A PROJECT REPORT

ON

FAKE NEWS DETECTION USING DEEP LEARNING TECHNIQUE (LSTM)

Submitted in partial fulfillment of the requirement of the award of

TRAINING

IN

Data Science, Machine Learning and AI using Python

Submitted By:

SANTOSH SAIKIA SAHANI (Assam Down Town University, Guwahati)

Under the guidance of

Mr. Shivam Bhatia

Table of Contents

ACKNOWLEDGEMENT

I would like to express our sincere admiration and gratitude to Assam down town University for the assistance they provided us in project contact. I would like to express my heartfelt gratitude and sincere appreciation to our respected guide Mr. Shivam Bhatia for his constant supervision, direction, and helpful recommendations throughout the duration of this project work, without which I would not have been able to complete my project successfully. I would like to thank our Dean of Engineering, for his inspiration, coordination, and assistant. I would also want to thank my sincere gratitude to the lecturers, research scholars and the lab technicians for their valuable guidance and helping attitude even in their very busy schedule. I desire to reach forth my real appreciation to the institute authority, specially the Faculties for giving us the aid and providing the essential facilities. Our gratitude and appreciation also go to our colleagues who assisted us in designing the project and to those who volunteered to assist me with their skills.

1. INTRODUCTION

Fake news is a type of news that consists of purposeful misinformation or hoaxes disseminated through traditional news media or online social media. In this study, I employed various NLP-based machine learning and deep learning algorithms, including BERT, to detect fake news from news headlines. A fake headline is a news headline that may read one way or state something as fact, but the body of the piece says something completely different. The Internet term for this form of deceptive fake news is "clickbait" – headlines that entice readers to click on the bogus news. This form of fake news is, at best, deceptive, and, at worst, false.

In this project, I used NLP to extract exciting patterns from headline text and exploratory data analysis to provide meaningful insight into fake headlines by developing intuitive features. This project comprises the following tasks:

- Using NLP for exploratory data analysis and feature engineering
- Machine Learning Modeling Using Textual Features
- LSTM (Deep Learning Modeling) with text-based characteristics
- BERT model construction using text-based features

2. PROBLEM STATEMENT

Fake news has been spreading at an alarming rate in recent years for a variety of economic and political goals due to the widespread use of online social networks. This is concerning since fake news has a variety of psychological impacts on offline society. The research firm by Gartner Research says

> BY 2022, most people in mature economics will consume more false information than true information.

Because fake news will immediately lower the value of brands, firms face a difficult burden as a result of the increased prevalence of fake news content on social media. They must not only carefully monitor what is being said about their brands, but also in what settings. By comparing the text of news articles with news headlines, we will use this project to show how the deep learning approach Long Short-Term Memory (LSTM) may be used to detect fake news.

3. ABOUT DATA

The ISOT Fake News Dataset was used for the case study's information. The dataset includes both **fake** news and **real** news content. The accurate articles were found via crawling Reuters.com, a news website, and this dataset was compiled from actual sources. The discredited websites that the false news items were gathered from were reported by ***Politifact*** (a fact-checking organization in the USA) and ***Wikipedia***. The dataset includes a variety of articles on various subjects, however, the majority of the articles covered political and global news stories.

There are two CSV files in the dataset. Over 12,600 stories from ***Reuters.com*** are in the first file, *True.csv*. The second file, *Fake.csv*, has more than 12,600 articles from various resources used by fake news outlets. . The following details are included in each article:

- Title of the article (News Headline),
- Body of the article (News Body),
- Subject, and
- Date

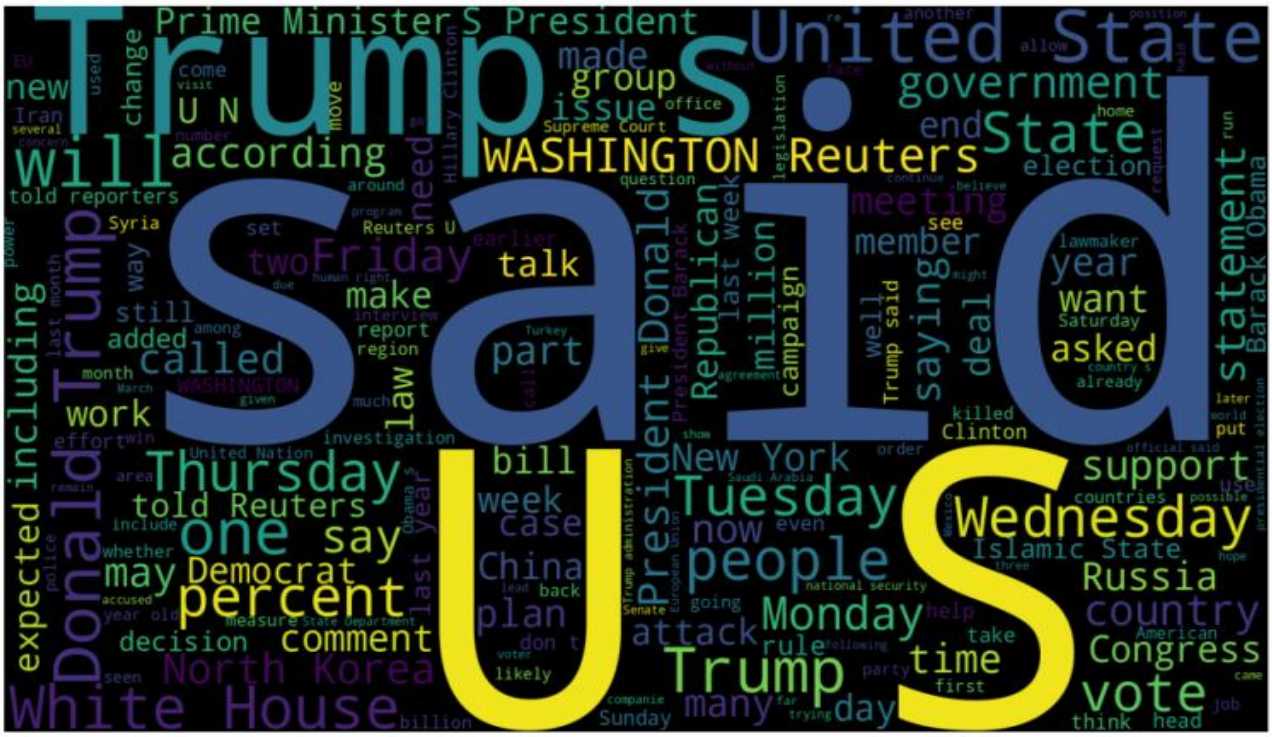
The dataset has 44898 entries in total, 21417 of which are factual news and 23481 of which are fraudulent news.

So, in the following section, we'll talk about the data pre-processing and data preparation processes needed to build **a Long Term Short Memory (LSTM)** model in Python using the **Keras** library package. You may read more about the intricate mathematics and architecture of LSTM here.

4. WORDCLOUD OF DATASET

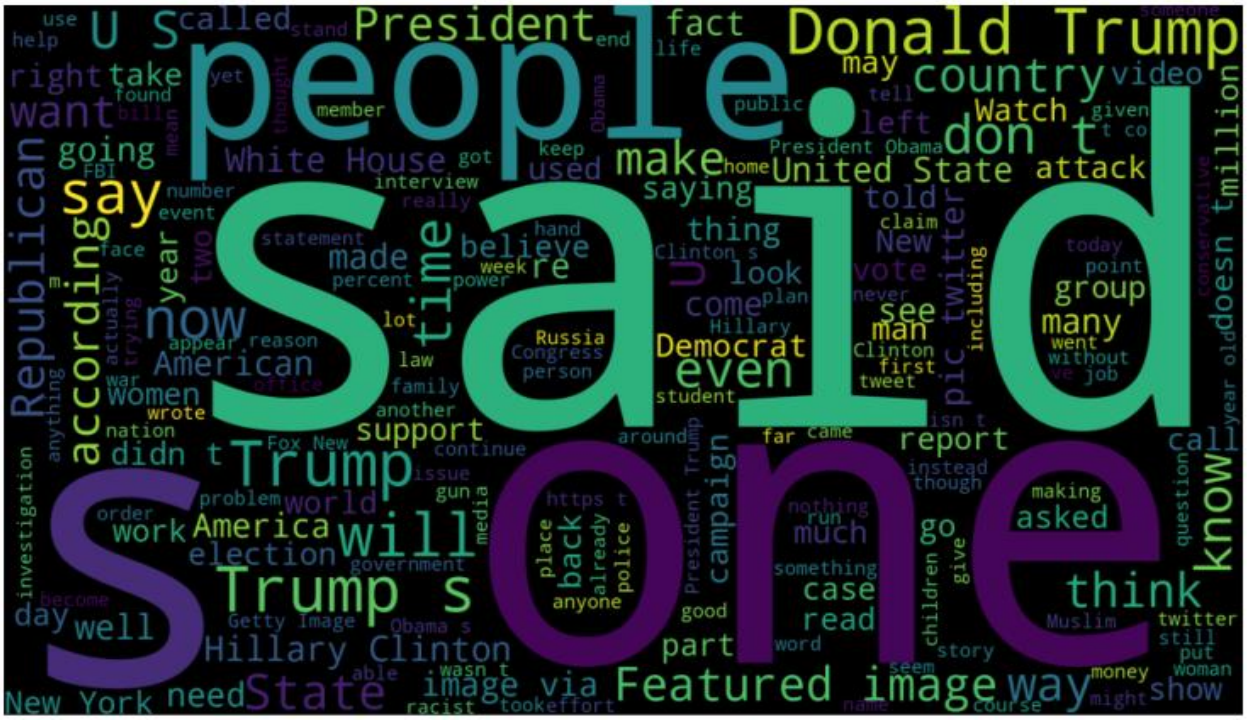
The word cloud will display the most frequently occurring words in the corpus. Words in the word cloud are sized according to how frequently they occur in the corpus. The word's font size will increase as it appears more frequently. The word clouds show that the most frequently occurring words in fake news are "video," "Obama," "Hillary Clinton," "Trump," , “President” and "Republican," whereas real news includes "Trump," "White House," "North Korea," "US," “Russia”, and other words

WordCloud of Real News



.

WordCloud of Fake News



4.1 Difference in Text

Real news seems to have source of publication which is not present in fake news set

Looking at the data

• Most of text Reuters information such as "WASHINGTON (Reuters)".

• Some text are tweets from Twitter

• Few text do not contain any publication info

4.2 Cleaning Data

Removing Reuters or Twitter Tweet information from the text Removing Reuters or Twitter Tweet information from the text

•Text can be splitted only once at which is always present after mentioning source of publication, this gives us publication part and text part

•If we do not get text part, this means publication details was given for that record

•The Twitter tweets always have same source, a long text of max 259 characters.

5. DATA PROCESSING

In this phase, the two datasets (Fake.csv and True.csv) will be read, some data cleaning will be done, the two datasets will be combined, and the resulting dataset will be shuffled.

The data processing code are shown below:

```
[22] import numpy as np
     import pandas as pd
     from collections import defaultdict
     import re
```

```
[23] import matplotlib.pyplot as plt
     import seaborn as sns
     import nltk
     from wordcloud import WordCloud
```

```
[4]  df_fake = pd.read_csv('/content/drive/MyDrive/news/Fake.csv')
```

```
[5]  df_fake.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

We will drop the other elements, such as the subject and date since we only need the title and text.

```
[29] # Dropping unneccesary features
     df_fake=df_fake.drop(['subject','date'],axis=1)
```

```
[31] # Assigning label 'FAKE' by creating target column i.e., label
     df_fake['label'] ='FAKE'
     df_fake.head()
```

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

Similar to that, in order to construct the final dataset, we will process the True dataset and combine it with the df_fake data frame. The corresponding code is provided below:

```
[48]
     df_true = pd.read_csv('/content/drive/MyDrive/news/True.csv')
     df_true=df_true.drop(['subject','date'],axis=1)
     df_true['label']='TRUE'

     data_train = pd.concat([df_true, df_fake], ignore_index=True)
     data_train.head()
```

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

So, even though the dataset has been combined, the True and Fake labels are organized in a specific order. So, in order to introduce randomness into the dataset, we must shuffle it.

```
[33] #Shuffling the dataset
     data_train=data_train.reindex(np.random.permutation(data_train.index))
     data_train.head()
```

	title	text	label
7914	U.S. Republican group hit by Russian-linked vi...	WASHINGTON (Reuters) - A U.S. Republican Party...	TRUE
34800	WIDOW OF BENGHAZI HERO In Powerful Ad To Call ...	Dr. Dorothy Woods, the widow of Benghazi hero ...	FAKE
40481	BUSTED! OHIO ATTORNEY GENERAL DISCOVERS ILLEGA...	Ohio Secretary of State Jon Husted announced a...	FAKE
10120	Plan to raise California minimum wage to $15 c...	SACRAMENTO, Calif. (Reuters) - A plan to raise...	TRUE
23999	POLL: Most Trump Supporters Say Fake 'Bowling...	Did you know that people who support Donald Tr...	FAKE

The dataset shows that True News text includes the news's source as well; for example, the third row of True News text begins with BERLIN (Reuters), whereas the fourth row of True News text begins with (Reuters) -.

Therefore, we must remove the True News text since it interferes with the model-building process even if it is not a crucial component of a false news detection model.

Therefore, we must create a look ahead regular expression to keep the content followed by "(Reuters) -". The following code is shared for regular expression-based extraction:

```
[40] import re
     # Function for extracting desired text using regex
     def extract_txt(text):
         regex = re.search(r"(?<=\(Reuters\)\s\-\s).*",text)
         if regex:
             return regex.group(0)
         return text
         #Applying regex function to retain only relevant text
     df['text_processed'] = df['text'].apply(extract_txt)
     df.head()
```

	Unnamed: 0	title	text	label	text_processed
0	7914	U.S. Republican group hit by Russian-linked vi...	WASHINGTON (Reuters) - A U.S. Republican Party...	TRUE	A U.S. Republican Party website selling campai...
1	34800	WIDOW OF BENGHAZI HERO In Powerful Ad To Call ...	Dr. Dorothy Woods, the widow of Benghazi hero ...	FAKE	Dr. Dorothy Woods, the widow of Benghazi hero ...
2	40481	BUSTED! OHIO ATTORNEY GENERAL DISCOVERS ILLEGA...	Ohio Secretary of State Jon Husted announced a...	FAKE	Ohio Secretary of State Jon Husted announced a...
3	10120	Plan to raise California minimum wage to $15 c...	SACRAMENTO, Calif. (Reuters) - A plan to raise...	TRUE	A plan to raise California's minimum wage to $...
4	23999	POLL: Most Trump Supporters Say Fake 'Bowling...	Did you know that people who support Donald Tr...	FAKE	Did you know that people who support Donald Tr...

```
[41] #Checking dataframe containing only True News
     df[df.label=="TRUE"]
```

	Unnamed: 0	title	text	label	text_processed
0	7914	U.S. Republican group hit by Russian-linked vi...	WASHINGTON (Reuters) - A U.S. Republican Party...	TRUE	A U.S. Republican Party website selling campai...
3	10120	Plan to raise California minimum wage to $15 c...	SACRAMENTO, Calif. (Reuters) - A plan to raise...	TRUE	A plan to raise California's minimum wage to $...
8	10548	Mexico says won't pay for Trump's 'terrible' b...	MEXICO CITY (Reuters) - There is no way Mexico...	TRUE	There is no way Mexico would fund Donald Trump...
9	16661	Saudis set $500 billion plan to develop border...	RIYADH (Reuters) - Saudi Arabia announced on T...	TRUE	Saudi Arabia announced on Tuesday a $500 billi...
10	9781	North Carolina lawmaker: 'we must fight to kee...	(Reuters) - A Republican candidate running to ...	TRUE	A Republican candidate running to become North...
...	...	...	...	...	...
44891	2096	After firing, Bannon returns to his 'killing m...	WASHINGTON (Reuters) - With Stephen Bannon, th...	TRUE	With Stephen Bannon, the worry always was that...
44893	8031	For some Democrats, it's voting for Clinton - ...	NEW YORK (Reuters) - One would expect voters f...	TRUE	One would expect voters from the heavily Democ...
44894	15905	Iran's Rouhani: Tehran-Moscow cooperation need...	ANKARA (Reuters) - Iran s President Hassan Rou...	TRUE	Iran s President Hassan Rouhani said on Wednes...
44895	6473	Trump says Mexico would repay U.S. funds spent...	WASHINGTON (Reuters) - U.S. President-elect Do...	TRUE	U.S. President-elect Donald Trump said on Frid...
44897	1081	Factbox: Trump on Twitter (Oct 23) - 401(k), N...	The following statements were posted to the ve...	TRUE	The following statements were posted to the ve...

21417 rows × 5 columns

Therefore, as shown in the dataset above, the text labeled as True has undergone cleaning. The next step is to build the LSTM model using this combined dataset.LSTM model with the merged dataset.

6. DATA PREPARATION AND CLEANING

The label for the target variable will be converted into a binary variable. True news will be denoted by 0, and fake news will be denoted by 1.

```
[42] # Drop extra column
     df = df.drop(['text','Unnamed: 0'],axis=1)
     df["label"] = df.label.apply(lambda  x:0 if x=='TRUE' else 1)
     df.head()
```

	title	label	text_processed
0	U.S. Republican group hit by Russian-linked vi...	0	A U.S. Republican Party website selling campai...
1	WIDOW OF BENGHAZI HERO In Powerful Ad To Call ...	1	Dr. Dorothy Woods, the widow of Benghazi hero ...
2	BUSTED! OHIO ATTORNEY GENERAL DISCOVERS ILLEGA...	1	Ohio Secretary of State Jon Husted announced a...
3	Plan to raise California minimum wage to $15 c...	0	A plan to raise California's minimum wage to $...
4	POLL: Most Trump Supporters Say Fake 'Bowling...	1	Did you know that people who support Donald Tr...

To conduct a thorough analysis of the News article, It is crucial that we combine the title and text_processed features. Moreover, it is recommended to eliminate the Unnamed: 0, title, text, and text_processed features from the dataset and solely employ the final merged news column when building the model.

```
[43] #Combining text_processed and title for creating full news article with headline
     df['final_news'] = df['title'] + " " + df['text_processed']
     df.head()
```

	title	label	text_processed	final_news
0	U.S. Republican group hit by Russian-linked vi...	0	A U.S. Republican Party website selling campai...	U.S. Republican group hit by Russian-linked vi...
1	WIDOW OF BENGHAZI HERO In Powerful Ad To Call ...	1	Dr. Dorothy Woods, the widow of Benghazi hero ...	WIDOW OF BENGHAZI HERO In Powerful Ad To Call ...
2	BUSTED! OHIO ATTORNEY GENERAL DISCOVERS ILLEGA...	1	Ohio Secretary of State Jon Husted announced a...	BUSTED! OHIO ATTORNEY GENERAL DISCOVERS ILLEGA...
3	Plan to raise California minimum wage to $15 c...	0	A plan to raise California's minimum wage to $...	Plan to raise California minimum wage to $15 c...
4	POLL: Most Trump Supporters Say Fake 'Bowling...	1	Did you know that people who support Donald Tr...	POLL: Most Trump Supporters Say Fake 'Bowling...

```
[44] # now we can delete extra columns
     cols_del =['title','text_processed']
     df = df.drop(cols_del,axis=1)

     df.head()
```

	label	final_news
0	0	U.S. Republican group hit by Russian-linked vi...
1	1	WIDOW OF BENGHAZI HERO In Powerful Ad To Call ...
2	1	BUSTED! OHIO ATTORNEY GENERAL DISCOVERS ILLEGA...
3	0	Plan to raise California minimum wage to $15 c...
4	1	POLL: Most Trump Supporters Say Fake 'Bowling...

The next step is to convert all the data to lowercase. This technique is often overlooked, but it can be highly effective when dealing with small amounts of data. Even though the words "Good," "good," and "GOOD" mean the same thing, a neural net model will assign different weights to them. This can lead to abrupt output and negatively impact the model's overall performance.

Additionally, we will eliminate all stopwords and non-alphabetic characters from the dataset. Below, you can find the code for carrying out these tasks.

```
[45] #creating list of possible stopwords from nltk library
     stop = stopwords.words('english')

     def cleanText(txt):
       # lowercaing
         txt = txt.lower()
       # removing stopwords
         txt = ' '.join([word for word in txt.split() if word not in (stop)])
       # removing non-alphabetic characters
         txt = re.sub('[^a-z]',' ',txt)
         return txt
```

```
[46] #applying text cleaning function to clean final_news
     df['final_news'] = df['final_news'].apply(cleanText)
     df.head()
```

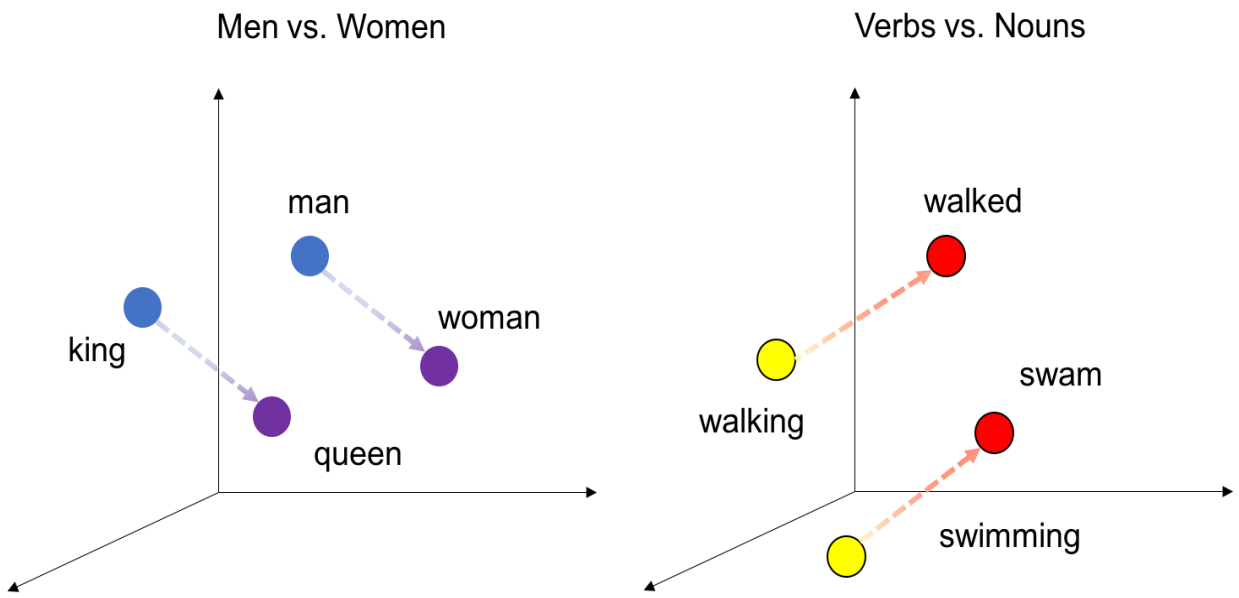
	label	final_news
0	0	u s republican group hit russian linked virus...
1	1	widow benghazi hero powerful ad call hillary c...
2	1	busted ohio attorney general discovers illega...
3	0	plan raise california minimum wage clears ...
4	1	poll trump supporters say fake bowling green...

6.1 Building Word Embeddings (GLOVE)

For this case study, we will be utilizing Global Vectors for Word Representation (GLOVE) word embeddings. This algorithm is used for unsupervised learning and helps to obtain vector representations for words.

The concept behind word embedding is to establish a contextual representation of words in a document, as well as to identify semantic and syntactic similarities among different words. This approach is proven to work well with Long Short-Term Memory (LSTM) in text classification tasks, as it provides a better understanding of contextual knowledge in comparison to vectorization methods alone.

Below is a visual representation of word embeddings:



The settings will be configured by the code snippet provided, and the location of the Glove embedding file will be specified. The Glove embedding file must be downloaded from the NLP Stanford website. No changes should be made to the code snippet.

```
[56] EMBEDDING_FILE ='/content/glove.6B.50d.txt'

[57] # configuration setting
     MAX_SEQUENCE_LENGTH = 100
     MAX_VOCAB_SIZE = 20000
     EMBEDDING_DIM = 50
     VALIDATION_SPLIT = 0.2
     BATCH_SIZE = 32
     EPOCHS = 10

[58] #Creating features and target variable
     X = df.drop(['label'],axis=1)
     y = df['label'].values
```

Upon reviewing our previous code, we noted that we set the maximum sequence length to 100 and the maximum vocabulary size to 20000. However, we can increase the maximum vocabulary size to 25000 or higher to improve accuracy, albeit at the expense of a longer training time. The embedding was set to 50 dimensions, meaning each word has 50 dimensions in vector space, but we can

experiment with 100 dimensions to refine accuracy. For the training phase, we will validate the model using 20% of the training data, which equates to a validation split of 0.2. While we utilized a batch size of 32, we can explore various batch sizes. The LSTM model was trained for only 10 epochs during the demo, but we can boost the number of epochs for superior results. Lastly, we will load the pre-trained word vectors from the embedding file.

6.2 Tokenize Text

In order for machine learning models to comprehend textual data, it is imperative that texts are converted into integers. This process is essential as these models do not have the ability to understand textual data.

```
[58] #Creating features and target variable
     X = df.drop(['label'],axis=1)
     y = df['label'].values
```

```
[59] # load in pre-trained word vectors
     print('Loading word vectors...')
     word2vec = {}
     with open(EMBEDDING_FILE) as f:
       # is just a space-separated text file in the format:
       # word vec[0] vec[1] vec[2] ...
       for line in f:
         values = line.split()
         word = values[0]
         vec = np.asarray(values[1:], dtype='float32')
         word2vec[word] = vec
     print('Found %s word vectors.' % len(word2vec))
```

```
     Loading word vectors...
     Found 400000 word vectors.
```

```
[60] # convert the sentences (strings) into integers
     tokenizer = Tokenizer(num_words=MAX_VOCAB_SIZE)
     tokenizer.fit_on_texts(list(X['final_news']))
     X = tokenizer.texts_to_sequences(list(X['final_news']))
```

```
[61] # pad sequences so that we get a N x T matrix
     X = pad_sequences(X, maxlen=MAX_SEQUENCE_LENGTH)
     print('Shape of data tensor:', X.shape)
```

```
     Shape of data tensor: (44898, 100)
```

Let's examine the code fragment that was previously displayed. First, we used a tokenizer to tokenize the final_news content, giving it a word count equal to the 20000-word limit we had earlier established. What are the purposes of tokenizer.fit_on_texts and tokenizer.text_to_sequences, respectively, is now the question.

As a result, a vocabulary index based on word frequency is created using the tokenizer.fit_on_texts, which is also used to construct a vocabulary index based on its frequency.

For instance, if you had the sentences "**My skill is different from other students**" and "**I am a good student**," word_index["skill"] would be equal to 0 and word_index["student"] to 1 (student appears twice, skill once), respectively. Tokenizer.text_to_sequences basically assigns each word in a sentence into a sequence of integers.

Thus, it substitutes each word in the phrase with its appropriate integer value from word_index.

6.3 Sequence Padding

The sequence has then been expanded to its maximum length of 100, as previously indicated. To form a neural network, all sequences must be the same length, therefore this is done to ensure that they are.

As a result, lengthier sequences are trimmed to a maximum length of 100, while shorter sequences that are shorter than the maximum length of 100 are padded with zeroes.

As of right now, we have designated this sequence-padded matrix as our feature vector X and labeled as our target variable Y, denoted by the notation df['label'].

The matrix shape is 44898 x 100 when the tensor shape is printed.

Next, we created a new variable called word2idx and saved the word to id (integers) mapping that was received from the tokenizer as shown below.

```
[62] # get word -> integer mapping
     word2idx = tokenizer.word_index
     print('Found %s unique tokens.' % len(word2idx))

     Found 115831 unique tokens.
```

6.3.1. Preparation of Embedding Matrix

After printing the length, we obtained 29101 distinct tokens in all. The following step is to make an embedding matrix. Below is a sharing of the code used to create the embedding matrix**.**

```
[63] # prepare embedding matrix
     print('Filling pre-trained embeddings...')
     num_words = min(MAX_VOCAB_SIZE, len(word2idx) + 1)
     embedding_matrix = np.zeros((num_words, EMBEDDING_DIM))
     for word, i in word2idx.items():
       if i < MAX_VOCAB_SIZE:
         embedding_vector = word2vec.get(word)
         if embedding_vector is not None:
           # words not found in embedding index will be all zeros.
           embedding_matrix[i] = embedding_vector

     Filling pre-trained embeddings...
```

According to the code, num_words must be at least as long as word2idx+1 and Max_VOCAB_SIZE.

The length of word2idx is 29101, while MAX_VOCAB_SIZE is 20000. The minimum of these two, being 20000 words, is hence the word count.

The next step is to generate an embedding matrix with a dimension of 50 and 20,000 words. The words that are absent from the matrix will be given the value zero.

6.4 Creation of Embedding Layer

Next, we will create an embedding layer that will be used as input in the LSTM model.

```
[64] # load pre-trained word embeddings into an Embedding layer
     # note that we set trainable = False so as to keep the embeddings fixed
     embedding_layer = Embedding(
       num_words,
       EMBEDDING_DIM,
       weights=[embedding_matrix],
       input_length=MAX_SEQUENCE_LENGTH,
       trainable=False
     )
```

6.5 Model Building

The deep learning model Long Term Short Memory (LSTM) will be constructed in this step. We'll be employing LSTM in both directions for this case study.

```
[66] print('Building model...')

     # create an LSTM network with a single LSTM
     input_ = Input(shape=(MAX_SEQUENCE_LENGTH,))
     x = embedding_layer(input_)
     # x = LSTM(15, return_sequences=True)(x)
     x = Bidirectional(LSTM(15, return_sequences=True))(x)
     x = GlobalMaxPool1D()(x)
     output = Dense(1, activation="sigmoid")(x)

     model = Model(input_, output)
     model.compile(
       loss='binary_crossentropy',
       optimizer='adam',
       metrics=['accuracy']
     )
     model.summary()
```

Based on the code displayed above, we utilized a single hidden layer of the Bidirectional LSTM layer consisting of 15 neurons. To access the hidden state output for each input time step, we set the value of return_sequences to "True". This is a hyper-parameter that can be modified to achieve the desired results.

In addition, we can also explore alternative versions of LSTMs such as unidirectional LSTM and GRU. The number of neurons can also be increased to determine if there are any enhancements in performance. The comprehensive model summary is presented below.

```
Building model...
Model: "model_1"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 input_2 (InputLayer)        [(None, 100)]             0

 embedding (Embedding)       (None, 100, 50)           1000000

 bidirectional_1 (Bidirecti  (None, 100, 30)           7920
 onal)

 global_max_pooling1d_1 (Gl  (None, 30)                0
 obalMaxPooling1D)

 dense_1 (Dense)             (None, 1)                 31

=================================================================
Total params: 1007951 (3.85 MB)
Trainable params: 7951 (31.06 KB)
Non-trainable params: 1000000 (3.81 MB)
_________________________________________________________________
```

6.6 Model Parameters

The essential thing to note in this situation is that the total number of model parameters is 1,007,951, whereas the training parameters are just 7951, which are made up of the parameters for the bidirectional LSTM, which are 7920, and the parameter for dense layers, which is 31. The cause of this is that we already set

trainable = false for the embedding layer, rendering 1000000 of its parameters non trainable.

6.7 Train test split

We now divide the feature and the target variable into train and test sets in the ratio of 80:20, where 80% of the data is used to train the model and 20% is utilised for testing.

```
[67] # train Test split in the ratio of 80:20
     X_train, X_test, y_train, y_test = train_test_split(X, y,test_size=0.20,stratify=y,random_state=0)
```

6.8 Fitting model

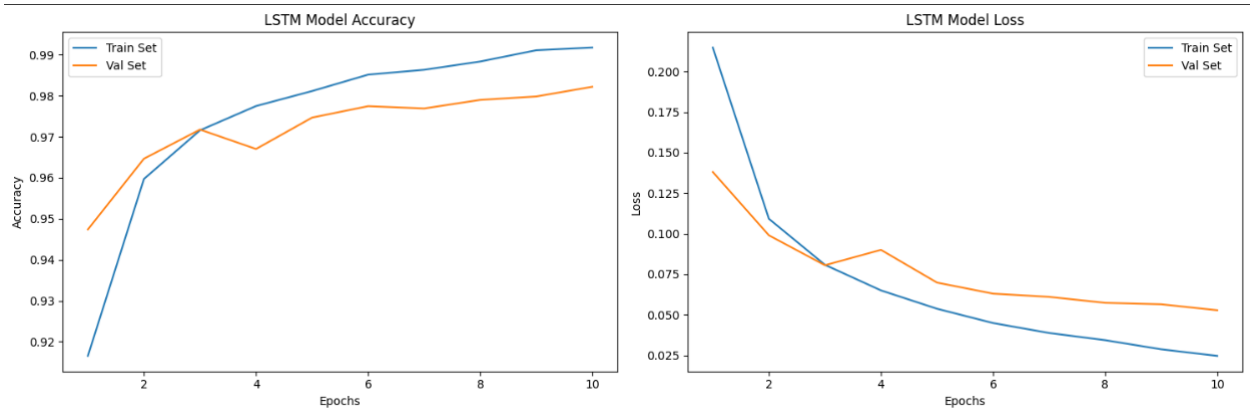
With a batch size equal to 32 and a validation split of 20%, we fit the model on the training data in this stage.

```
[68] print('Training model...')
     r = model.fit(
       X_train,
       y_train,
       batch_size=BATCH_SIZE,
       epochs=EPOCHS,
       validation_split=VALIDATION_SPLIT
     )

     Training model...
     Epoch 1/10
     898/898 [==============================] - 67s 70ms/step - loss: 0.2147 - accuracy: 0.9165 - val_loss: 0.1380 - val_accuracy: 0.9474
     Epoch 2/10
     898/898 [==============================] - 59s 66ms/step - loss: 0.1092 - accuracy: 0.9597 - val_loss: 0.0990 - val_accuracy: 0.9646
     Epoch 3/10
     898/898 [==============================] - 59s 66ms/step - loss: 0.0808 - accuracy: 0.9716 - val_loss: 0.0805 - val_accuracy: 0.9717
     Epoch 4/10
     898/898 [==============================] - 59s 66ms/step - loss: 0.0650 - accuracy: 0.9775 - val_loss: 0.0900 - val_accuracy: 0.9670
     Epoch 5/10
     898/898 [==============================] - 59s 66ms/step - loss: 0.0537 - accuracy: 0.9811 - val_loss: 0.0698 - val_accuracy: 0.9747
     Epoch 6/10
     898/898 [==============================] - 63s 70ms/step - loss: 0.0449 - accuracy: 0.9851 - val_loss: 0.0630 - val_accuracy: 0.9774
     Epoch 7/10
     898/898 [==============================] - 61s 68ms/step - loss: 0.0388 - accuracy: 0.9863 - val_loss: 0.0610 - val_accuracy: 0.9769
     Epoch 8/10
     898/898 [==============================] - 60s 67ms/step - loss: 0.0343 - accuracy: 0.9883 - val_loss: 0.0573 - val_accuracy: 0.9790
     Epoch 9/10
     898/898 [==============================] - 59s 65ms/step - loss: 0.0287 - accuracy: 0.9911 - val_loss: 0.0564 - val_accuracy: 0.9798
     Epoch 10/10
     898/898 [==============================] - 58s 65ms/step - loss: 0.0246 - accuracy: 0.9918 - val_loss: 0.0528 - val_accuracy: 0.9822
```

As we can see from the results above, the model was validated with an accuracy of 98.58% after only 10 epochs.

To demonstrate how the model training is progressing, the following loss and accuracy graphs for training and validation are displayed.

```
[69] acc = r.history['accuracy']
     val_acc = r.history['val_accuracy']
     loss = r.history['loss']
     val_loss = r.history['val_loss']
     epochs_range = range(1, len(r.epoch) + 1)

     plt.figure(figsize=(15,5))

     plt.subplot(1, 2, 1)
     plt.plot(epochs_range, acc, label='Train Set')
     plt.plot(epochs_range, val_acc, label='Val Set')
     plt.legend(loc="best")
     plt.xlabel('Epochs')
     plt.ylabel('Accuracy')
     plt.title('LSTM Model Accuracy')

     plt.subplot(1, 2, 2)
     plt.plot(epochs_range, loss, label='Train Set')
     plt.plot(epochs_range, val_loss, label='Val Set')
     plt.legend(loc="best")
     plt.xlabel('Epochs')
     plt.ylabel('Loss')
     plt.title('LSTM Model Loss')

     plt.tight_layout()
     plt.show()
```



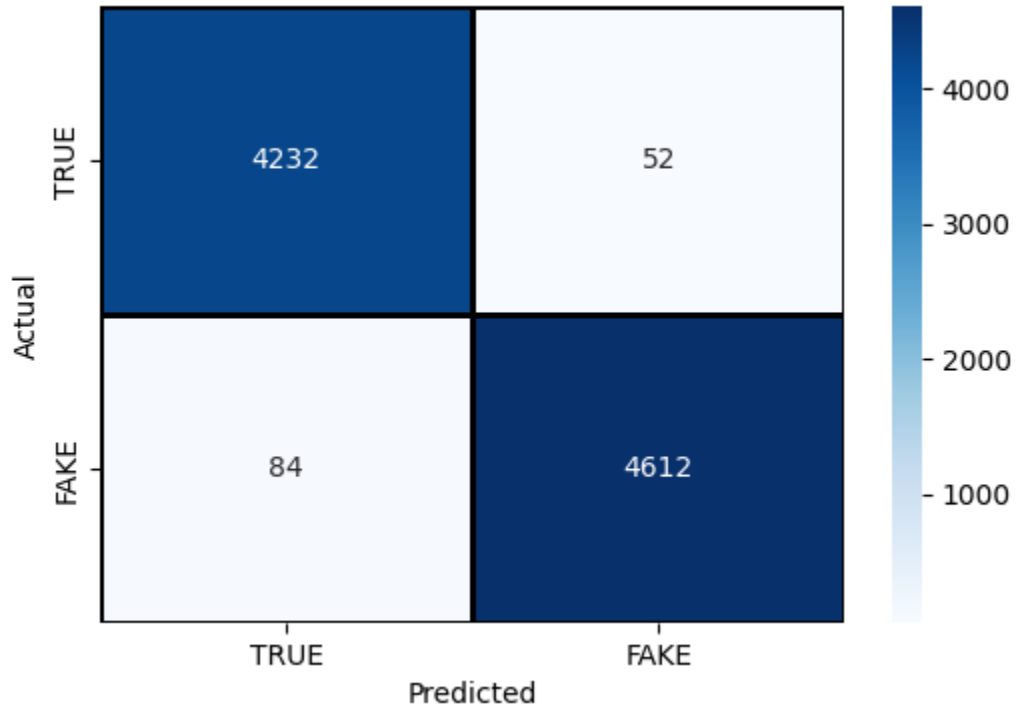
As we can see from the above plot, the model's validation accuracy reached to 98% at around the 10th epoch.

6.9 Model result

The training and test accuracy of the model is shown below.

```
1123/1123 [==============================] - 28s 25ms/step - loss: 0.0257 - accuracy: 0.9921
Accuracy of the model on Training Data is -  99.20930862426758
281/281 [==============================] - 4s 14ms/step - loss: 0.0450 - accuracy: 0.9849
Accuracy of the model on Testing Data is -  98.48552346229553
```

The confusion matrix of the model is shown below.



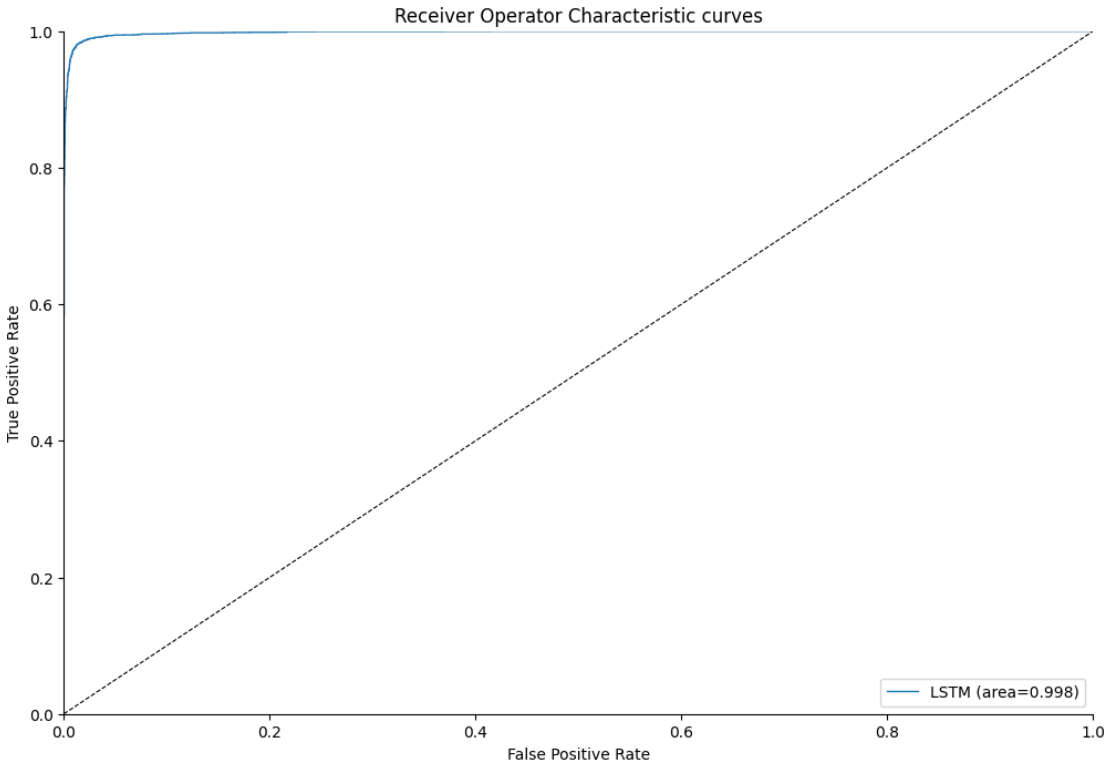
The model has performed impressively, as can be seen from the confusion matrix above. Let's now look at the classification report to better comprehend the overall performance from a statistical perspective.

	precision	recall	f1-score	support
0	0.98	0.99	0.98	4284
1	0.99	0.98	0.99	4696
accuracy			0.98	8980
macro avg	0.98	0.98	0.98	8980
weighted avg	0.98	0.98	0.98	8980

As a result, while precision is better for real news than fake news, our model has a higher recall and F1 score of 99% and 98% for both, respectively. The F1-score, which is the harmonic mean of recall and precision, provides a superior way to assess the performance of any machine learning or deep learning model.

The model's performance can be enhanced by adjusting the hyper-parameters, but an F1 score of 98% or above is still a respectable result.

In addition to that, the Receiver Operating Characteristics curve, commonly known as the ROC curve, is another performance indicator that is frequently employed in binary classification jobs. Below is a figure showing the ROC curve for the model.



The model works reasonably well, as shown by the ROC figure above, which shows that the model has a greater Area under the Curve (AUC) of 0.998. The optimal model has an AUC of 1.

6.10 Model Prediction

Using the algorithm to distinguish between fake and real news in examples of news items is the most crucial aspect of this project. But to do that, sample texts must first go through some preprocessing before being fed into the LSTM model.

```
[80] testSent =["In a big win for India, the New Delhi Declaration was adopted during the G20 Summit. Prime Minister Narendra Modi announced the adoption and
          "Trey Gowdy destroys this clueless DHS employee when asking about the due process of getting on the terror watch list. Her response is priceless:
        ]
```

```
[81] def cleanText(txt):
        txt = txt.lower()
        txt = ' '.join([word for word in txt.split() if word not in (stop)])
        txt = re.sub('[^a-z]',' ',txt)
        return txt
```

```
[84] def predict_text(lst_text):
        test = tokenizer.texts_to_sequences(lst_text)
        # pad sequences so that we get a N x T matrix
        testX = pad_sequences(test, maxlen=MAX_SEQUENCE_LENGTH)
        df_test = pd.DataFrame(lst_text, columns = ['test_sent'])

        prediction = model.predict(testX)
        df_test['prediction']=prediction
        df_test["test_sent"] = df_test["test_sent"].apply(cleanText)
        df_test['prediction']=df_test['prediction'].apply(lambda x: "Fake" if x>=0.5 else "Real")
        return df_test
```

```
[83] #getting the prediction by passing list of sample news articles
     df_testsent = predict_text(testSent)
     df_testsent
```

```
1/1 [==============================] - 0s 56ms/step
```

	test_sent	prediction
0	big win india new delhi declaration adopted g...	Real
1	trey gowdy destroys clueless dhs employee aski...	Fake
2	poland new prime minister faces difficult bala...	Real

6.11 Source Code

Here's the source Code of my Git repository:

https://github.com/Santosh20248/Diginique-TechLab-Project.git

```
import numpy as np
import pandas as pd
```

```
from collections import defaultdict
import re

import matplotlib.pyplot as plt
import seaborn as sns
import nltk
from wordcloud import WordCloud

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

```
Mounted at /content/drive
```

```
df_fake = pd.read_csv('/content/drive/MyDrive/news/Fake.csv')
```

```
df_fake.head()
```

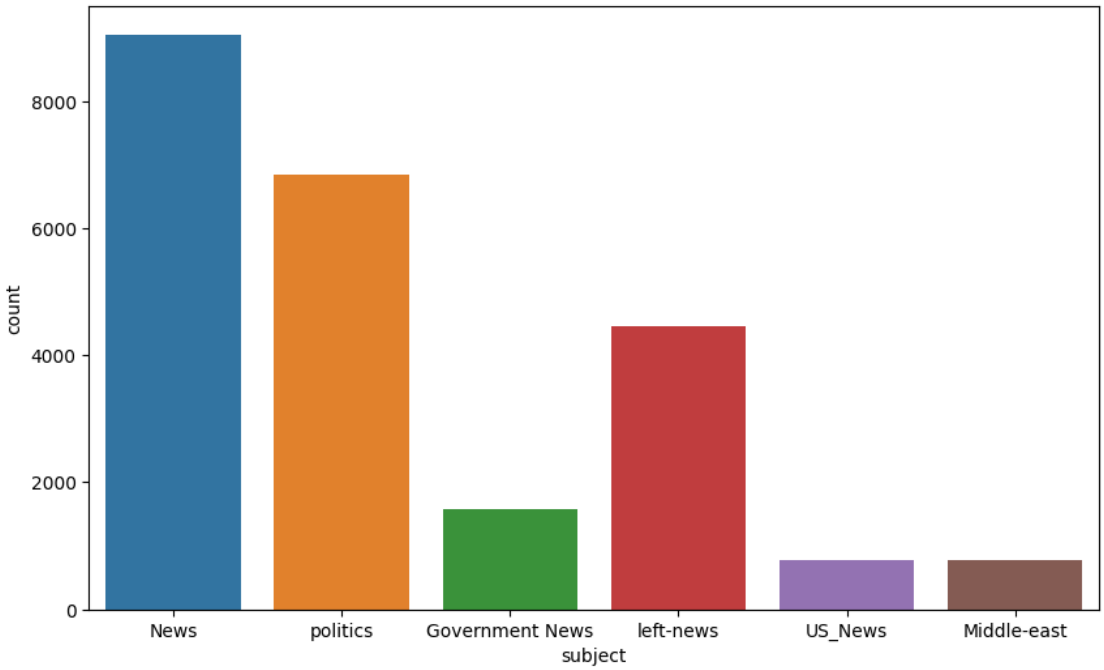
```
                                               title  \
0   Donald Trump Sends Out Embarrassing New Year’...
1   Drunk Bragging Trump Staffer Started Russian ...
2   Sheriff David Clarke Becomes An Internet Joke...
3   Trump Is So Obsessed He Even Has Obama’s Name...
4   Pope Francis Just Called Out Donald Trump Dur...

                                                text subject  \
0  Donald Trump just couldn t wish all Americans ...    News
1  House Intelligence Committee Chairman Devin Nu...    News
2  On Friday, it was revealed that former Milwauk...    News
3  On Christmas day, Donald Trump announced that ...    News
4  Pope Francis used his annual Christmas Day mes...    News

                date
0  December 31, 2017
1  December 31, 2017
2  December 30, 2017
3  December 29, 2017
4  December 25, 2017
```

```
df_fake['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 = df_fake)
```

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




```
# Dropping unneccesary features
df_fake=df_fake.drop(['subject','date'],axis=1)

df_fake.head()
```

```
                                                title  \
0   Donald Trump Sends Out Embarrassing New Year’...
1   Drunk Bragging Trump Staffer Started Russian ...
2   Sheriff David Clarke Becomes An Internet Joke...
3   Trump Is So Obsessed He Even Has Obama’s Name...
4   Pope Francis Just Called Out Donald Trump Dur...

                                                text
0  Donald Trump just couldn t wish all Americans ...
1  House Intelligence Committee Chairman Devin Nu...
2  On Friday, it was revealed that former Milwauk...
3  On Christmas day, Donald Trump announced that ...
4  Pope Francis used his annual Christmas Day mes...
```

```
# Assigning label 'FAKE' by creating target column i.e., label
df_fake['label'] ='FAKE'
df_fake.head()
```

```
                                                title  \
0   Donald Trump Sends Out Embarrassing New Year’...
1   Drunk Bragging Trump Staffer Started Russian ...
2   Sheriff David Clarke Becomes An Internet Joke...
3   Trump Is So Obsessed He Even Has Obama’s Name...
4   Pope Francis Just Called Out Donald Trump Dur...
```

```
                                                text label
0  Donald Trump just couldn t wish all Americans ...  FAKE
1  House Intelligence Committee Chairman Devin Nu...  FAKE
2  On Friday, it was revealed that former Milwauk...  FAKE
3  On Christmas day, Donald Trump announced that ...  FAKE
4  Pope Francis used his annual Christmas Day mes...  FAKE
```

```
df_fake.text.shape
```

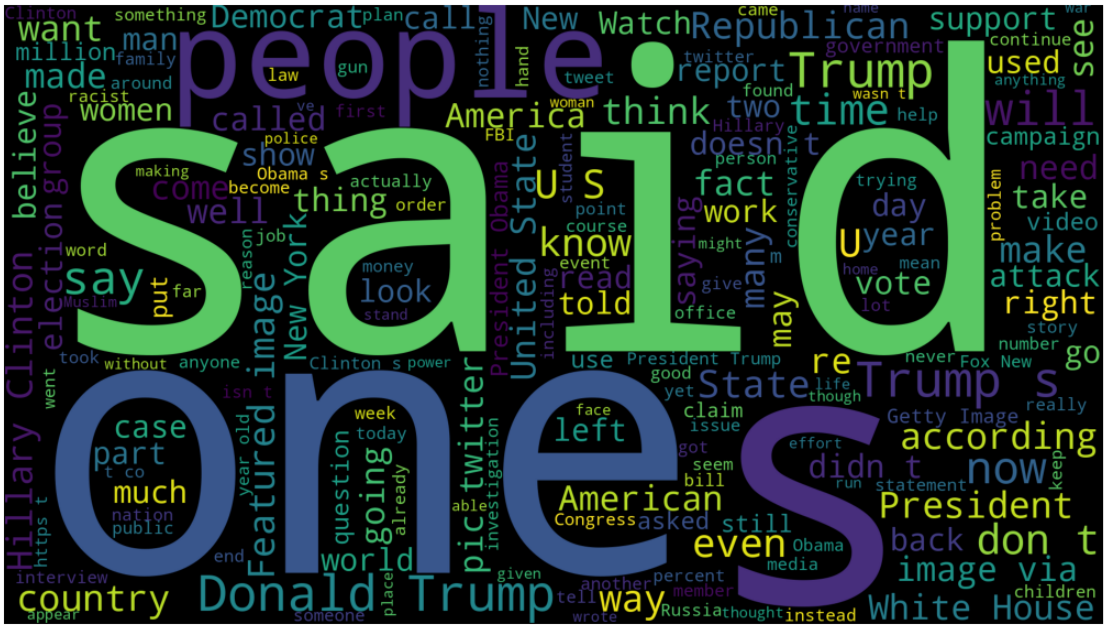
```
(23481,)
```

Wordcloud

```
text = ' '.join(df_fake['text'].tolist())
' '.join(['This','is', 'the', 'data'])
```

```
{"type":"string"}
```

```
wordcloud = WordCloud(width=1920, height=1080).generate(text)
fig = plt.figure(figsize=(10, 10))
plt.imshow(wordcloud)
plt.axis('off')
plt.tight_layout(pad=0)
plt.show()
```




```
df_true = pd.read_csv('/content/drive/MyDrive/news/True.csv')
df_true.head()
```

```
                                               title  \
0  As U.S. budget fight looms, Republicans flip t...
1  U.S. military to accept transgender recruits o...
2  Senior U.S. Republican senator: 'Let Mr. Muell...
```

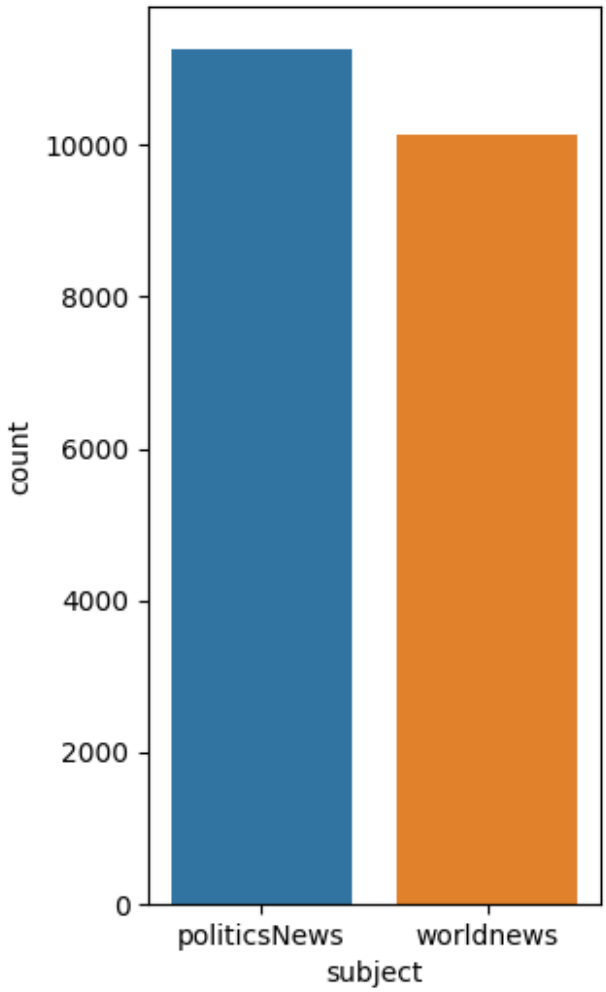
```
3  FBI Russia probe helped by Australian diplomat...
4  Trump wants Postal Service to charge 'much mor...

                                                text       subject  \
0  WASHINGTON (Reuters) - The head of a conservat...  politicsNews
1  WASHINGTON (Reuters) - Transgender people will...  politicsNews
2  WASHINGTON (Reuters) - The special counsel inv...  politicsNews
3  WASHINGTON (Reuters) - Trump campaign adviser ...  politicsNews
4  SEATTLE/WASHINGTON (Reuters) - President Donal...  politicsNews

                 date
0  December 31, 2017
1  December 29, 2017
2  December 31, 2017
3  December 30, 2017
4  December 29, 2017
```

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

```
politicsNews    11272
worldnews       10145
Name: subject, dtype: int64
```

```
plt.figure(figsize=(3, 6))
sns.countplot(x = 'subject', data = df_true)
```

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




```
df_true=df_true.drop(['subject','date'],axis=1)
df_true['label']='TRUE'
df_true.head()
```

```
                                                title  \
0  As U.S. budget fight looms, Republicans flip t...
1  U.S. military to accept transgender recruits o...
2  Senior U.S. Republican senator: 'Let Mr. Muell...
3  FBI Russia probe helped by Australian diplomat...
4  Trump wants Postal Service to charge 'much mor...

                                                text label
0  WASHINGTON (Reuters) - The head of a conservat...  TRUE
1  WASHINGTON (Reuters) - Transgender people will...  TRUE
2  WASHINGTON (Reuters) - The special counsel inv...  TRUE
3  WASHINGTON (Reuters) - Trump campaign adviser ...  TRUE
4  SEATTLE/WASHINGTON (Reuters) - President Donal...  TRUE
```

```
data_train = pd.concat([df_true, df_fake], ignore_index=True)
data_train.head()
```

```
                                                title  \
0  As U.S. budget fight looms, Republicans flip t...
1  U.S. military to accept transgender recruits o...
2  Senior U.S. Republican senator: 'Let Mr. Muell...
3  FBI Russia probe helped by Australian diplomat...
4  Trump wants Postal Service to charge 'much mor...

                                                text label
0  WASHINGTON (Reuters) - The head of a conservat...  TRUE
1  WASHINGTON (Reuters) - Transgender people will...  TRUE
2  WASHINGTON (Reuters) - The special counsel inv...  TRUE
3  WASHINGTON (Reuters) - Trump campaign adviser ...  TRUE
4  SEATTLE/WASHINGTON (Reuters) - President Donal...  TRUE
```

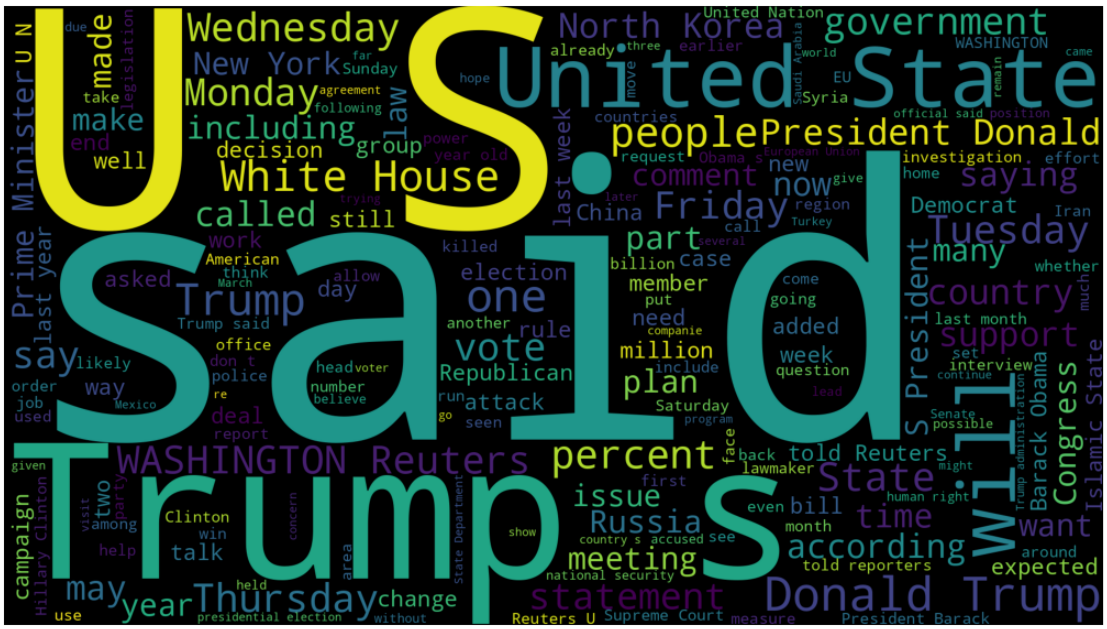
```
data_train.shape
```

```
(44898, 3)
```

```
text = ' '.join(df_true['text'].tolist())
' '.join(['This','is', 'the', 'data'])
```

```
{"type":"string"}
```

```
wordcloud = WordCloud(width=1920, height=1080).generate(text)
fig = plt.figure(figsize=(10, 10))
plt.imshow(wordcloud)
plt.axis('off')
plt.tight_layout(pad=0)
plt.show()
```



Preprocessing Text

```
df_fake['label'] = 'FAKE'
df_fake.head()
```

```
                                               title  \
0   Donald Trump Sends Out Embarrassing New Year’...
1   Drunk Bragging Trump Staffer Started Russian ...
2   Sheriff David Clarke Becomes An Internet Joke...
3   Trump Is So Obsessed He Even Has Obama’s Name...
4   Pope Francis Just Called Out Donald Trump Dur...

                                                text label
0  Donald Trump just couldn t wish all Americans ...  FAKE
1  House Intelligence Committee Chairman Devin Nu...  FAKE
2  On Friday, it was revealed that former Milwauk...  FAKE
3  On Christmas day, Donald Trump announced that ...  FAKE
4  Pope Francis used his annual Christmas Day mes...  FAKE
```

```
df_true['label']='TRUE'
```

```
df_true.head()
```

```
                                               title  \
0  As U.S. budget fight looms, Republicans flip t...
1  U.S. military to accept transgender recruits o...
2  Senior U.S. Republican senator: 'Let Mr. Muell...
3  FBI Russia probe helped by Australian diplomat...
4  Trump wants Postal Service to charge 'much mor...

                                                text label
0  WASHINGTON (Reuters) - The head of a conservat...  TRUE
1  WASHINGTON (Reuters) - Transgender people will...  TRUE
2  WASHINGTON (Reuters) - The special counsel inv...  TRUE
3  WASHINGTON (Reuters) - Trump campaign adviser ...  TRUE
4  SEATTLE/WASHINGTON (Reuters) - President Donal...  TRUE
```

```
data_train = pd.concat([df_true, df_fake], ignore_index=True)
data_train.head()
```

```
                                               title  \
0  As U.S. budget fight looms, Republicans flip t...
1  U.S. military to accept transgender recruits o...
2  Senior U.S. Republican senator: 'Let Mr. Muell...
3  FBI Russia probe helped by Australian diplomat...
4  Trump wants Postal Service to charge 'much mor...

                                                text label
0  WASHINGTON (Reuters) - The head of a conservat...  TRUE
1  WASHINGTON (Reuters) - Transgender people will...  TRUE
2  WASHINGTON (Reuters) - The special counsel inv...  TRUE
3  WASHINGTON (Reuters) - Trump campaign adviser ...  TRUE
4  SEATTLE/WASHINGTON (Reuters) - President Donal...  TRUE
```

```
#Shuffling the dataset
data_train=data_train.reindex(np.random.permutation(data_train.index))
data_train.head()
```

```
                                                   title  \
12833  U.N. to assess if either side trying to 'sabot...
7827   U.S. conservative group backs Republicans who ...
29808   Martin O’Malley Suspends Campaign After Predi...
42552  HUNGARIANS FIND SHOCKING VIDEOS On Phones Left...
35441  DEMOCRATS FREAK OUT As Shocking Number Of Unio...

                                                    text label
12833  GENEVA (Reuters) - The mediator of U.N.-led Sy...  TRUE
7827   WASHINGTON (Reuters) - A conservative nonprofi...  TRUE
29808  Democratic presidential candidate Martin O Mal...  FAKE
42552  If you were to believe Barack Hussein Obama, H...  FAKE
35441  As millions of dollars of union dues flow into...  FAKE
```

```
data_train.shape
```

```
(44898, 3)
```

```
data_train.to_csv('Fake_news.csv')
```

```
import pandas as pd
import numpy as np
import itertools
import seaborn as sns
import nltk, re, string
from string import punctuation
from nltk.corpus import stopwords
from sklearn import metrics
import matplotlib.pyplot as plt
from wordcloud import WordCloud,STOPWORDS
from sklearn.metrics import
classification_report,confusion_matrix,accuracy_score,roc_auc_score
from sklearn.model_selection import train_test_split
from keras.models import Model
from keras.layers import Dense, Embedding, Input,LSTM, Bidirectional,
GlobalMaxPool1D, Dropout
from keras.preprocessing.text import Tokenizer
nltk.download('stopwords')
```

```
[nltk_data] Downloading package stopwords to /root/nltk_data...
[nltk_data]   Unzipping corpora/stopwords.zip.
```

```
True
```

```
import tensorflow as tf
tf.__version__
! sudo pip3 install keras
from tensorflow.python.keras.models import Sequential
```

```
from keras.preprocessing.sequence import pad_sequences
from keras.layers import Dense, Embedding, LSTM, Conv1D, MaxPool1D
from keras.models import Model
from keras.preprocessing.text import Tokenizer
```

```
Requirement already satisfied: keras in /usr/local/lib/python3.10/dist-
packages (2.13.1)
```

```
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).
```

```
df = pd.read_csv('/content/Fake_news.csv')
df.head()
```

```
   Unnamed: 0                                              title  \
0       12833  U.N. to assess if either side trying to 'sabot...
1        7827  U.S. conservative group backs Republicans who ...
2       29808   Martin O’Malley Suspends Campaign After Predi...
3       42552  HUNGARIANS FIND SHOCKING VIDEOS On Phones Left...
4       35441  DEMOCRATS FREAK OUT As Shocking Number Of Unio...

                                                text label
0  GENEVA (Reuters) - The mediator of U.N.-led Sy...  TRUE
1  WASHINGTON (Reuters) - A conservative nonprofi...  TRUE
2  Democratic presidential candidate Martin O Mal...  FAKE
3  If you were to believe Barack Hussein Obama, H...  FAKE
4  As millions of dollars of union dues flow into...  FAKE
```

```
import re
# Function for extracting desired text using regex
def extract_txt(text):
    regex = re.search(r"(?<=\(Reuters\)\s\-\s).*",text)
    if regex:
        return regex.group(0)
    return text
    #Applying regex function to retain only relevant text
df['text_processed'] = df['text'].apply(extract_txt)
df.head()
```

```
   Unnamed: 0                                              title  \
0       12833  U.N. to assess if either side trying to 'sabot...
1        7827  U.S. conservative group backs Republicans who ...
2       29808   Martin O’Malley Suspends Campaign After Predi...
3       42552  HUNGARIANS FIND SHOCKING VIDEOS On Phones Left...
4       35441  DEMOCRATS FREAK OUT As Shocking Number Of Unio...

                                                text label  \
0  GENEVA (Reuters) - The mediator of U.N.-led Sy...  TRUE
```

```
1  WASHINGTON (Reuters) - A conservative nonprofi...  TRUE
2  Democratic presidential candidate Martin O Mal...  FAKE
3  If you were to believe Barack Hussein Obama, H...  FAKE
4  As millions of dollars of union dues flow into...  FAKE

                                      text_processed
0  The mediator of U.N.-led Syrian peace talks in...
1  A conservative nonprofit group that lobbies Re...
2  Democratic presidential candidate Martin O Mal...
3  If you were to believe Barack Hussein Obama, H...
4  As millions of dollars of union dues flow into...
```

```
#Checking dataframe containing only True News
df[df.label=="TRUE"]
```

```
       Unnamed: 0                                              title  \
0           12833  U.N. to assess if either side trying to 'sabot...
1            7827  U.S. conservative group backs Republicans who ...
5            7440  Danish PM says world needs U.S. 'not to close ...
7            2922  Factbox: Trump on Twitter - Media, American tr...
11           4554  N.J. Democrats divided on renewing 'Bridgegate...
...           ...                                                ...
44885        4080  U.S. commerce secretary eyes more trade moves:...
44886        6715  French Foreign Minister says Trump's handling ...
44887       16027  Nigeria President Buhari plans to expand his c...
44889        9944  Unions endorse Sanders, Clinton for president ...
44894        5010  Maryland to join other states in court challen...

                                                    text label  \
0      GENEVA (Reuters) - The mediator of U.N.-led Sy...  TRUE
1      WASHINGTON (Reuters) - A conservative nonprofi...  TRUE
5      COPENHAGEN (Reuters) - Danish Prime Minister L...  TRUE
7      The following statements were posted to the ve...  TRUE
11      (This March 30 story was corrected to note Pr...  TRUE
...                                                  ...   ...
44885  WASHINGTON (Reuters) - The Trump administratio...  TRUE
44886  PARIS (Reuters) - France’s foreign minister on...  TRUE
44887  ABUJA (Reuters) - Nigeria s cabinet will be ex...  TRUE
44889  WASHINGTON (Reuters) - Democratic presidential...  TRUE
44894  WASHINGTON (Reuters) - Maryland became the lat...  TRUE

                                           text_processed
0      The mediator of U.N.-led Syrian peace talks in...
1      A conservative nonprofit group that lobbies Re...
5      Danish Prime Minister Lars Lokke Rasmussen joi...
7      The following statements were posted to the ve...
11     A day after two former allies of New Jersey Go...
...                                                  ...
44885  The Trump administration may undertake trade a...
44886  France’s foreign minister on Wednesday accused...
```

```
44887  Nigeria s cabinet will be expanded to bring in...
44889  Democratic presidential candidate Bernie Sande...
44894  Maryland became the latest state to join in le...

[21417 rows x 5 columns]
```

```
# Drop extra column
df = df.drop(['text','Unnamed: 0'],axis=1)
df["label"] = df.label.apply(lambda  x:0 if x=='TRUE' else 1)
df.head()
```

```
                                               title  label  \
0  U.N. to assess if either side trying to 'sabot...      0
1  U.S. conservative group backs Republicans who ...      0
2   Martin O’Malley Suspends Campaign After Predi...      1
3  HUNGARIANS FIND SHOCKING VIDEOS On Phones Left...      1
4  DEMOCRATS FREAK OUT As Shocking Number Of Unio...      1

                                      text_processed
0  The mediator of U.N.-led Syrian peace talks in...
1  A conservative nonprofit group that lobbies Re...
2  Democratic presidential candidate Martin O Mal...
3  If you were to believe Barack Hussein Obama, H...
4  As millions of dollars of union dues flow into...
```

```
#Combining text_processed and title for creating full news article with
headline
df['final_news'] = df['title'] + " " + df['text_processed']
df.head()
```

```
                                               title  label  \
0  U.N. to assess if either side trying to 'sabot...      0
1  U.S. conservative group backs Republicans who ...      0
2   Martin O’Malley Suspends Campaign After Predi...      1
3  HUNGARIANS FIND SHOCKING VIDEOS On Phones Left...      1
4  DEMOCRATS FREAK OUT As Shocking Number Of Unio...      1

                                      text_processed  \
0  The mediator of U.N.-led Syrian peace talks in...
1  A conservative nonprofit group that lobbies Re...
2  Democratic presidential candidate Martin O Mal...
3  If you were to believe Barack Hussein Obama, H...
4  As millions of dollars of union dues flow into...

                                          final_news
0  U.N. to assess if either side trying to 'sabot...
1  U.S. conservative group backs Republicans who ...
2   Martin O’Malley Suspends Campaign After Predi...
3  HUNGARIANS FIND SHOCKING VIDEOS On Phones Left...
4  DEMOCRATS FREAK OUT As Shocking Number Of Unio...
```

```
# now we can delete extra columns
cols_del =['title','text_processed']
df = df.drop(cols_del,axis=1)

df.head()
```

```
   label                                          final_news
0      0  U.N. to assess if either side trying to 'sabot...
1      0  U.S. conservative group backs Republicans who ...
2      1   Martin O’Malley Suspends Campaign After Predi...
3      1  HUNGARIANS FIND SHOCKING VIDEOS On Phones Left...
4      1  DEMOCRATS FREAK OUT As Shocking Number Of Unio...
```

```
#creating list of possible stopwords from nltk library
stop = stopwords.words('english')

def cleanText(txt):
  # lowercaing
    txt = txt.lower()
  # removing stopwords
    txt = ' '.join([word for word in txt.split() if word not in (stop)])
  # removing non-alphabetic characters
    txt = re.sub('[^a-z]',' ',txt)
    return txt

#applying text cleaning function to clean final_news
df['final_news'] = df['final_news'].apply(cleanText)
df.head()
```

```
   label                                          final_news
0      0  u n  assess either side trying  sabotage  syri...
1      0  u s  conservative group backs republicans favo...
2      1  martin o malley suspends campaign predictably ...
3      1  hungarians find shocking videos phones left be...
4      1  democrats freak shocking number union members ...
```

```
y = df['label'].values
X = df.drop(['label'],axis=1)

y.shape
```

```
(44898,)
```

```
import gensim

!wget http://nlp.stanford.edu/data/glove.6B.zip
```

```
--2023-09-11 12:59:11--  http://nlp.stanford.edu/data/glove.6B.zip
Resolving nlp.stanford.edu (nlp.stanford.edu)... 171.64.67.140
Connecting to nlp.stanford.edu (nlp.stanford.edu)|171.64.67.140|:80...
connected.
HTTP request sent, awaiting response... 302 Found
```

```
Location: https://nlp.stanford.edu/data/glove.6B.zip [following]
--2023-09-11 12:59:11--  https://nlp.stanford.edu/data/glove.6B.zip
Connecting to nlp.stanford.edu (nlp.stanford.edu)|171.64.67.140|:443...
connected.
HTTP request sent, awaiting response... 301 Moved Permanently
Location: https://downloads.cs.stanford.edu/nlp/data/glove.6B.zip [following]
--2023-09-11 12:59:12--
https://downloads.cs.stanford.edu/nlp/data/glove.6B.zip
Resolving downloads.cs.stanford.edu (downloads.cs.stanford.edu)...
171.64.64.22
Connecting to downloads.cs.stanford.edu
(downloads.cs.stanford.edu)|171.64.64.22|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 862182613 (822M) [application/zip]
Saving to: ‘glove.6B.zip’

glove.6B.zip        100%[===================>] 822.24M  5.07MB/s    in 2m 41s

2023-09-11 13:01:53 (5.12 MB/s) - ‘glove.6B.zip’ saved [862182613/862182613]
```

```
!unzip glove*.zip
```

```
Archive:  glove.6B.zip
  inflating: glove.6B.50d.txt
  inflating: glove.6B.100d.txt
  inflating: glove.6B.200d.txt
  inflating: glove.6B.300d.txt
```

```
from gensim.scripts.glove2word2vec import glove2word2vec

EMBEDDING_FILE ='/content/glove.6B.50d.txt'

# configuration setting
MAX_SEQUENCE_LENGTH = 100
MAX_VOCAB_SIZE = 20000
EMBEDDING_DIM = 50
VALIDATION_SPLIT = 0.2
BATCH_SIZE = 32
EPOCHS = 10

#Creating features and target variable
X = df.drop(['label'],axis=1)
y = df['label'].values

# load in pre-trained word vectors
print('Loading word vectors...')
word2vec = {}
with open(EMBEDDING_FILE) as f:
  # is just a space-separated text file in the format:
  # word vec[0] vec[1] vec[2] ...
```

```
  for line in f:
    values = line.split()
    word = values[0]
    vec = np.asarray(values[1:], dtype='float32')
    word2vec[word] = vec
print('Found %s word vectors.' % len(word2vec))
```

```
Loading word vectors...
Found 400000 word vectors.
```

```
# convert the sentences (strings) into integers
tokenizer = Tokenizer(num_words=MAX_VOCAB_SIZE)
tokenizer.fit_on_texts(list(X['final_news']))
X = tokenizer.texts_to_sequences(list(X['final_news']))

# pad sequences so that we get a N x T matrix
X = pad_sequences(X, maxlen=MAX_SEQUENCE_LENGTH)
print('Shape of data tensor:', X.shape)
```

```
Shape of data tensor: (44898, 100)
```

```
# get word -> integer mapping
word2idx = tokenizer.word_index
print('Found %s unique tokens.' % len(word2idx))
```

```
Found 115831 unique tokens.
```

```
# prepare embedding matrix
print('Filling pre-trained embeddings...')
num_words = min(MAX_VOCAB_SIZE, len(word2idx) + 1)
embedding_matrix = np.zeros((num_words, EMBEDDING_DIM))
for word, i in word2idx.items():
  if i < MAX_VOCAB_SIZE:
    embedding_vector = word2vec.get(word)
    if embedding_vector is not None:
      # words not found in embedding index will be all zeros.
      embedding_matrix[i] = embedding_vector
```

```
Filling pre-trained embeddings...
```

```
# load pre-trained word embeddings into an Embedding layer
# note that we set trainable = False so as to keep the embeddings fixed
embedding_layer = Embedding(
  num_words,
  EMBEDDING_DIM,
  weights=[embedding_matrix],
  input_length=MAX_SEQUENCE_LENGTH,
  trainable=False
)

print('Building model...')
```

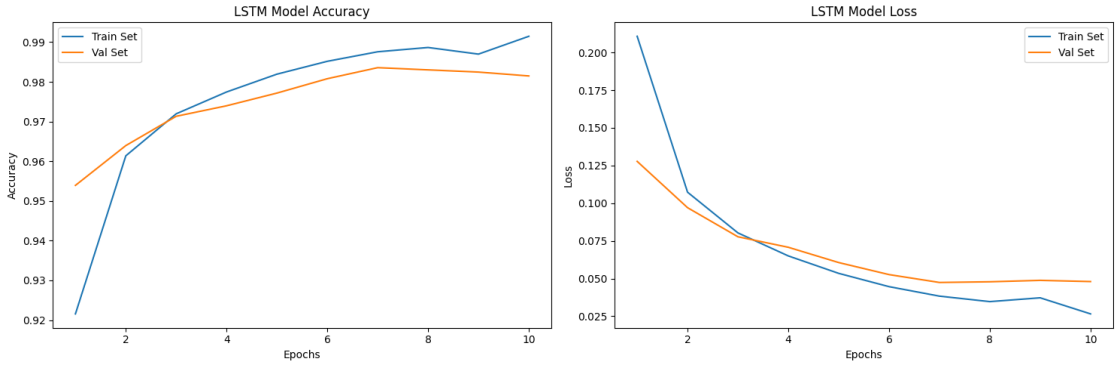
```
# create an LSTM network with a single LSTM
input_ = Input(shape=(MAX_SEQUENCE_LENGTH,))
x = embedding_layer(input_)
# x = LSTM(15, return_sequences=True)(x)
x = Bidirectional(LSTM(15, return_sequences=True))(x)
x = GlobalMaxPool1D()(x)
output = Dense(1, activation="sigmoid")(x)

model = Model(input_, output)
model.compile(
  loss='binary_crossentropy',
  optimizer='adam',
  metrics=['accuracy']
)
model.summary()
```

```
Building model...
Model: "model"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 input_1 (InputLayer)        [(None, 100)]             0

 embedding (Embedding)       (None, 100, 50)           1000000

 bidirectional (Bidirection  (None, 100, 30)           7920
 al)

 global_max_pooling1d (Glob  (None, 30)                0
 alMaxPooling1D)

 dense (Dense)               (None, 1)                 31

=================================================================
Total params: 1007951 (3.85 MB)
Trainable params: 7951 (31.06 KB)
Non-trainable params: 1000000 (3.81 MB)
_________________________________________________________________
```

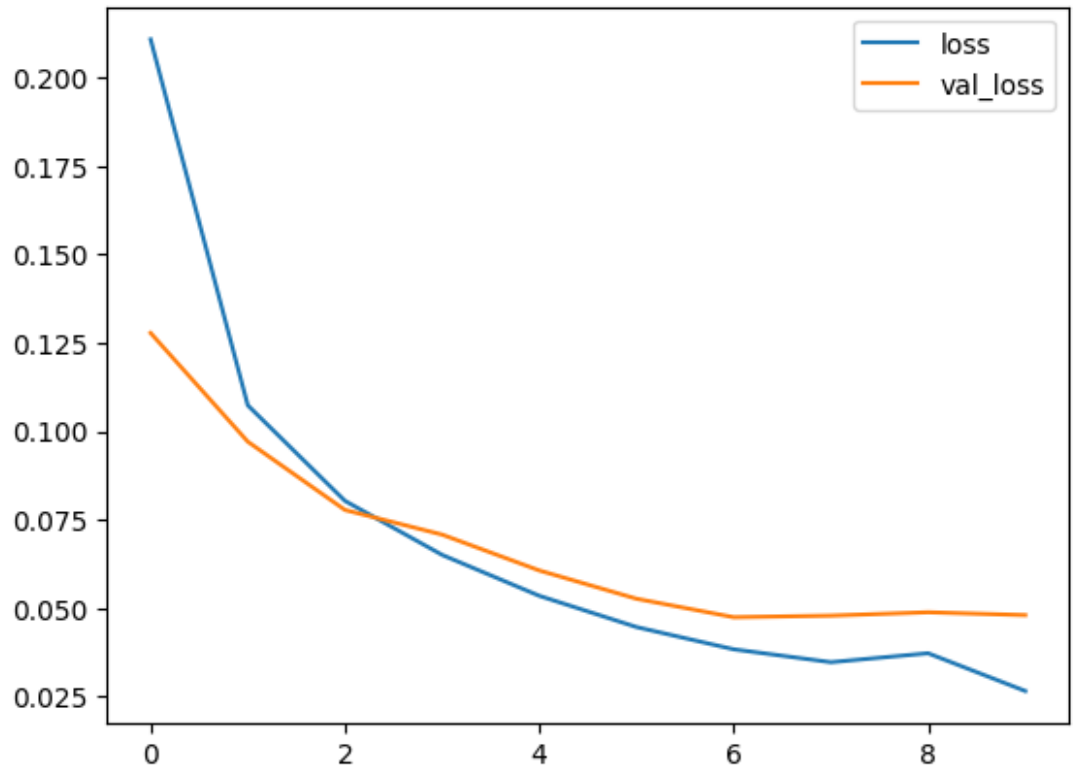
```
# train Test split in the ratio of 80:20
X_train, X_test, y_train, y_test = train_test_split(X,
y,test_size=0.20,stratify=y,random_state=0)

print('Training model...')
r = model.fit(
  X_train,
  y_train,
  batch_size=BATCH_SIZE,
  epochs=EPOCHS,
```

```
  validation_split=VALIDATION_SPLIT
)

Training model...
Epoch 1/10
898/898 [==============================] - 24s 15ms/step - loss: 0.2107 -
accuracy: 0.9216 - val_loss: 0.1278 - val_accuracy: 0.9539
Epoch 2/10
898/898 [==============================] - 16s 18ms/step - loss: 0.1073 -
accuracy: 0.9614 - val_loss: 0.0970 - val_accuracy: 0.9639
Epoch 3/10
898/898 [==============================] - 12s 14ms/step - loss: 0.0803 -
accuracy: 0.9719 - val_loss: 0.0777 - val_accuracy: 0.9713
Epoch 4/10
898/898 [==============================] - 12s 13ms/step - loss: 0.0650 -
accuracy: 0.9774 - val_loss: 0.0708 - val_accuracy: 0.9740
Epoch 5/10
898/898 [==============================] - 12s 14ms/step - loss: 0.0535 -
accuracy: 0.9819 - val_loss: 0.0606 - val_accuracy: 0.9772
Epoch 6/10
898/898 [==============================] - 13s 15ms/step - loss: 0.0446 -
accuracy: 0.9852 - val_loss: 0.0526 - val_accuracy: 0.9808
Epoch 7/10
898/898 [==============================] - 13s 14ms/step - loss: 0.0383 -
accuracy: 0.9876 - val_loss: 0.0474 - val_accuracy: 0.9836
Epoch 8/10
898/898 [==============================] - 13s 15ms/step - loss: 0.0347 -
accuracy: 0.9887 - val_loss: 0.0479 - val_accuracy: 0.9830
Epoch 9/10
898/898 [==============================] - 17s 19ms/step - loss: 0.0372 -
accuracy: 0.9870 - val_loss: 0.0488 - val_accuracy: 0.9825
Epoch 10/10
898/898 [==============================] - 12s 13ms/step - loss: 0.0266 -
accuracy: 0.9915 - val_loss: 0.0480 - val_accuracy: 0.9815

acc = r.history['accuracy']
val_acc = r.history['val_accuracy']
loss = r.history['loss']
val_loss = r.history['val_loss']
epochs_range = range(1, len(r.epoch) + 1)

plt.figure(figsize=(15,5))

plt.subplot(1, 2, 1)
plt.plot(epochs_range, acc, label='Train Set')
plt.plot(epochs_range, val_acc, label='Val Set')
plt.legend(loc="best")
plt.xlabel('Epochs')
plt.ylabel('Accuracy')
plt.title('LSTM Model Accuracy')
```

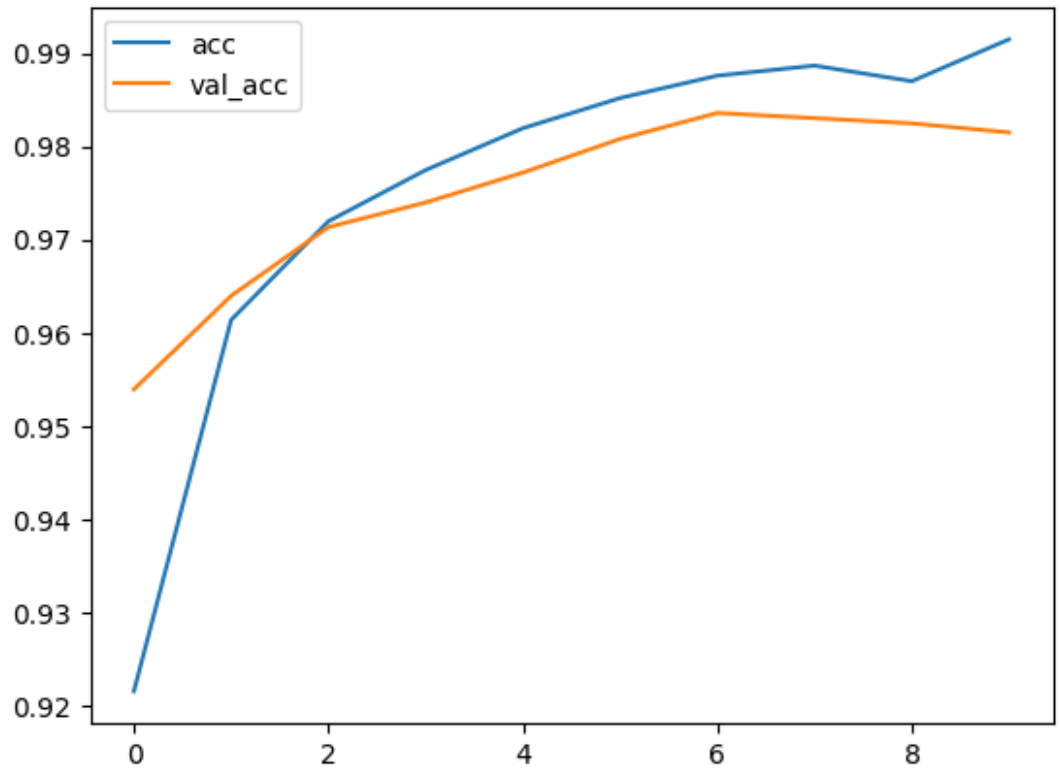
```
plt.subplot(1, 2, 2)
plt.plot(epochs_range, loss, label='Train Set')
plt.plot(epochs_range, val_loss, label='Val Set')
plt.legend(loc="best")
plt.xlabel('Epochs')
plt.ylabel('Loss')
plt.title('LSTM Model Loss')

plt.tight_layout()
plt.show()
```




```
# plot some data
plt.plot(r.history['loss'], label='loss')
plt.plot(r.history['val_loss'], label='val_loss')
plt.legend()
plt.show()
```




```
# Plotting accuracies
plt.plot(r.history['accuracy'], label='acc')
plt.plot(r.history['val_accuracy'], label='val_acc')
plt.legend()
plt.show()
```




```
print("Accuracy of the model on Training Data is - " ,
model.evaluate(X_train,y_train)[1]*100)
print("Accuracy of the model on Testing Data is - " , model.evaluate(X_test,
y_test)[1]*100)
```

```
1123/1123 [==============================] - 6s 5ms/step - loss: 0.0279 -
accuracy: 0.9899
Accuracy of the model on Training Data is -  98.98936748504639
281/281 [==============================] - 1s 5ms/step - loss: 0.0520 -
accuracy: 0.9834
Accuracy of the model on Testing Data is -  98.34075570106506
```

```
pred = model.predict(X_test)
pred[:5]
```

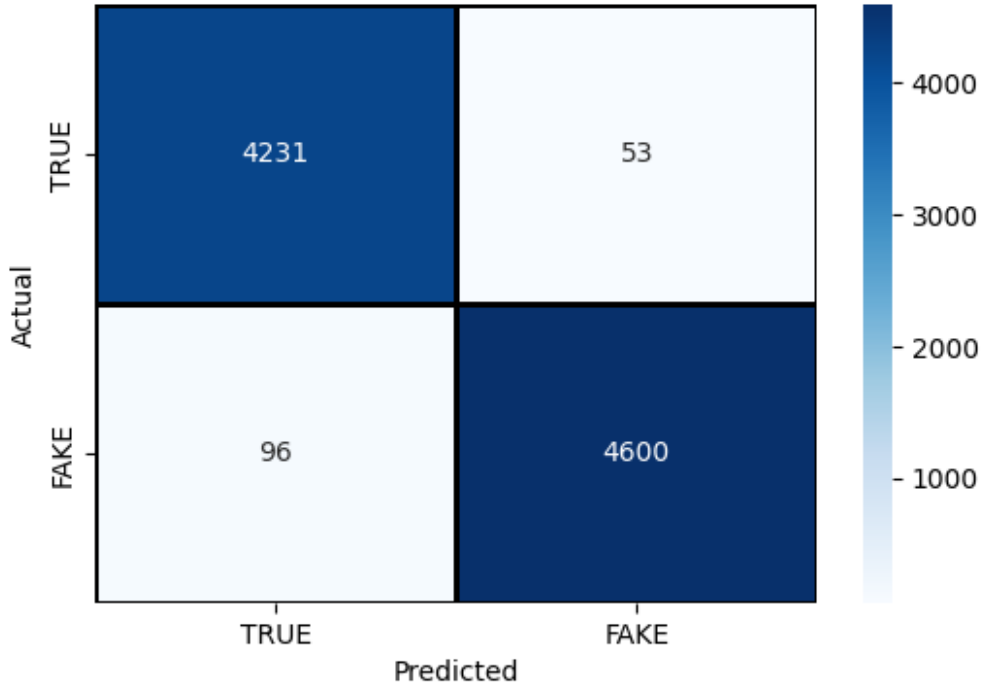
```
281/281 [==============================] - 2s 6ms/step
```

```
array([[9.9999416e-01],
       [3.1832285e-05],
       [9.9954540e-01],
       [9.9999094e-01],
       [9.9982482e-01]], dtype=float32)
```

Confusion Matrix

```
cm = confusion_matrix(y_test,pred.round())
cm = pd.DataFrame(cm , index = ['TRUE','FAKE'] , columns = ['TRUE','FAKE'])
```

```
plt.figure(figsize = (6,4))
sns.heatmap(cm,cmap= "Blues", linecolor = 'black' , linewidth = 1 , annot =
True, fmt='' , xticklabels = ['TRUE','FAKE'] , yticklabels = ['TRUE','FAKE'])
plt.ylabel('Actual')
plt.xlabel('Predicted')
plt.show()
```



Classification report

```
print(classification_report(y_test,pred.round()))
```

```
              precision    recall  f1-score   support

           0       0.98      0.99      0.98      4284
           1       0.99      0.98      0.98      4696

    accuracy                           0.98      8980
   macro avg       0.98      0.98      0.98      8980
weighted avg       0.98      0.98      0.98      8980
```

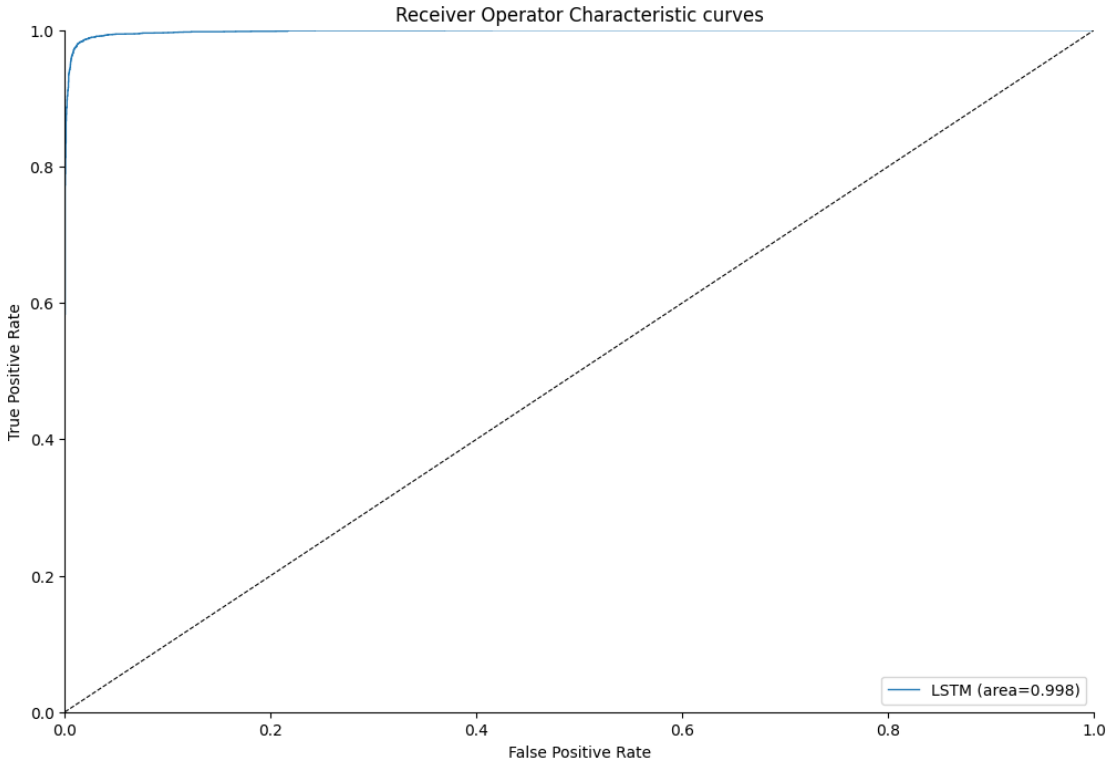
```
y_pred = model.predict(X_test).ravel()
```

```
281/281 [==============================] - 1s 5ms/step
```

ROC AUC PLOT

```
def roc_auc_plot(y_true, y_proba, label=' ', l='-', lw=1.0):
    from sklearn.metrics import roc_curve, roc_auc_score
```

```
    fpr, tpr, _ = roc_curve(y_true, y_proba)
    ax.plot(fpr, tpr, linestyle=l, linewidth=lw,
            label="%s (area=%.3f)"%(label,roc_auc_score(y_true, y_proba)))

f, ax = plt.subplots(figsize=(12,8))

roc_auc_plot(y_test,y_pred,label='LSTM', l='-')

ax.plot([0,1], [0,1], color='k', linewidth=0.8, linestyle='--',
        )
ax.legend(loc="lower right")
ax.set_xlabel('False Positive Rate')
ax.set_ylabel('True Positive Rate')
ax.set_xlim([0, 1])
ax.set_ylim([0, 1])
ax.set_title('Receiver Operator Characteristic curves')
sns.despine()
```



Model Prediction

```
testSent =["In a big win for India, the New Delhi Declaration was adopted
during the G20 Summit. Prime Minister Narendra Modi announced the adoption
and said it was made possible after cooperation from all member states.There
is good news, with everyone's cooperation, a consensus has been reached on
the New Delhi G20 Leadership Declaration,the prime minister said." ,
       "Trey Gowdy destroys this clueless DHS employee when asking about the
```

due process of getting on the terror watch list. Her response is priceless: I m sorry, um, there s not a process afforded the citizen prior to getting on the list. ","Poland s new prime minister faces a difficult balancing act trying to repair bruised relations with the European Union without alienating the eurosceptic government s core voters. A Western-educated former banker who is fluent in German and English and was sworn in on Monday, Mateusz Morawiecki boasts the credentials needed to negotiate with Brussels. But any compromises to improve relations with Brussels, which sees the ruling Law and Justice (PiS) party as a threat to democracy, would risk upsetting the traditional, Catholic supporters who propelled it into power two years ago. It is a gamble that could backfire, and it is not yet clear how far Morawiecki, 49, and his party, dominated by former Prime Minister Jaroslaw Kaczynski, are ready to go to please Brussels. The idea to build up international credibility seems rational, said Jaroslaw Flis, a sociologist at the Jagiellonian University. But such actions would have to be in complete contrast with what Mateusz Morawiecki would have to do domestically to prevent the PiS from falling apart. The PiS government has alienated many people at home and abroad with its nationalist rhetoric and changes to state institutions which the EU says subvert the bloc s laws. The European Commission, the EU executive, opened an inquiry into the rule of law in Poland in January 2016 and the European Parliament has started a process that could deprive Poland of its voting rights in the 28-nation bloc. Any hope in Brussels that Morawiecki s appointment signals a change of course by PiS will have been tempered by Polish parliament approving legal changes to the judiciary in defiance of the EU on Friday - the day after his nomination. The changes give parliament, where PiS has a majority, de facto control over the selection of judges. EU leaders looking for clues about Morawiecki s plans will also have taken little comfort from comments he has made since being nominated, making clear he backs a tough line on the EU and believes in PiS s traditional vision of the Polish state. We want to transform Europe, this is my dream, to re-christianise it, Morawiecki told the Catholic Radio Maryja broadcaster. We want Poland to be strong, but also to contain ... Christian values. We will defend them against the background of laicisation and a deepening consumerism. Asked by the radio interviewer about demands by French President Emmanuel Macron for Poland to face sanctions over a subversion of democratic rules, Morawiecki said he would not bow down to blackmail. In comments to parliament on Tuesday, Morawiecki suggested Poland might relent in a conflict with Brussels over logging in an ancient forest, which an EU court has said contravenes EU laws. But he said Poland s national interests came first in any debate over the future of the EU and that he wholeheartedly supported PiS s overhaul of the judiciary. Like Beata Szydlo, whom he replaced as prime minister, Morawiecki is likely to have to defer to PiS leader and co-founder Jaroslaw Kaczynski. Prime minister from July 2006 to November 2007, Kaczynski is widely seen as the power behind the party and Poland s main decision-maker. How much scope that will leave Morawiecki to carve out his own path remains to be seen. Former Polish President Lech Walesa, a PiS critic, has suggested that nothing of substance will change. The circus has stayed the same, only the clowns have changed their roles, Walesa, who led the Solidarity trade union movement that ended communist rule, said on Twitter. The appointment of Morawiecki, whose father founded

```
and led a radical offshoot of Solidarity in the 1980s, appears designed in
part to present a new face of Poland to the EU. Szydlo, 54, at times
responded angrily to EU criticism and relations with the bloc soured under
her government. Underlining PiS opposition to Muslim immigration, she said
last month Poland wanted to be sure Christian traditions were not subject to
ideological censorship  in the EU. Along with Hungary, Poland has refused to
take in any of its quota of the wave of refugees from Syria and elsewhere who
have come to Europe since 2015, on the grounds that Muslim immigrants are a
threat to national security and stability.         Such comments appeal to
core PiS voters, and Szydlo s government, which promised generous welfare
payouts and a dedication to traditional Catholic values, was one of Poland s
most popular since communist rule ended in 1989. A relative newcomer to
politics, Morawiecki lacks Szydlo s broad appeal. But he has overseen
significant economic achievements since becoming finance minister in 2016, a
position he has retained in the new government. Tusk has welcomed what he
sees as signs that Morawiecki is a liberal economist who wants better ties
with the EU.  There is no doubt that (Morawiecki s) liberal bias and some
pro-western gestures could be a sign that there is a lurking desire to
improve relations,  Tusk said last week. But an economic stimulus plan
Morawiecki unveiled in 2016 has been criticized by economists who say it
depends heavily on private investment, which is low in Poland despite fast
economic growth.  What Morawiecki sees as a solution, meaning more political
influence in the economy, is actually dangerous,  said Leszek Balcerowicz, a
former finance minister who coordinated the transition to a market economy
after decades of communist rule. Any hint of protectionism is also likely to
worry EU leaders, who seek to break down trade barriers.  Morawiecki has
called the privatization of state-owned companies a tragedy and said he will
give more power to domestic capital at the expense of foreign investors. In
his comments to parliament on Tuesday, he said economic policy should not
change.  ",
      ]

def cleanText(txt):
    txt = txt.lower()
    txt = ' '.join([word for word in txt.split() if word not in (stop)])
    txt = re.sub('[^a-z]',' ',txt)
    return txt

def predict_text(lst_text):
    test = tokenizer.texts_to_sequences(lst_text)
    # pad sequences so that we get a N x T matrix
    testX = pad_sequences(test, maxlen=MAX_SEQUENCE_LENGTH)
    df_test = pd.DataFrame(lst_text, columns = ['test_sent'])

    prediction = model.predict(testX)
    df_test['prediction']=prediction
    df_test["test_sent"] = df_test["test_sent"].apply(cleanText)
    df_test['prediction']=df_test['prediction'].apply(lambda x: "Fake" if
x>=0.5 else "Real")
    return df_test
```

```
#getting the prediction by passing list of sample news articles
df_testsent = predict_text(testSent)
df_testsent
```

```
1/1 [==============================] - 0s 40ms/step
```

	test_sent	prediction
0	big win india new delhi declaration adopted g...	Real
1	trey gowdy destroys clueless dhs employee aski...	Fake
2	poland new prime minister faces difficult bala...	Real

7. Conclusion

The use of deep learning model LSTM for identifying fake news from news articles has yielded remarkable results. It's truly fascinating to witness how technology can aid us in verifying the authenticity of online news. One possibility is to integrate this model into a Google Chrome extension or a separate web application. This will enable us to conveniently verify the veracity of news articles that are shared on social media platforms. It's a commendable advancement towards ensuring that we only consume truthful and trustworthy information.

8. REFERENCES:

1. News _dataset.zip - Google Drive
2. Understanding LSTM Networks -- colah's blog
3. https://kgptalkie.com/
4. https://www.kaggle.com/datasets/emineyetm/fake-news-detection-datasets