PHASE 5 : PROJECT DOCUMENTATION

INTRODUCTION

In the current digital era, the rapid and widespread dissemination of information and news articles has reached unprecedented levels. Nonetheless, this convenience in sharing information has also given rise to the rampant proliferation of false or misleading news. Fake news can yield extensive consequences, ranging from shaping public perceptions to impacting political decisions and causing harm across different sectors. To address this issue, the application of Natural Language Processing (NLP) techniques and machine learning has become increasingly crucial.

This project centers on the objective of utilizing NLP for detecting fake news. The primary aim is to create a robust model capable of distinguishing between authentic and counterfeit news articles by analyzing their textual content.

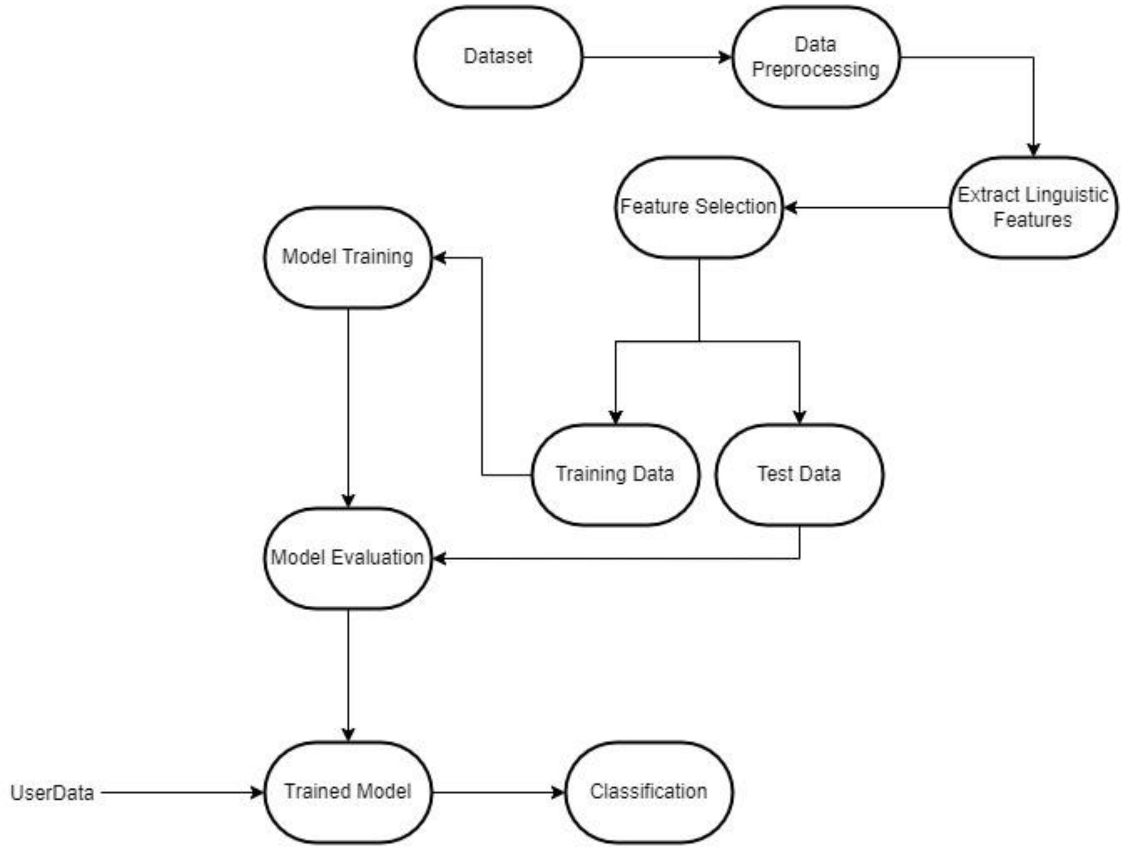
PROBLEM STATEMENT

The problem is to develop an effective fake news detection system using NLP. Given a dataset of news articles labeled as "genuine" or "fake," the goal is to build a machine learning model that can accurately classify articles based on their content. This is essential to combat the spread of misinformation in the digital age and promote informed decision-making. The project involves data collection, preprocessing, feature extraction, model development, and ethical considerations regarding fairness and bias mitigation in fake news detection.

DESIGN THINKING

INPUT LAYER:

The model's input consists of two main components:
Title Input: Represents the title of the news article.
Text Input: Represents the main textual content of the article.
These inputs are sequences of words or tokens



.

PHASES OF DEVELOPMENT

TEXT PREPROCESSING:

Before feeding the text into the model, perform text preprocessing steps, including:

- Tokenization: Splitting text into words or subword tokens.
- Embedding Layer: Converting tokens into dense word embeddings (e.g., Word2Vec or GloVe).
- Sequence Padding: Ensuring that input sequences have uniform length by padding or truncating.

FEATURE EXTRACTION:

Combine the word embeddings from the title and text inputs to create a unified representation of the article.
Optionally, you can apply techniques like attention mechanisms to give more weight to specific parts of the text.

MODEL LAYERS:

The core of the architecture consists of multiple layers of neural networks. Here's a common setup:

- Bidirectional LSTM/GRU Layers: These layers capture contextual information from both directions of the input sequence, helping the model understand the context of words in a sentence.
- Convolutional Layers (Optional): These layers can capture local patterns in the text, particularly useful for short phrases or titles.
- Attention Layers (Optional): To give varying levels of importance to different parts of the text.
- Dense Layers: The fully connected layers that combine the information from previous layers.
- Output Layer: Typically, a single neuron with a sigmoid activation function for binary classification ("genuine" or "fake").

MODEL TRAINING:

Split the dataset into training, validation, and test sets.
Train the model using appropriate loss functions (e.g., binary cross-entropy) and optimization algorithms (e.g., Adam or SGD).
Monitor the model's performance on the validation set and use techniques like early stopping to prevent overfitting

EVALUATION:

Assess the model's performance using various metrics, including accuracy, precision, recall, F1-score, and ROC-AUC, to measure its ability to classify news articles correctly.

DEPLOYMENT:

If the model meets the desired performance criteria, deploy it in a real-world setting, such as integrating it into a news verification platform or social media platform for automated fake news detection.

ETHICAL CONSIDERATIONS:

Throughout the model development process, consider ethical aspects, such as bias mitigation and fairness, to ensure that the model's predictions are unbiased and trustworthy.

DATASET:

The dataset available at the following Kaggle link: Fake and Real News Dataset consists of two main components: real news articles and fake news articles. This dataset is used for the task of distinguishing between genuine and fake news articles based on their content. Below is a detailed description of this dataset:

REAL NEWS ARTICLES:

This portion of the dataset comprises a collection of real news articles that were retrieved from reputable news sources. These articles provide a representation of legitimate and fact-based news content. The real news articles cover a wide range of topics, including politics, international affairs, economics, science, and more. Each real news article is typically represented as a text document with a title and the main textual content of the article.
The articles in this category serve as a reference for genuine news content.

FAKE NEWS ARTICLES:

This portion of the dataset comprises a collection of real news articles that were retrieved from reputable news sources. These articles provide a representation of legitimate and fact-based news content. The real news articles cover a wide range of topics, including politics, international affairs, economics, science, and more. Each real news article is typically represented as a text document with a title and the main textual content of the article. The articles in this category serve as a reference for genuine news content.

DATASET ORGANIZATION:

After the concatenation of the two datasets into a single data frame we have the shape of the dataset to be : (44898, 5)

WHY LSTM?

Long Short-Term Memory (LSTM) networks are a powerful choice for fake news detection due to their unique ability to handle sequential data effectively. News articles often have a structured narrative that unfolds over time, and LSTMs can capture the contextual dependencies and relationships within the text. In the context of fake news, these dependencies are crucial for discerning subtle patterns and inconsistencies in the language and content of articles. LSTMs excel at modeling long-range dependencies, making them well-suited to tasks where understanding the context and order of words is essential.

Additionally, fake news detection often involves analyzing vast amounts of textual data, and LSTMs are capable of processing and retaining information from large text sequences. They can learn to distinguish between genuine and fake news by recognizing subtle linguistic cues, such as misleading headlines, biased language, or inconsistent information.

Moreover, LSTMs can adapt to different article lengths, making them versatile for handling various news articles. They can learn to recognize key features and patterns across articles of different sizes.

DEVELOPEMENT:

LIBRARIES USED:

For going on with our project we use and therefore import the following libraries.

- Tensorflow
- pandas
- numpy
- matplotlib.pyplo
- seaborn
- nltk
- re

STEPS IMPLEMENTED:

IMPORTING LIBRARIES

```
[1]: # import modules
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import re
import nltk
import warnings
import sklearn
%matplotlib inline

warnings.filterwarnings('ignore')
```

LOADING THE DATASET

```
[2]: true_data = pd.read_csv('True.csv')
fake_data = pd.read_csv('Fake.csv')

true_data['Target'] = ['True'] * len(true_data)
fake_data['Target'] = ['Fake'] * len(fake_data)

# Concatenate the data frames using pd.concat
data = pd.concat([true_data, fake_data]).sample(frac=1).reset_index(drop=True)
print(data.shape)
data.head()
```

(44898, 5)

[2]:

	title	text	subject	date	Target
0	Rights groups urge EU, Japan to consider halt ...	BANGKOK (Reuters) - Rights groups on Wednesday...	worldnews	October 18, 2017	True
1	WATCH: IRRELEVANT DEM POLITICAL ANALYST James ...	On Friday s broadcast of HBO s Real Time, fo...	left-news	Oct 21, 2017	Fake
2	Trump Asks O'Reilly, 'Do you think our country...	21st Century Wire says Regardless of what one ...	US_News	February 6, 2017	Fake
3	Factbox: Trump fills top jobs for his administ...	(Reuters) - U.S. President-elect Donald Trump ...	politicsNews	December 5, 2016	True
4	ONE LAST TIME ON OUR DIME: Mooch and Barack Ar...	The hard working First Family, in need of an...	politics	Aug 6, 2016	Fake

```
[3]: data['title'][0]
```

```
[3]: 'Rights groups urge EU, Japan to consider halt in funding for  Cambodian election'
```

```
[4]: data['text'][0]
```

```
[4]: 'BANGKOK (Reuters) - Rights groups on Wednesday urged the European Union and Japan to consider halting their funding for the election panel in Cambodia,
if the ruling party succeeds in a bid to dissolve the main opposition party ahead of next year s general election. The ruling Cambodia People s Party (C
PP) has launched a crackdown on its critics, including politicians, independent media and non-government bodies. Nearly half the opposition members of p
arliament have fled abroad since September. In a session boycotted by the opposition, Cambodia s parliament voted on Monday to change party laws to re-d
istribute seats if a party is dissolved. The measure came after the government filed a lawsuit this month seeking to dissolve the main opposition Cambod
ia National Rescue Party (CNRP).  If the government s position to dissolve the opposition Cambodia National Rescue Party succeeds, next year s election
will be a joke,  Phil Robertson, deputy director for Asia at New York-based group Human Rights Watch, told Reuters.  At that point, both the EU and Japa
n should face reality and terminate their financial and technical assistance to avoid lending credibility to what will be a charade of democracy,  he ad
ded, speaking after a news conference in Bangkok. Japan and the EU are the two biggest foreign funders of the 2018 vote. China and the United States hav
e also contributed, with the United States providing trucks and technical support, while Japan has given computers. Japan s embassy in Phnom Penh did no
t reply to a Reuters request for comment on the matter. George Edgar, head of the EU delegation to Cambodia, said the EU remains  ready to support a cre
dible electoral process  but added that the polls should only go ahead with the opposition s involvement. He urged Cambodian authorities not to go ahead
with the dissolution of the opposition party.  The EU remains ready to support a credible electoral process up to the National Assembly election in 201
8. However we do not believe that a process from which the main opposition party was arbitrarily excluded could be seen as legitimate,  Edgar told Reute
rs. Cambodia s election commission was not immediately available for comment. The CPP has used  dirty tricks  to jail opposition leaders and force other
s into exile, Robertson added. CNRP leader Kem Sokha was arrested on Sept. 3 and charged with treason after the government said he had conspired with fo
reign advisers to topple it. Cambodia is relying on a partisan judiciary to silence critics and  dismantle democracy , the Geneva-based International Co
mmission of Jurists (ICJ) said in a report on Wednesday.      Old laws are being dusted off and new laws are being created,  Kingsley Abbot, the group s
southeast Asia adviser, told reporters.  And all of these have led to a list of seemingly politicized investigations.   In recent months, the government
of Prime Minister Hun Sen, who has ruled Cambodia for more than 30 years, has revoked the licenses of about 15 independent radio stations. The Cambodia
Daily, an English-language newspaper, was also forced to shut in September following government allegations of non-payment of millions of dollars in tax
es.  Hun Sen has the backing of Beijing, which says it supports Cambodia s right to maintain its national security. '
```

```
[5]: data.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 44898 entries, 0 to 44897
Data columns (total 5 columns):
 #   Column   Non-Null Count  Dtype 
---  ------   --------------  ----- 
 0   title    44898 non-null  object
 1   text     44898 non-null  object
 2   subject  44898 non-null  object
 3   date     44898 non-null  object
 4   Target   44898 non-null  object
dtypes: object(5)
memory usage: 1.7+ MB
```

PREPROCESSING:

Preprocessing is essential to clean and prepare your text data for modeling. Common preprocessing steps include:

- Removing punctuation: This helps in normalizing the text and reducing dimensionality.
- Converting to lowercase: Ensures uniformity in text data.
- Lemmatization/Stemming: Reduces words to their base forms, e.g., "running" to "run," to handle variations of words.
- Removing stop words: Common words (e.g., "the," "and") that don't carry much information are removed.

In EDA , we remove the unwanted columns and merge both the true and fake news dataset into a single dataframe and add a target class column to indicate whether the news is real or fake.

REMOVING NULL VALUES:

(from the information above there is no null values so nothing is removed)

```
[6]: #preprocessing
     #drop null values
     data=data.dropna(axis=0)
```

```
[7]: len(data)
```

```
[7]: 44898
```

CONVERTING ALL STRINGS TO LOWERCASE:

```
[8]: #converting all strings to lowercase
data['clean_news']=data['text'].str.lower()
data['clean_news']
```

```
[8]: 0        bangkok (reuters) - rights groups on wednesday...
1        on friday s broadcast of hbo s  real time,  fo...
2        21st century wire says regardless of what one ...
3        (reuters) - u.s. president-elect donald trump ...
4        the  hard working  first family, in need of an...
                               ...
44893    21st century wire says does the american ideal...
44894    barinas, venezuela (reuters) - tirelessly trav...
44895    phnom penh (reuters) - cambodian prime ministe...
44896    geneva (reuters) - the united states wants to ...
44897    beijing (reuters) - u.s. president donald trum...
Name: clean_news, Length: 44898, dtype: object
```

REMOVE SPECIAL CHARACTER, EXTRA SPACES AND ESCAPE CHARACTER :

```
[9]: #removing special characters , extra spaces and escape characters
data['clean_news']=data['clean_news'].str.replace('[^A-Za-z0-9\s]','')
data['clean_news']=data['clean_news'].str.replace('[\n]','')
data['clean_news']=data['clean_news'].str.replace('[\s+]',' ')
data['clean_news']
```

```
[9]: 0        bangkok (reuters) - rights groups on wednesday...
1        on friday s broadcast of hbo s  real time,  fo...
2        21st century wire says regardless of what one ...
3        (reuters) - u.s. president-elect donald trump ...
4        the  hard working  first family, in need of an...
                               ...
44893    21st century wire says does the american ideal...
44894    barinas, venezuela (reuters) - tirelessly trav...
44895    phnom penh (reuters) - cambodian prime ministe...
44896    geneva (reuters) - the united states wants to ...
44897    beijing (reuters) - u.s. president donald trum...
Name: clean_news, Length: 44898, dtype: object
```

REMOVING STOP WORDS :

```
[10]: nltk.download('stopwords')

[nltk_data] Downloading package stopwords to
[nltk_data]     C:\Users\rsriv\AppData\Roaming\nltk_data...
[nltk_data]   Package stopwords is already up-to-date!

[10]: True
```

```
[11]: #remove stop words
from nltk.corpus import stopwords
stop=stopwords.words('english')
data['clean_news']=data['clean_news'].apply(lambda x: " ".join([word for word in x.split() if word not in stop]))
data.head()
```

[11]:

	title	text	subject	date	Target	clean_news
0	Rights groups urge EU, Japan to consider halt ...	BANGKOK (Reuters) - Rights groups on Wednesday...	worldnews	October 18, 2017	True	bangkok (reuters) - rights groups wednesday ur...
1	WATCH: IRRELEVANT DEM POLITICAL ANALYST James ...	On Friday s broadcast of HBO s Real Time, fo...	left-news	Oct 21, 2017	Fake	friday broadcast hbo real time, former clinton...
2	Trump Asks O'Reilly, 'Do you think our country...	21st Century Wire says Regardless of what one ...	US_News	February 6, 2017	Fake	21st century wire says regardless one thinks d...
3	Factbox: Trump fills top jobs for his administ...	(Reuters) - U.S. President-elect Donald Trump ...	politicsNews	December 5, 2016	True	(reuters) - u.s. president-elect donald trump ...
4	ONE LAST TIME ON OUR DIME: Mooch and Barack Ar...	The hard working First Family, in need of an...	politics	Aug 6, 2016	Fake	hard working first family, need another taxpay...

TOKENIZATION :

```
[13]: nltk.download('punkt')
```

```
[nltk_data] Downloading package punkt to
[nltk_data]     C:\Users\rsriv\AppData\Roaming\nltk_data...
[nltk_data]   Package punkt is already up-to-date!
```

```
[13]: True
```

```
[14]: #Tokenization
from nltk.tokenize import word_tokenize
data['tokenized_news'] = data['clean_news'].apply(lambda x: word_tokenize(x))
data.head()
```

[14]:

	title	text	subject	date	Target	clean_news	tokenized_news
0	Rights groups urge EU, Japan to consider halt ...	BANGKOK (Reuters) - Rights groups on Wednesday...	worldnews	October 18, 2017	True	bangkok (reuters) - rights groups wednesday ur...	[bangkok, (, reuters,), -, rights, groups, we...
1	WATCH: IRRELEVANT DEM POLITICAL ANALYST James ...	On Friday s broadcast of HBO s Real Time, fo...	left-news	Oct 21, 2017	Fake	friday broadcast hbo real time, former clinton...	[friday, broadcast, hbo, real, time, ,, former...
2	Trump Asks O'Reilly, 'Do you think our country...	21st Century Wire says Regardless of what one ...	US_News	February 6, 2017	Fake	21st century wire says regardless one thinks d...	[21st, century, wire, says, regardless, one, t...
3	Factbox: Trump fills top jobs for his administ...	(Reuters) - U.S. President-elect Donald Trump ...	politicsNews	December 5, 2016	True	(reuters) - u.s. president-elect donald trump ...	[(, reuters,), -, u.s., president-elect, dona...
4	ONE LAST TIME ON OUR DIME: Mooch and Barack Ar...	The hard working First Family, in need of an...	politics	Aug 6, 2016	Fake	hard working first family, need another taxpay...	[hard, working, first, family, ,, need, anothe...

LEMMATIZATION :

```
[15]: nltk.download('wordnet')
```

```
[nltk_data] Downloading package wordnet to
[nltk_data]     C:\Users\rsriv\AppData\Roaming\nltk_data...
[nltk_data]   Package wordnet is already up-to-date!
```

```
[15]: True
```

```
[16]: #Lemmatization
from nltk.stem import WordNetLemmatizer
lemmatizer = WordNetLemmatizer()
def lemmatize_text(tokens, lemmatizer):
    return [lemmatizer.lemmatize(token) for token in tokens]
data['lemmatized_news'] = data['tokenized_news'].apply(lambda x: lemmatize_text(x, lemmatizer))
data.head()
```

[16]:

	title	text	subject	date	Target	clean_news	tokenized_news	lemmatized_news
0	Rights groups urge EU, Japan to consider halt ...	BANGKOK (Reuters) - Rights groups on Wednesday...	worldnews	October 18, 2017	True	bangkok (reuters) - rights groups wednesday ur...	[bangkok, (, reuters,), -, rights, groups, we...	[bangkok, (, reuters,), -, right, group, wedn...
1	WATCH: IRRELEVANT DEM POLITICAL ANALYST James ...	On Friday s broadcast of HBO s Real Time, fo...	left-news	Oct 21, 2017	Fake	friday broadcast hbo real time, former clinton...	[friday, broadcast, hbo, real, time, ,, former...	[friday, broadcast, hbo, real, time, ,, former...
2	Trump Asks O'Reilly, 'Do you think our country...	21st Century Wire says Regardless of what one ...	US_News	February 6, 2017	Fake	21st century wire says regardless one thinks d...	[21st, century, wire, says, regardless, one, t...	[21st, century, wire, say, regardless, one, th...
3	Factbox: Trump fills top jobs for his administ...	(Reuters) - U.S. President-elect Donald Trump ...	politicsNews	December 5, 2016	True	(reuters) - u.s. president-elect donald trump ...	[(, reuters,), -, u.s., president-elect, dona...	[(, reuters,), -, u.s., president-elect, dona...
4	ONE LAST TIME ON OUR DIME: Mooch and Barack Ar...	The hard working First Family, in need of an...	politics	Aug 6, 2016	Fake	hard working first family, need another taxpay...	[hard, working, first, family, ,, need, anothe...	[hard, working, first, family, ,, need, anothe...

CREATE SENTENCES TO GET CLEAN TEXT AS INPUT FOR VECTORS :

```
[17]: def return_sentences(tokenized_news):
          return " ".join([word for word in tokenized_news])
```

```
[18]: data['clean_text'] = data['lemmatized_news'].apply(lambda x : return_sentences(x))
      data.head()
```

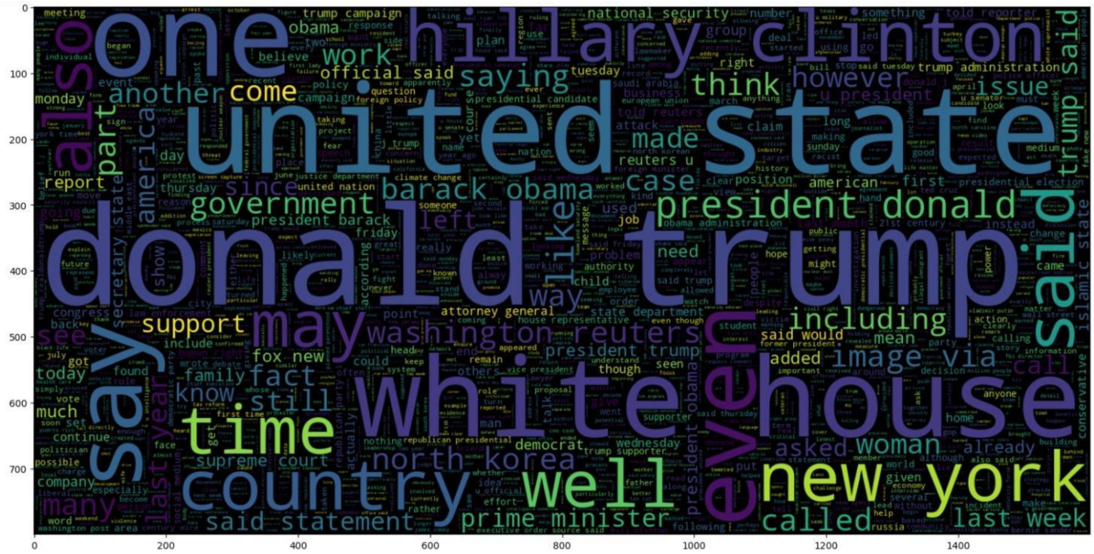
[18]:

	title	text	subject	date	Target	clean_news	tokenized_news	lemmatized_news	clean_text
0	Rights groups urge EU, Japan to consider halt ...	BANGKOK (Reuters) - Rights groups on Wednesday...	worldnews	October 18, 2017	True	bangkok (reuters) - rights groups wednesday ur...	[bangkok, (, reuters,), -, rights, groups, we...	[bangkok, (, reuters,), -, right, group, wedn...	bangkok (reuters) - right group wednesday ur...
1	WATCH: IRRELEVANT DEM POLITICAL ANALYST James ...	On Friday s broadcast of HBO s Real Time, fo...	left-news	Oct 21, 2017	Fake	friday broadcast hbo real time, former clinton...	[friday, broadcast, hbo, real, time, ,, former...	[friday, broadcast, hbo, real, time, ,, former...	friday broadcast hbo real time , former clinto...
2	Trump Asks O'Reilly, 'Do you think our country...	21st Century Wire says Regardless of what one ...	US_News	February 6, 2017	Fake	21st century wire says regardless one thinks d...	[21st, century, wire, says, regardless, one, t...	[21st, century, wire, say, regardless, one, th...	21st century wire say regardless one think don...
3	Factbox: Trump fills top jobs for his administ...	(Reuters) - U.S. President-elect Donald Trump ...	politicsNews	December 5, 2016	True	(reuters) - u.s. president-elect donald trump ...	[(, reuters,), -, u.s., president-elect, dona...	[(, reuters,), -, u.s., president-elect, dona...	(reuters) - u.s. president-elect donald trum...
4	ONE LAST TIME ON OUR DIME: Mooch and Barack Ar...	The hard working First Family, in need of an...	politics	Aug 6, 2016	Fake	hard working first family, need another taxpay...	[hard, working, first, family, ,, need, anothe...	[hard, working, first, family, ,, need, anothe...	hard working first family , need another taxpa...

PLOTTING THE WORDCLOUD FOR CLEAN TEXT :

```
from wordcloud import WordCloud
plt.figure(figsize = (20,20))
wc = WordCloud(max_words = 2000 , width = 1600 , height = 800 , stopwords = stop).generate(" ".join(data['clean_text']))
plt.imshow(wc, interpolation = 'bilinear')
```

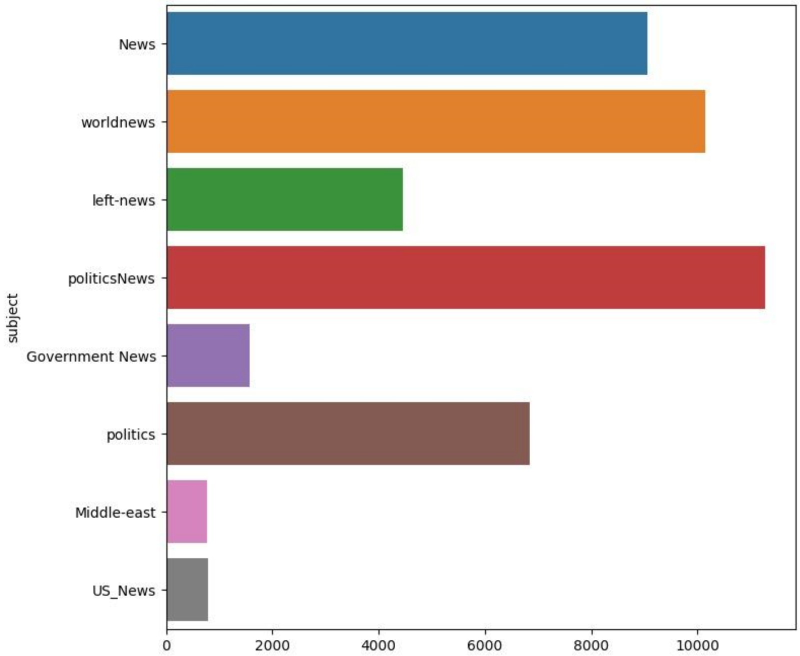
```
<matplotlib.image.AxesImage at 0x7c2e921bd2a0>
```



PLOTTING THE NUMBER OF SAMPLES IN 'SUBJECT' :

```
plt.figure(figsize = (8, 8))
sns.countplot(y = "subject", data = data)
```

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



PREPARE DATA FOR THE MODEL TO CONVERT LABEL INTO BINARY :

```
[20]: data['Target'] = [1 if x == 'Fake' else 0 for x in data['Target']]
data.head()
```

[20]:

	title	text	subject	date	Target	clean_news	tokenized_news	lemmatized_news	clean_text
0	Rights groups urge EU, Japan to consider halt ...	BANGKOK (Reuters) - Rights groups on Wednesday...	worldnews	October 18, 2017	0	bangkok (reuters) - rights groups wednesday ur...	[bangkok, (, reuters,), -, rights, groups, we...	[bangkok, (, reuters,), -, right, group, wedn...	bangkok (reuters) - right group wednesday ur...
1	WATCH: IRRELEVANT DEM POLITICAL ANALYST James ...	On Friday s broadcast of HBO s Real Time, fo...	left-news	Oct 21, 2017	1	friday broadcast hbo real time, former clinton...	[friday, broadcast, hbo, real, time, ,, former...	[friday, broadcast, hbo, real, time, ,, former...	friday broadcast hbo real time , former clinto...
2	Trump Asks O'Reilly, 'Do you think our country...	21st Century Wire says Regardless of what one ...	US_News	February 6, 2017	1	21st century wire says regardless one thinks d...	[21st, century, wire, says, regardless, one, t...	[21st, century, wire, say, regardless, one, th...	21st century wire say regardless one think don...
3	Factbox: Trump fills top jobs for his administ...	(Reuters) - U.S. President-elect Donald Trump ...	politicsNews	December 5, 2016	0	(reuters) - u.s. president-elect donald trump ...	[(, reuters,), -, u.s., president-elect, dona...	[(, reuters,), -, u.s., president-elect, dona...	(reuters) - u.s. president-elect donald trum...
4	ONE LAST TIME ON OUR DIME: Mooch and Barack Ar...	The hard working First Family, in need of an...	politics	Aug 6, 2016	1	hard working first family, need another taxpay...	[hard, working, first, family, ,, need, anothe...	[hard, working, first, family, ,, need, anothe...	hard working first family , need another taxpa...

SPLIT THE DATASET :

```
[21]: from sklearn.model_selection import train_test_split
```

```
[22]: X_train, X_test, y_train, y_test = train_test_split(data['clean_text'], data['Target'], test_size=0.2, random_state=5)

print(X_train.shape)
print(X_test.shape)
```

```
(35918,)
(8980,)
```

FEATURE EXTRACTION :

(words to vectors)

Count vectorizer which considers the frequency of occurrence of a word across the corpus.

```
[23]: from sklearn.feature_extraction.text import CountVectorizer, TfidfVectorizer

# Assuming 'Lemmatized_news' is a column in your DataFrame 'data'
count_vectorizer = CountVectorizer()
X = count_vectorizer.fit_transform(data['clean_text'])

# Get feature names
feature_names_count = count_vectorizer.get_feature_names_out()

print("CountVectorizer feature names:", feature_names_count)
```

```
CountVectorizer feature names: ['00' '000' '0000' ... 'zzzzzzzz' 'zzzzzzzzzzzzz' 'émigré']
```

TF-IDF : Term Frequency - Inverse Document Frequency

The term frequency is the number of times a term occurs in a document. Inverse document frequency is an inverse function of the number of documents in which that a given word occurs.

The product of these two terms gives tf-idf weight for a word in the corpus.

```
[26]:
tfidf_vectorizer = TfidfVectorizer()
X_tfidf = tfidf_vectorizer.fit_transform(data['clean_text'])
feature_names_tfidf = tfidf_vectorizer.get_feature_names_out()
print("TfidfVectorizer feature names:", feature_names_tfidf)
```

```
TfidfVectorizer feature names: ['00' '000' '0000' ... 'zzzzzzzz' 'zzzzzzzzzzzzz' 'émigré']
```

```
[27]:
tfidf = TfidfVectorizer()
tfidf_train = tfidf.fit_transform(X_train)
tfidf_test = tfidf.transform(X_test)

print(tfidf_train.shape)
print(tfidf_test.shape)
```

```
(35918, 106465)
(8980, 106465)
```

CREATE WORD EMBEDDINGS WITH GLOVE FILE:

```
[ ] from sklearn.metrics import classification_report, confusion_matrix, accuracy_score,roc_auc_score
    from keras.models import Model
    from keras.layers import Dense,Embedding,Input,LSTM, Bidirectional,GlobalMaxPool1D,Dropout
    from keras.preprocessing.text import Tokenizer
    from keras.preprocessing.sequence import pad_sequences
    from keras import Sequential
```

```
[ ] EMBEDDING_FILE=r"/content/drive/MyDrive/AI_Phase3/glove.6B.100d.txt"
    MAX_SEQUENCE_LENGTH=100
    MAX_VOCAB_SIZE=20000
    EMBEDDING_DIM=100
    VALIDATION_SPLIT=0.2
    BATCH_SIZE=32
    EPOCHS=10
```

LOADING THE PRETRAINED WORD VECTORS :

```
[ ] print('Loading word vectors...')
    word2vec = {}
    with open(EMBEDDING_FILE) as f:
      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 STRING INTO INTEGERS :

```
[ ] tokenizer = Tokenizer(num_words=MAX_VOCAB_SIZE)
    tokenizer.fit_on_texts(list(data['clean_text']))
    X = tokenizer.texts_to_sequences(list(data['clean_text']))

    # 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)
```

CREATE WORD-TO-INTEGER MAPPING :

```
[ ] word2idx = tokenizer.word_index
    print('Found %s unique tokens.' % len(word2idx))
```

```
Found 218659 unique tokens.
```

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...
```

EMBEDDING LAYER :

```
[ ]  embedding_layer = Embedding(
       num_words,
       EMBEDDING_DIM,
       weights=[embedding_matrix],
       input_length=MAX_SEQUENCE_LENGTH,
       trainable=False
     )
```

CREATE AN LSTM NETWORK WITH A SINGLE LSTM :

```
[ ]  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, 100)          2000000

 bidirectional (Bidirection  (None, 100, 30)           13920
 al)

 global_max_pooling1d (Glob  (None, 30)                0
 alMaxPooling1D)

 dense (Dense)               (None, 1)                 31

=================================================================
Total params: 2013951 (7.68 MB)
Trainable params: 13951 (54.50 KB)
Non-trainable params: 2000000 (7.63 MB)
_________________________________________________________________
```

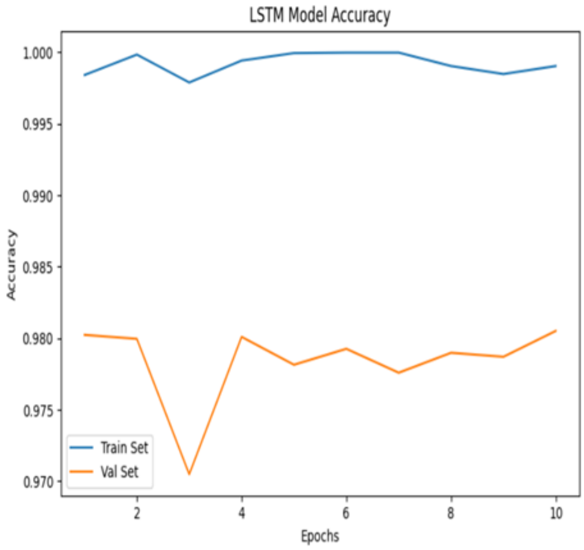
TRAIN THE MODEL :

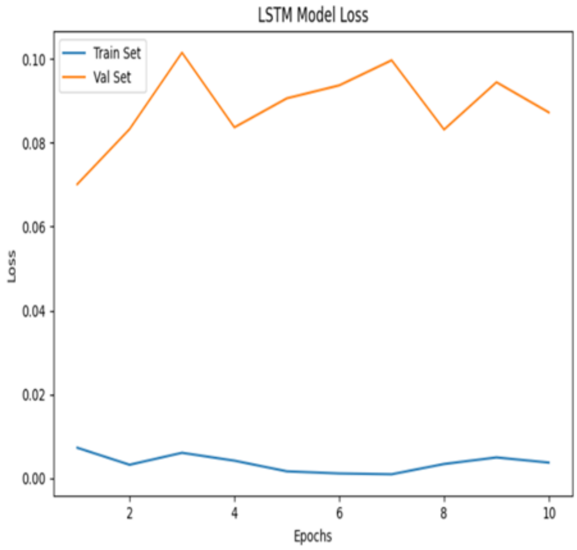
```
[ ]  print('Training model...')
     r = model.fit(
       X_train,
       y_train,
       batch_size=BATCH_SIZE,
       epochs=EPOCHS,
       validation_split=VALIDATION_SPLIT
     )
```

```
Training model...
Epoch 1/10
898/898 [==============================] - 81s 90ms/step - loss: 0.0072 - accuracy: 0.9984 - val_loss: 0.0701 - val_accuracy: 0.9802
Epoch 2/10
898/898 [==============================] - 75s 84ms/step - loss: 0.0032 - accuracy: 0.9998 - val_loss: 0.0833 - val_accuracy: 0.9800
Epoch 3/10
898/898 [==============================] - 91s 101ms/step - loss: 0.0060 - accuracy: 0.9979 - val_loss: 0.1015 - val_accuracy: 0.9705
Epoch 4/10
898/898 [==============================] - 74s 83ms/step - loss: 0.0041 - accuracy: 0.9994 - val_loss: 0.0837 - val_accuracy: 0.9801
Epoch 5/10
898/898 [==============================] - 76s 85ms/step - loss: 0.0016 - accuracy: 0.9999 - val_loss: 0.0906 - val_accuracy: 0.9781
Epoch 6/10
898/898 [==============================] - 74s 83ms/step - loss: 0.0011 - accuracy: 1.0000 - val_loss: 0.0937 - val_accuracy: 0.9793
Epoch 7/10
898/898 [==============================] - 73s 81ms/step - loss: 9.0125e-04 - accuracy: 1.0000 - val_loss: 0.0997 - val_accuracy: 0.9776
Epoch 8/10
898/898 [==============================] - 76s 85ms/step - loss: 0.0034 - accuracy: 0.9990 - val_loss: 0.0832 - val_accuracy: 0.9790
Epoch 9/10
898/898 [==============================] - 73s 81ms/step - loss: 0.0049 - accuracy: 0.9985 - val_loss: 0.0945 - val_accuracy: 0.9787
Epoch 10/10
898/898 [==============================] - 77s 86ms/step - loss: 0.0037 - accuracy: 0.9990 - val_loss: 0.0873 - val_accuracy: 0.9805
```

ACCURACY :

```
[ ]  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()
```






```
[ ] 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 [==============================] - 24s 22ms/step - loss: 0.0184 - accuracy: 0.9961
Accuracy of the model on Training Data is -  99.6074378490448
281/281 [==============================] - 6s 22ms/step - loss: 0.0740 - accuracy: 0.9835
Accuracy of the model on Testing Data is -  98.35189580917358
```

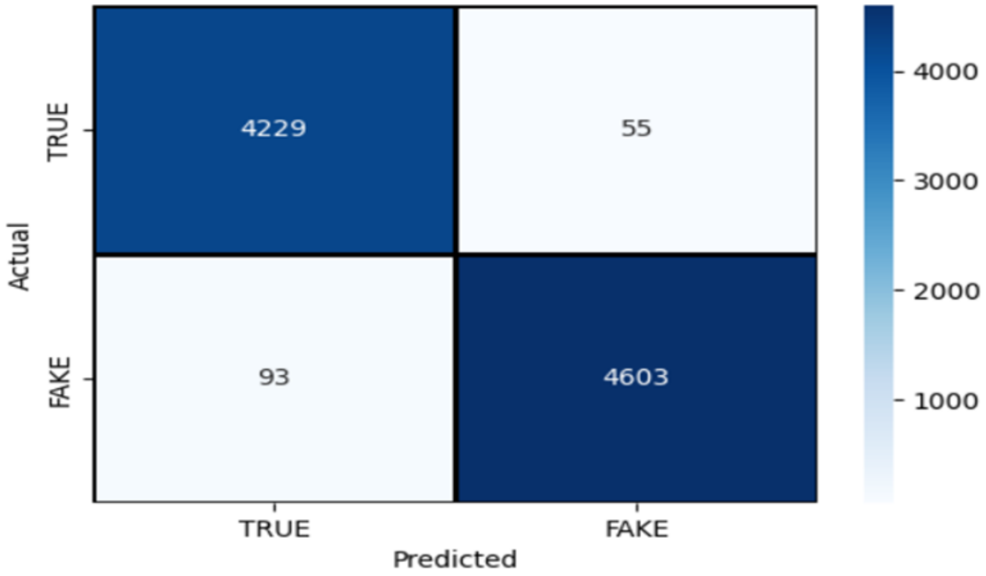
PREDICTION :

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

```
281/281 [==============================] - 9s 31ms/step
array([[1.0000000e+00],
       [2.4139799e-08],
       [9.9999994e-01],
       [1.0000000e+00],
       [1.0000000e+00]], 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()
```



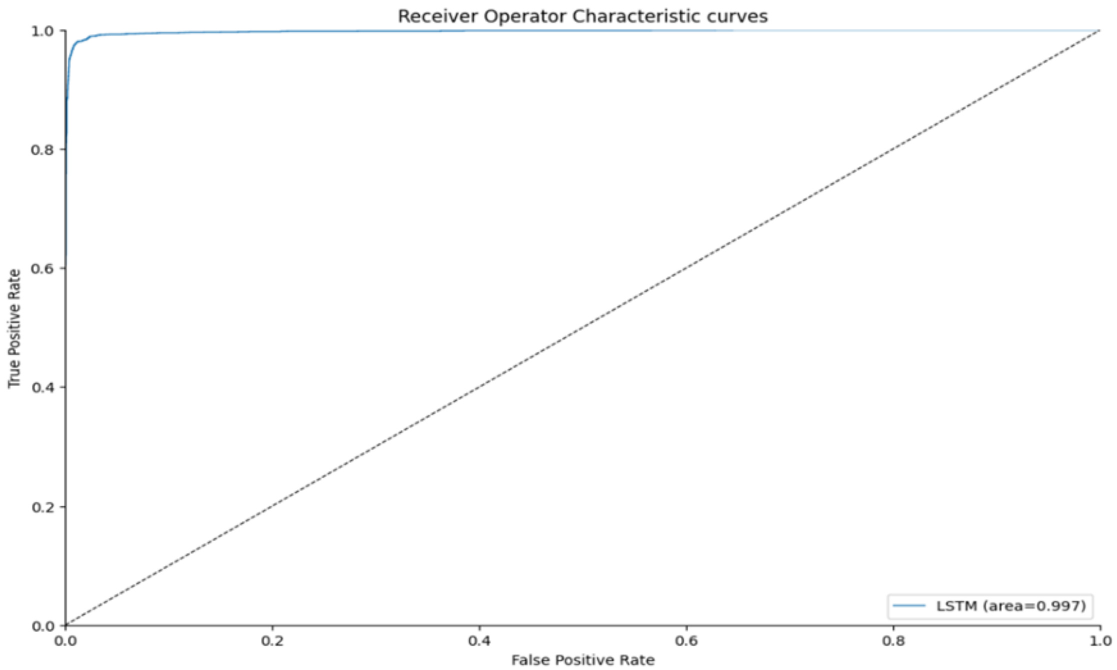
CLASSIFICATION REPORT :

```
[ ]  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 [==============================] - 9s 33ms/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 =["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 nc
          "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 vot
    ]
```

```
[ ]  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
```

PREDICT TEXT AND TOKENIZED :

```
[ ]  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
```

PREDICTION OF THE MODEL :

```
[ ]  df_testsent = predict_text(testSent)
     df_testsent
```

```
1/1 [==============================] - 0s 61ms/step
```

	test_sent	prediction
0	trey gowdy destroys clueless dhs employee aski...	Fake
1	poland new prime minister faces difficult bala...	Real

CONCLUSION:

Our goal of developing an effective fake news detection model. We've made substantial progress by carefully preprocessing text data, extracting relevant features, training a classification model, and evaluating its performance. Leveraging NLP techniques, we've ensured that the model works with high-quality, cleansed text data, enabling it to distinguish between fake and real news articles. This project contributes to the fight against misinformation, reinforcing responsible journalism, and aligning with efforts to combat fake news.