ,  0.73491406, -1.7531322 ,  1.2770014 ,
             3.557539  ,  2.2037764 , -0.4719132 ,  1.6767381 ,  2.088745  ,
             0.7665555 ,  0.39926797, -2.281819  , -1.1530005 ,  1.840919  ],
           dtype=float32)
```



```
[ ]  w2v_model.wv.most_similar('india')
```

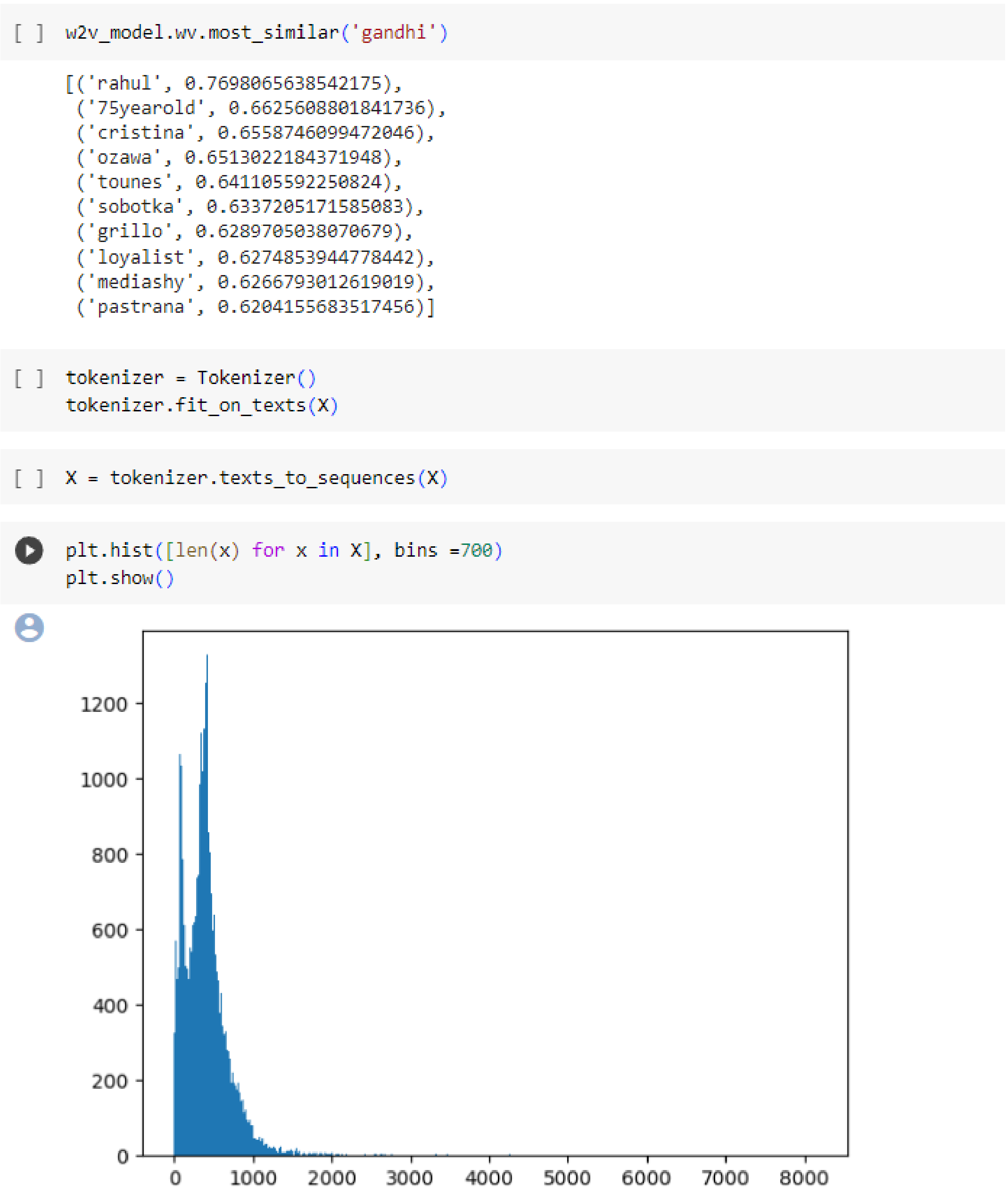
```
[('pakistan', 0.7414124011993408),
 ('malaysia', 0.6891069412231445),
 ('china', 0.662636220451697),
 ('australia', 0.645916759967804),
 ('beijings', 0.6376063227653503),
 ('norway', 0.6274385452270508),
 ('japan', 0.611946702003479),
 ('controlchina', 0.6110749244689941),
 ('indian', 0.6049240827560425),
 ('indias', 0.5988717079162598)]
```

```
[ ]  w2v_model.wv.most_similar('china')
```

```
[('beijing', 0.8647976517677307),
 ('taiwan', 0.8008958101272583),
 ('chinas', 0.7648460268974304),
 ('pyongyang', 0.6972832679748535),
 ('chinese', 0.6958582401275635),
 ('india', 0.662636220451697),
 ('japan', 0.6597095131874084),
 ('beijings', 0.6444934010505676),
 ('xi', 0.635979294776165),
 ('waterway', 0.6162828803062439)]
```

```
[ ]  w2v_model.wv.most_similar('usa')
```

```
[('mccullougthis', 0.5617169141769409),
 ('wirecom', 0.5184991955757141),
 ('nl2n1gc0i1', 0.510539710521698),
 ('pacsharyl', 0.4913540482521057),
 ('pictwittercomsfe6zfdoli', 0.48563042283058167),
 ('orgs', 0.472089290618896),
 ('pictwittercomzkutv76jll', 0.4677456021308899),
 ('biz', 0.4658149182796478),
 ('flopped', 0.4636586606502533),
 ('gospel', 0.4619619846343994)]
```

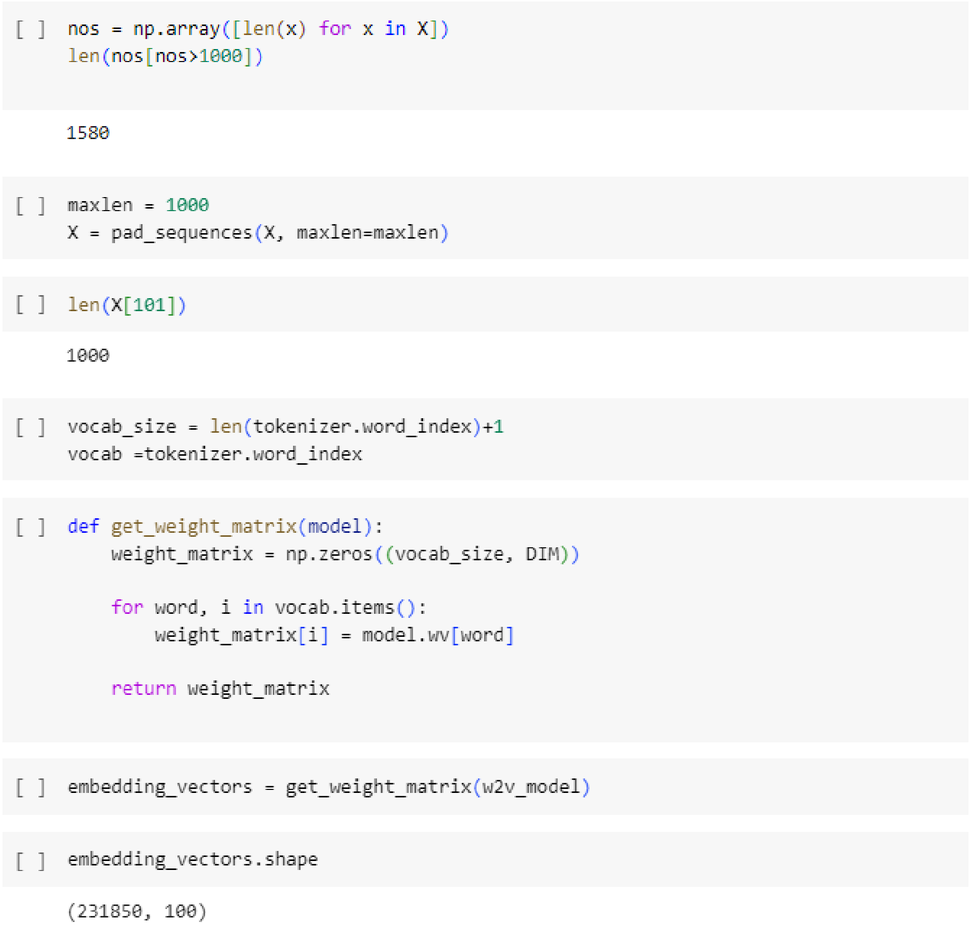


```
[ ] w2v_model.wv.most_similar('gandhi')

    [('rahul', 0.7698065638542175),
     ('75yearold', 0.6625608801841736),
     ('cristina', 0.6558746099472046),
     ('ozawa', 0.651302218437194 8),
     ('tounes', 0.641105592250824),
     ('sobotka', 0.6337205171585083),
     ('grillo', 0.6289705038070679),
     ('loyalist', 0.6274853944778442),
     ('mediashy', 0.6266793012619019),
     ('pastrana', 0.6204155683517456)]
```

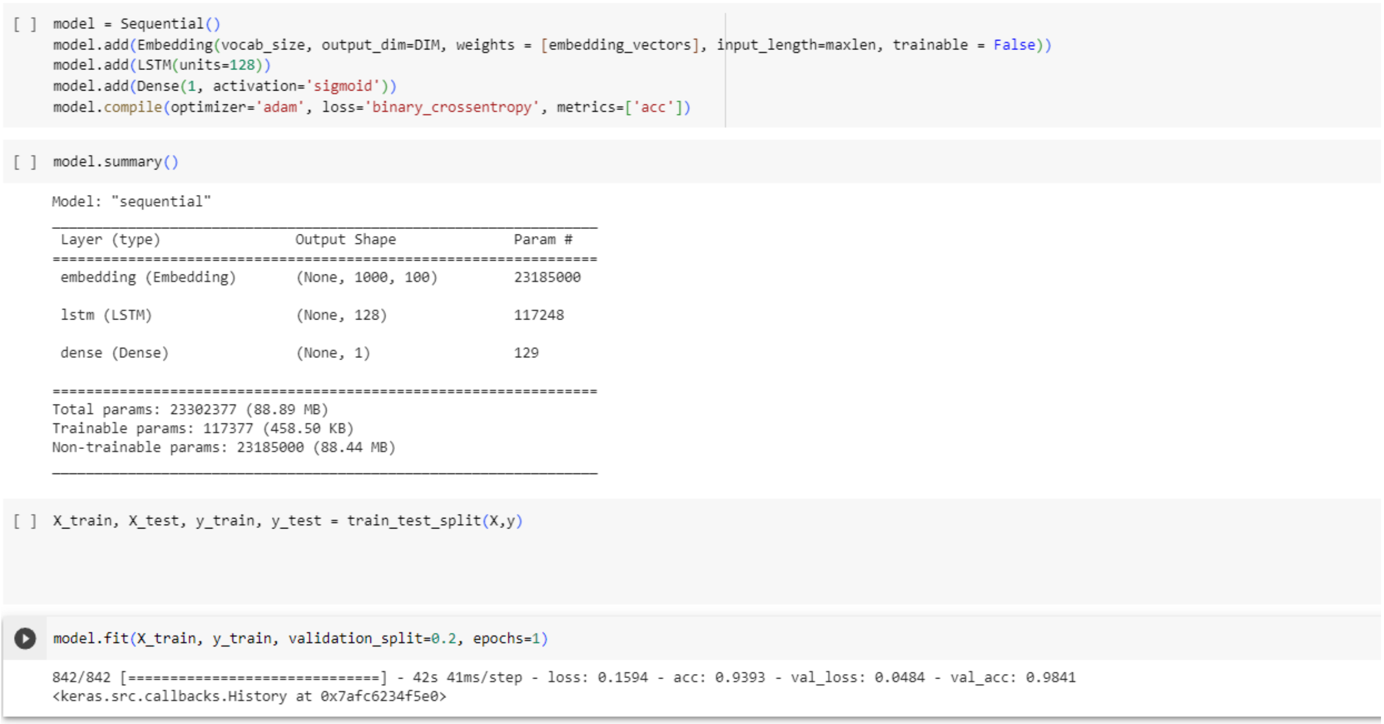
```
[ ] tokenizer = Tokenizer()
    tokenizer.fit_on_texts(X)
```

```
[ ] X = tokenizer.texts_to_sequences(X)
```

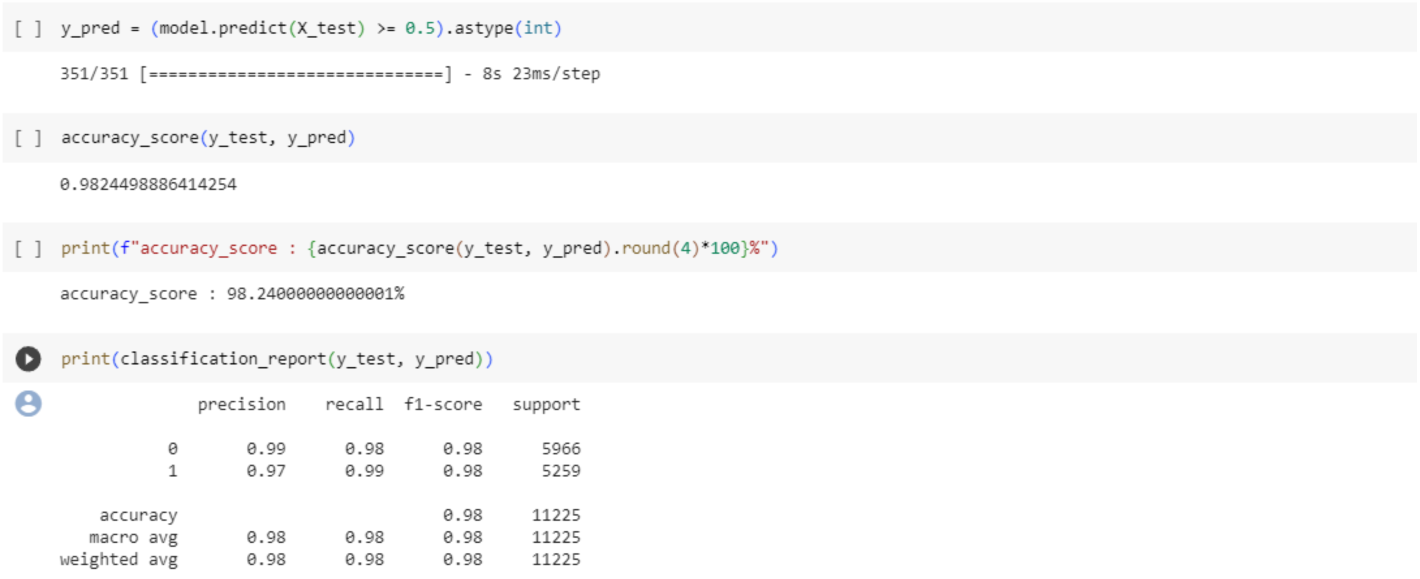
```
plt.hist([len(x) for x in X], bins =700)
plt.show()
```



```
[ ]  nos = np.array([len(x) for x in X])
     len(nos[nos>1000])

     1580

[ ]  maxlen = 1000
     X = pad_sequences(X, maxlen=maxlen)

[ ]  len(X[101])

     1000

[ ]  vocab_size = len(tokenizer.word_index)+1
     vocab =tokenizer.word_index

[ ]  def get_weight_matrix(model):
         weight_matrix = np.zeros((vocab_size, DIM))

         for word, i in vocab.items():
             weight_matrix[i] = model.wv[word]

         return weight_matrix

[ ]  embedding_vectors = get_weight_matrix(w2v_model)

[ ]  embedding_vectors.shape

     (231850, 100)
```



```
[ ]  model = Sequential()
     model.add(Embedding(vocab_size, output_dim=DIM, weights = [embedding_vectors], input_length=maxlen, trainable = False))
     model.add(LSTM(units=128))
     model.add(Dense(1, activation='sigmoid'))
     model.compile(optimizer='adam', loss='binary_crossentropy', metrics=['acc'])

[ ]  model.summary()

     Model: "sequential"
     _________________________________________________________________
      Layer (type)                Output Shape              Param #   
     =================================================================
      embedding (Embedding)       (None, 1000, 100)         23185000  
                                                                      
      lstm (LSTM)                 (None, 128)               117248    
                                                                      
      dense (Dense)               (None, 1)                 129       
                                                                      
     =================================================================
     Total params: 23302377 (88.89 MB)
     Trainable params: 117377 (458.50 KB)
     Non-trainable params: 23185000 (88.44 MB)
     _________________________________________________________________

[ ]  X_train, X_test, y_train, y_test = train_test_split(X,y)

     model.fit(X_train, y_train, validation_split=0.2, epochs=1)

     842/842 [==============================] - 42s 41ms/step - loss: 0.1594 - acc: 0.9393 - val_loss: 0.0484 - val_acc: 0.9841
     <keras.src.callbacks.History at 0x7afc6234f5e0>
```



```
y_pred = (model.predict(X_test) >= 0.5).astype(int)
```

```
351/351 [==============================] - 8s 23ms/step
```

```
accuracy_score(y_test, y_pred)
```

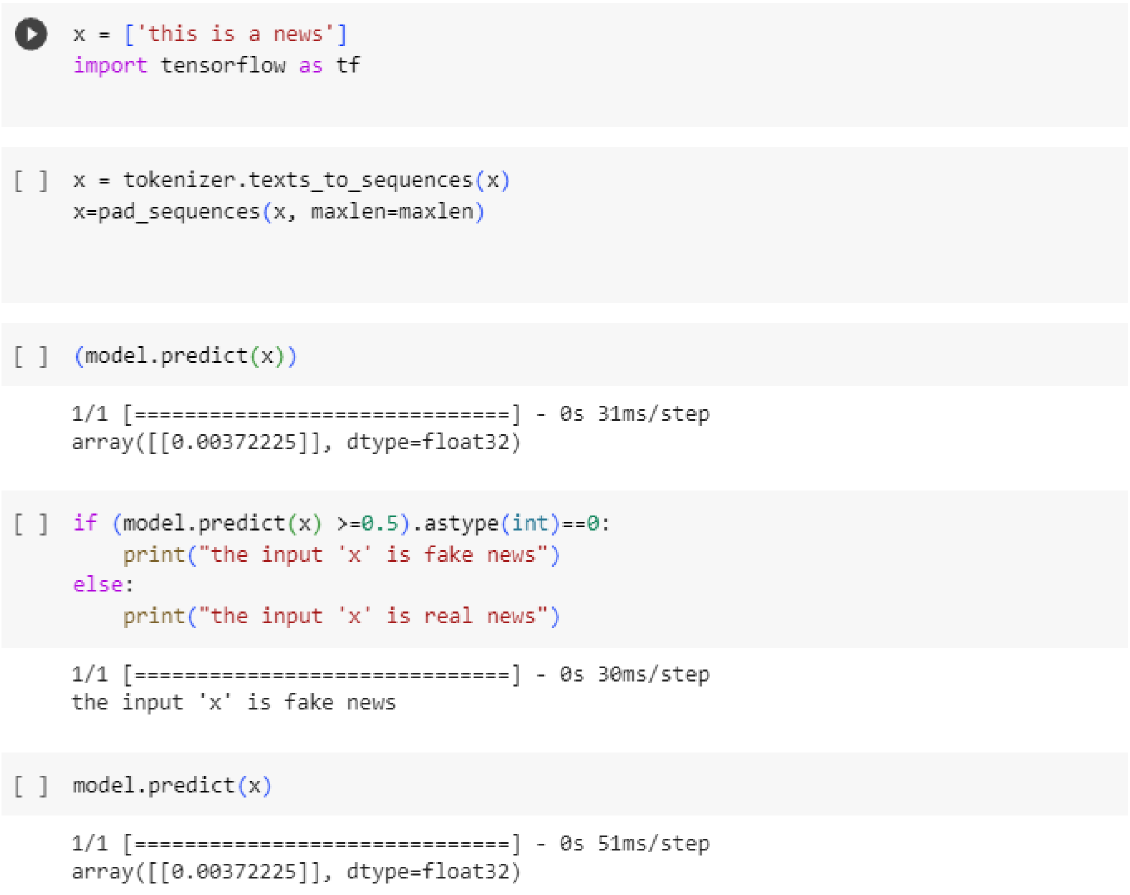
```
0.9824498886414254
```

```
print(f"accuracy_score : {accuracy_score(y_test, y_pred).round(4)*100}%")
```

```
accuracy_score : 98.24000000000001%
```

```
print(classification_report(y_test, y_pred))
```

```
              precision    recall  f1-score   support

           0       0.99      0.98      0.98      5966
           1       0.97      0.99      0.98      5259

    accuracy                           0.98     11225
   macro avg       0.98      0.98      0.98     11225
weighted avg       0.98      0.98      0.98     11225
```



```
x = ['this is a news']
import tensorflow as tf
```

```
x = tokenizer.texts_to_sequences(x)
x=pad_sequences(x, maxlen=maxlen)
```

```
(model.predict(x))
```

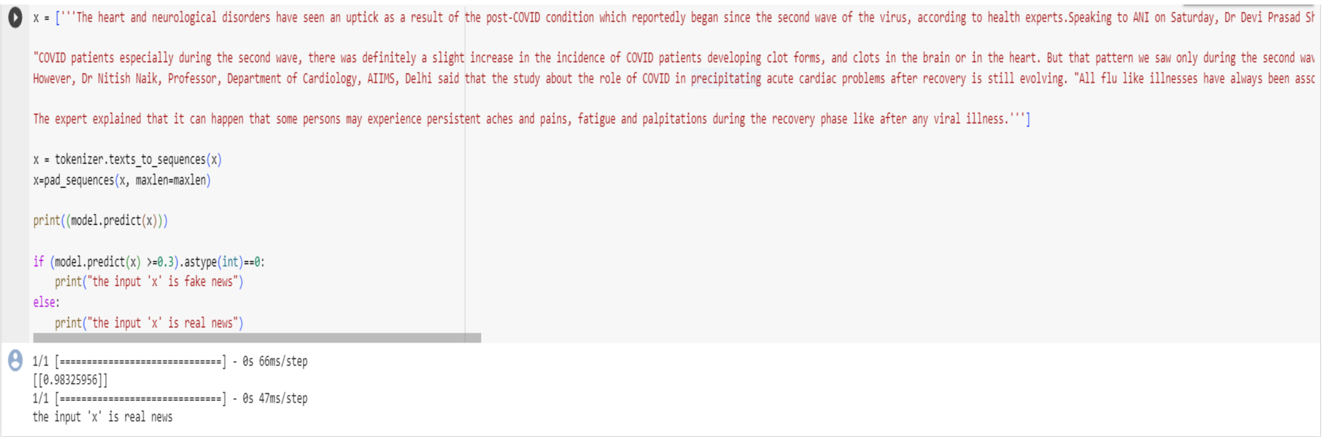
```
1/1 [==============================] - 0s 31ms/step
array([[0.00372225]], dtype=float32)
```

```
if (model.predict(x) >=0.5).astype(int)==0:
    print("the input 'x' is fake news")
else:
    print("the input 'x' is real news")
```

```
1/1 [==============================] - 0s 30ms/step
the input 'x' is fake news
```

```
model.predict(x)
```

```
1/1 [==============================] - 0s 51ms/step
array([[0.00372225]], dtype=float32)
```



```
x = ['''The heart and neurological disorders have seen an uptick as a result of the post-COVID condition which reportedly began since the second wave of the virus, according to health experts.Speaking to ANI on Saturday, Dr Devi Prasad Sh

"COVID patients especially during the second wave, there was definitely a slight increase in the incidence of COVID patients developing clot forms, and clots in the brain or in the heart. But that pattern we saw only during the second wav
However, Dr Nitish Naik, Professor, Department of Cardiology, AIIMS, Delhi said that the study about the role of COVID in precipitating acute cardiac problems after recovery is still evolving. "All flu like illnesses have always been assc

The expert explained that it can happen that some persons may experience persistent aches and pains, fatigue and palpitations during the recovery phase like after any viral illness.''']

x = tokenizer.texts_to_sequences(x)
x=pad_sequences(x, maxlen=maxlen)

print((model.predict(x)))

if (model.predict(x) >=0.3).astype(int)==0:
    print("the input 'x' is fake news")
else:
    print("the input 'x' is real news")
```

```
1/1 [==============================] - 0s 66ms/step
[[0.98325956]]
1/1 [==============================] - 0s 47ms/step
the input 'x' is real news
```

Conclusion:

In conclusion, utilizing Natural Language Processing (NLP) techniques for fake news detection has proven to be a significant advancement in combating misinformation. The model developed demonstrates the potential of machine learning in identifying deceptive content, contributing to the ongoing efforts to maintain the integrity of information online. By leveraging NLP algorithms, the accuracy and efficiency of fake news detection have been greatly enhanced, empowering users to make informed decisions and fostering a more reliable digital information ecosystem. As we move forward, continued research and development in this field will play a pivotal role in ensuring the authenticity and trustworthiness of online content, thereby promoting a healthier and more informed society.