FAKE NEWS DETECTION USING NLP

Problem Statement:

Addressing this issue requires leveraging advanced technologies such as Natural Language Processing (NLP) and Artificial Intelligence (AI) to develop accurate and efficient fake news detection systems. The problem at hand involves creating an innovative AI-based solution that can effectively identify and classify fake news articles from genuine ones, utilizing the power of NLP techniques.

DESIGN THINKING:

Functionality:

Fake news detection using NLP involves analysing the content of a news piece against reliable sources or databases, sentimental analysis, semantic analysis, fact-checking, and recognizing sensationalism and clickbait.

Data Collection:

To train our model, we are going to use a large dataset of news articles, including both real and fake examples, from the Kaggle Dataset to train the AI model.

Data Preprocessing:

Since the data may involve tasks like tokenization, stop word removal, and text normalization, we are going to clean and preprocess the data.

Feature Extraction:

We are going to extract relevant features from the text data, such as word embeddings or TF-IDF vectors, to represent the articles.

Natural Language Processing (NLP):

To train our chatbot understand/recognize various user inputs it is necessary to implement NLP&NLU techniques. As the chatbot is created using python, python libraries like NLTK (natural language tool kit) and RASA-NLU can be used to analyse the user inputs. It will help the chatbot to process the output in a conversational manner.

Training:

We are going to train the selected model on the labeled dataset, adjusting hyperparameters and using techniques like cross-validation to ensure good performance.

Evaluation:

We are going to assess the model's performance using metrics like accuracy, precision, recall, and F1 score on a separate test dataset.

Fine-Tuning:

Since it is a rejection type mode, we have to refine the model and its features to improve accuracy and reduce false positives/negatives.

Deployment:

We are going to integrate the trained model into an application and develop a user-friendly interface for users to input news articles and receive detection results.

Continuous Updation:

We are going to continuously update the model with new data to adapt to evolving fake news tactics.

Finally, it’s our responsibility to announce that even though AI can assist in fake news detection, it's not foolproof, and We, human fact-checkers should remain part of the process to verify results.

Implementation of Fake News Detection:

Step 1: Data Collection and Preprocessing

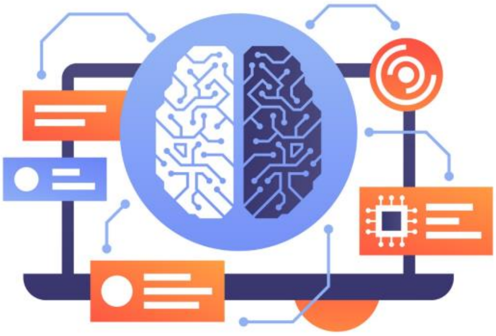
The first step is to collect a large dataset of news articles, both real and fake, and preprocess them by removing stop words, punctuation, and any other irrelevant information. This will help the AI model focus on the relevant features that distinguish real from fake news.

Step 2: Feature Extraction

Next, various features are extracted from each article, such as sentence length, word frequency, sentiment analysis, and topic modeling. These features can be used to train machine learning models to differentiate between real and fake news.

Step 3: Training Machine Learning Models

Once the relevant features have been extracted, they are fed into machine learning algorithms, such as Random Forest, Support Vector Machines (SVM), or Neural Networks, to train them to classify news articles as real or fake.



Step 4: Model Evaluation

After training the machine learning models, their performance is evaluated using metrics such as accuracy, precision, recall, and F1-score. The models are fine-tuned based on the results to improve their performance.

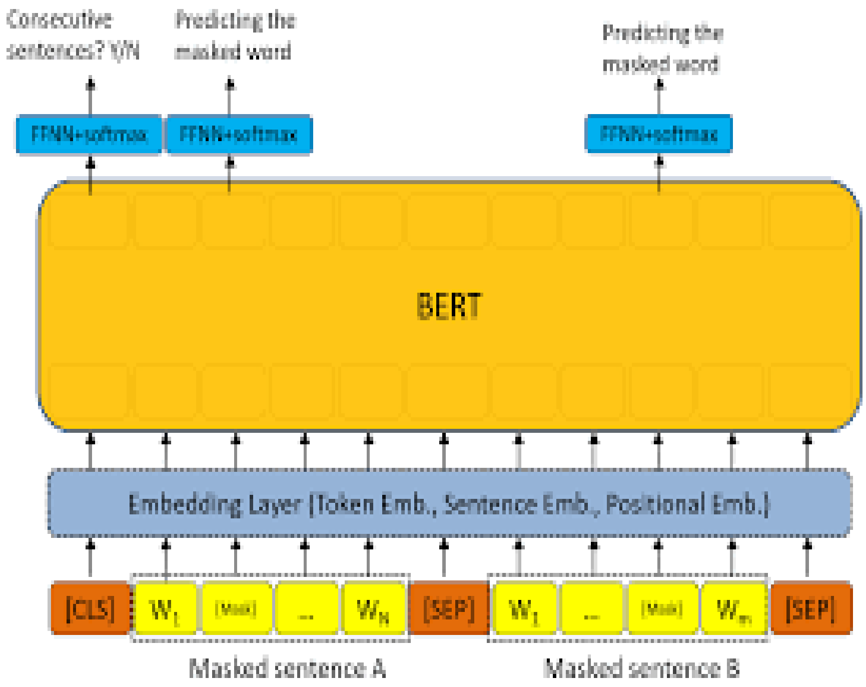
Step 5: Deployment

Finally, the trained models are deployed in a production environment where they can automatically analyze new news articles and classify them as real or fake. This can be done through web scraping tools or by integrating the models with existing news aggregator platforms.

Let us see some of the different Deep Learning Language models for Fake news Detection:

BERT:

BERT (Bidirectional Encoder Representations from Transformers) is a powerful language model developed by Google in 2018. It was designed primarily for natural language processing tasks like text classification, sentiment analysis, question-answering, and more.


Consecutive sentences? Y/N
Predicting the masked word
Predicting the masked word
FFNN+softmax
FFNN+softmax
FFNN+softmax
BERT
Embedding Layer (Token Emb., Sentence Emb., Positional Emb.)
[CLS]
W1
[Mask]
...
WN
[SEP]
W1
...
[Mask]
Wm
[SEP]
Masked sentence A
Masked sentence B

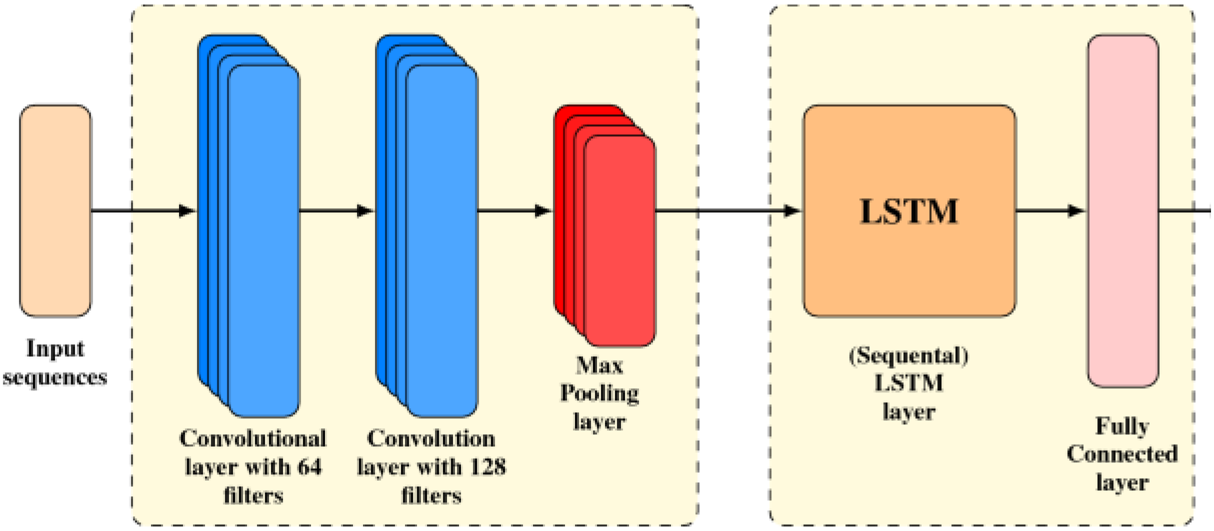
BERT uses a multi-layer bidirectional transformer encoder to generate contextualized representations of words in a sentence. These representations capture the semantic meaning of each word based on its context, which helps improve the performance of downstream NLP tasks. Unlike traditional language models that only consider the left or right context of a word, BERT can jointly encode both contexts using a self-attention mechanism. This allows it to better understand the nuances of language and handle complex sentences with ease.

One of the key innovations of BERT is its ability to fine-tune pre-trained models for specific NLP tasks. By adding task-specific output layers on top of a pre-trained BERT model, you can train the model to perform well on your desired task without requiring much additional training data.

This has led to state-of-the-art results in many NLP benchmarks and has become a de facto standard in the field.

In summary, BERT is a ground breaking language model that has revolutionized the field of natural language processing. Its ability to capture contextual semantics and fine-tune for specific tasks has made it an indispensable tool for NLP practitioners around the world.

LSTM:



LSTM (Long Short-Term Memory) is a type of Recurrent Neural Network (RNN) architecture that is commonly used for processing sequential data, such as time series data or natural language text. Unlike traditional RNNs, which have a fixed-size internal memory, LSTMs have a dynamic memory capacity that can selectively retain or forget information from previous time steps. This allows LSTMs to learn long-term dependencies in the input data more effectively than traditional RNNs.

In other words, LSTMs are designed to handle the problem of vanishing gradients that occurs when training traditional RNNs over long sequences. The vanishing gradients problem arises because the gradients of the model's parameters with respect to the loss function become smaller as they are backpropagated through time, making it difficult to train the model over long sequences. LSTMs address this problem by introducing a cell state and gates (input, output, and forget gates) that control the flow of information into and out of the cell state, allowing the model to selectively retain or forget information from previous time steps.

LSTMs have been applied to a wide range of applications, including language modeling, speech recognition, machine translation, and gesture recognition. They have also been used in conjunction with other techniques, such as attention mechanisms and convolutional neural networks (CNNs), to further improve performance.

We can use some other techniques for fake News Detection as well. Some of the techniques are,

Use of domain-specific language models

Fake news articles often contain language patterns that are different from those found in legitimate news sources.

By training domain-specific language models, we can better identify these differences and detect fake news.

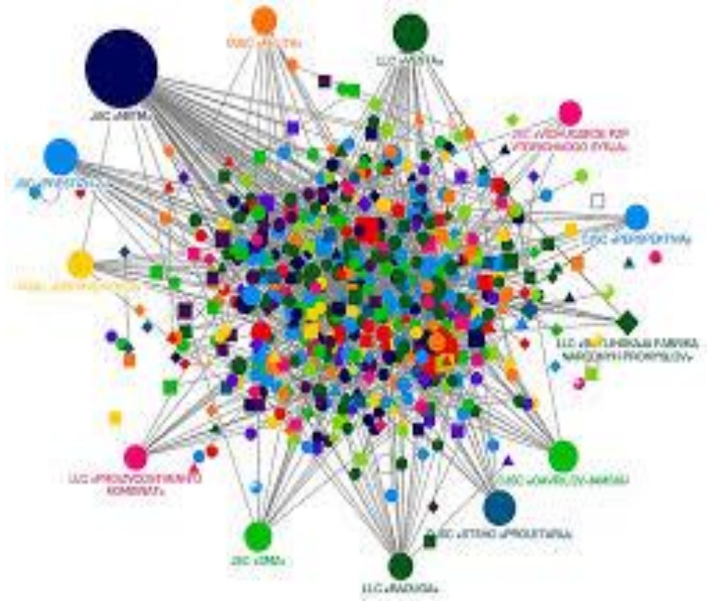
Use of fact-checking websites

Fact-checking websites like Snopes, PolitiFact, and FactCheck.org can provide valuable information about the veracity of specific news claims.

By leveraging this data, AI models can learn to associate certain phrases or topics with known false or true facts.

Use of social network analysis

Social networks can be analyzed to identify fake news campaigns.



For example, if multiple Twitter accounts are retweeting the same fake news article, it may indicate a coordinated disinformation campaign.

Use of sentiment analysis

Sentiment analysis can help distinguish between genuine news articles and those written with the intention of deceiving .

Fake news articles tend to have more negative sentiments than real news articles.

Necessary step to follow:

1. Loading the dataset:

- First, you need a dataset containing labeled examples of real and fake news articles.
- This dataset could be in CSV, JSON, or any other suitable format.You can use libraries like pandas in Python to load data from CSV or Excel files.

Program:

```
import pandas as pd # Load data

from CSV file data =

pd.read_csv('fake_news_dataset

.csv')
```

2. Exploring the dataset:

- Understand the structure of your dataset: the columns, data types, and the distribution of real vs. fake news labels.
- Use functions like **head()**, **info()**, and **describe()** in pandas to explore the dataset.

3. Text preprocessing:

- Text cleaning: Remove special characters, links, and irrelevant symbols from the text.
- Tokenization: Split the text into words or tokens.
- Lowercasing: Convert all text to lowercase to ensure consistency.
- Stopword removal: Remove common words like "and," "the," etc., as they don't carry significant meaning.
- Lemmatization or stemming: Reduce words to their base or root form to capture core meaning. **Program:**

```
import re from nltk.tokenize import word_tokenize
from nltk.corpus import stopwords from nltk.stem
import WordNetLemmatizer
def clean_text(text):
    text = re.sub(r'http\S+|www\S+|https\S+', '',
        text, flags=re.MULTILINE)
    text = re.sub(r'\s+', ' ', text)     text =
re.sub('[^A-Za-z]', ' ', text)     text =
text.lower()     words = word_tokenize(text)
words = [word for word in words if
word.isalpha()]     stop_words =
set(stopwords.words('english'))     words =
[word for word in words if word not in
stop_words]     lemmatizer =
WordNetLemmatizer()
```

```
words = [lemmatizer.lemmatize(word) for
word in words]     return ' '.join(words)
 # Apply the cleaning function to the 'text' column
data['cleaned_text'] =data['text'].apply(clean_text)
```

4. Vectorization:

Convert text data into numerical format that machine learning algorithms can understand. Common techniques include TF-IDF (Term Frequency-Inverse Document Frequency) or word embeddings like Word2Vec or GloVe.

Program: from

```
sklearn.feature_extraction.text import

TfidfVectorizer

  tfidf_vectorizer =

TfidfVectorizer(max_features=5000)  # You can

adjust the       number of features based on your dataset

size    tfidf_matrix =

tfidf_vectorizer.fit_transform(data['cleaned_text'])
```

5.Splitting the dataset:

Split the preprocessed data into training and testing sets to evaluate the model's performance.

```
test_size=0.2, random_state=42)
```

Program:

```
from sklearn.model_selection import train_test_split

X_train, X_test, y_train, y_test = train_test_split(tfidf_matrix, data['label'],
```

Importance of loading and preprocessing dataset:

Importance of loading and processing dataset: Loading and preprocessing the dataset is an important first step in building any machine learning model. However, it is especially important for house price prediction models, as house price datasets are often complex and noisy. By loading and preprocessing the dataset, we can ensure that the machine learning algorithm is able to learn from the data effectively and accurately.

HOW TO OVERCOME THE CHALLENGES OF LOADING AND PREPROCESSING A FAKE NEWS DETECTION USION NLP:

1.Lowercase text & URL removal:

Before we start any of the pre-processing heavy lifting, we want to convert our text to lowercase and remove any URLs in our text. A simple regex expression can handle this for us.

Program:

```
Import re
```

```
 text  =  "http://www.google.com
hello  world"              text  =
re.sub(r'http\S+',  '',  text.lower())
print(text)
```

OUTPUT:

```
 hello world
```

2.Split contractions:

Similar to URLs, contractions can produce unintended results if left alone. The aptly named contractions python library to the rescue! It looks for contractions and splits them into root words.

Program:

```
Importcontractions def
                    remove_contractions(t
                    ext):
                        return ' '.join([contractions.fix(word)
                    for word in text.split()])

                    text = """can't won't shouldn't there's
                    mustn't"""
                    print(remove_contractions(tex
                    t)) OUTPUT:
```

```
can not will not should not there is must not
```

3. Tokenization:

Tokenization is breaking the raw text into small chunks.Tokenization breaks the raw text into words, sentences called tokens. These tokens help in understanding the context or developing the model for the NLP. The tokenization helps in interpreting the meaning of the text by analyzing the sequence of the words.

There are a multitude of ways to implement tokenization and their approaches varied. For our project we utilized RegexpTokenizer within the NLTK library. Using regular expressions, RegexpTokenizer will match either tokens or separators (i.e. include or exclude regex matches).

Program:

```
import nltk    from
nltk.tokenize import
RegexpTokenizer      # Create
tokens out of alphanumeric
characters    tokenizer =
RegexpTokenizer(r'\w+')
   tokens = tokenizer.tokenize("I think pineapple pizza
is gross and not worth $15!")    print(tokens)
```

OUTPUT:

```
 ['I', 'think', 'pineapple', 'pizza', 'is', 'gross', 'and', 'not',
'worth', '15']
```

4. Stemming:

Text normalization is the process of simplifying multiple variations or tenses of the same word. Stemming and lemmatization are two methods of text normalization, the former being the simpler of the two. To stem a word, we simply remove the suffix of a word to reduce it to its root.

Program:

```
 #using porter stemmer

implementation in nltk

from nltk.stem import

PorterStemmer    stemmer =

PorterStemmer()       def

stem(tokens):

          return [stemmer.stem(token) for token

in tokens]    tokens = ['jumped', 'jumps',

'jumped']     print(stem(tokens))
```

OUTPUT:

```
['jump', 'jump', 'jump']
```

As an example, “jumping”, “jumps”, and “jumped” all are stemmed to “jump.”

Stemming is not without its faults, however. We can run into the issue of *overstemming*. Overstemming is when words with different meanings are stemmed to the same root — a false positive.

Program:

```
token=[‘universal’, ‘university’, ‘universe’]
    print(stem(tokens))
```

OUTPUT:

```
['univers', 'univers', 'univers']
```

Understemming is also a concern. See how words that should stem to the same root do not — a false negative.

Let’s take a look at a more nuanced approach to text normalization, lemmatization.

Program:

```
token=[‘alumnus’,’alumni’,’alumnae’]

    print(stem(tokens))
```

OUTPUT:

```
    ['alumnu', 'alumni', 'alumna']
```

5.Lemmatization:

Lemmatization is the process of converting a word to its base form. The difference between stemming and lemmatization is, lemmatization considers the context and converts the word to its meaningful base form, whereas stemming just removes the last few characters, often leading to incorrect meanings and spelling errors.

Lemmatization differs from stemming in that it determines a words part of speech by looking at surrounding words for context. For this example we use nltk.pos_tag to assign parts of speech to tokens. We then pass the token and its assigned tag into WordNetLemmatizer, which decides how to lemmatize the token.

Program:

```
import nltk

          from nltk.corpus import
         wordnet  lemmatizer =
         WordNetLemmatizer()
          # Convert the nltk pos tags to tags that
         wordnet can recognize  def
         nltkToWordnet(nltk_tag):   if
         nltk_tag.startswith('J'):     return
         wordnet.ADJ   elif nltk_tag.startswith('V'):
             return
         wordnet.VERB   elif
         nltk_tag.startswith('
         N'):     return
         wordnet.NOUN
         elif
         nltk_tag.startswith('
         R'):
             return
         wordnet.ADV
         else:
             return None
             # Lemmatize a list of
         words/tokens  def
         lemmatize(tokens):
           pos_tags = nltk.pos_tag(tokens)       res_words
         = []   for
```

```
word, tag in pos_tags:
tag = nltkToWordnet(ta
g)          if tag is None:

res_words.append(word
)      else:
res_words.append(lem
matizer.lemmatize(word
, tag))    return
res_words
```

Using the following text we can compare the results of our approaches to stemming and lemmatization.

Program:

text=”it takes a great deal of bravery to stand up to our enemies, but just as much to stand up to our friends”

```
# STEMMING RESULTS

print(stem(tokens))
```

OUTPUT:

```
 ['it', 'take', 'a', 'great', 'deal', 'of', 'braveri', 'to', 'stand',
'up', 'to',
           'our', 'enemi', 'but', 'just', 'as', 'much', 'to',
           'stand', 'up', 'to', 'our', 'friend']
```

```
# LEMMATIZING RESULTS

print(lemmatize(tokens))

OUTPUT:

['it', 'take', 'a', 'great', 'deal', 'of', 'bravery', 'to', 'stand',
'up', 'to',
    'our', 'enemy', 'but', 'just', 'as', 'much', 'to',
    'stand', 'up', 'to', 'our', 'friend']
```

Notice that ‘enemies’ was stemmed to ‘enemi’ but lemmatized to ‘enemy’. Interestingly, ‘bravery’ was stemmed to ‘braveri’ but the lemmatizer did not change the original word. In general, lemmatization is more precise, but at the expense of complexity.

6.Stop Word Removal:

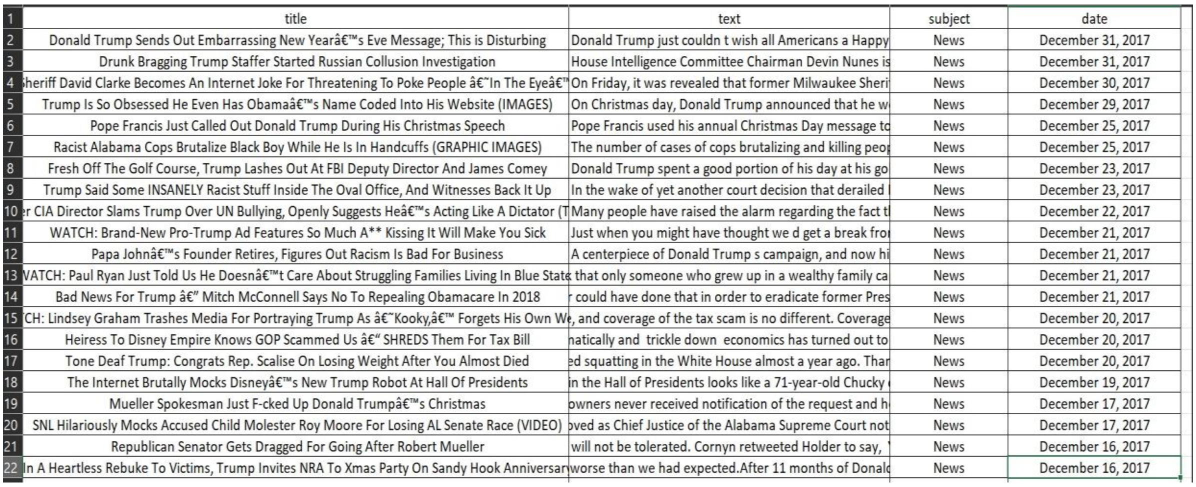
Stop words are words in the text which do not add any meaning to the sentence and their removal will not affect the processing of text for the defined purpose. They are removed from the vocabulary to reduce noise and to reduce the dimension of the feature set.

Program: Import nltk

nltk.download('words') #download list of english words nltk.download('stopwords') #download list of stopwords from nltk.corpus import stopwords stopWords = stopwords.words('english') englishWords = set(nltk.corpus.words.words()) def remove_stopWords(tokens):

Dataset description:

Dataset: [Fake news detection]

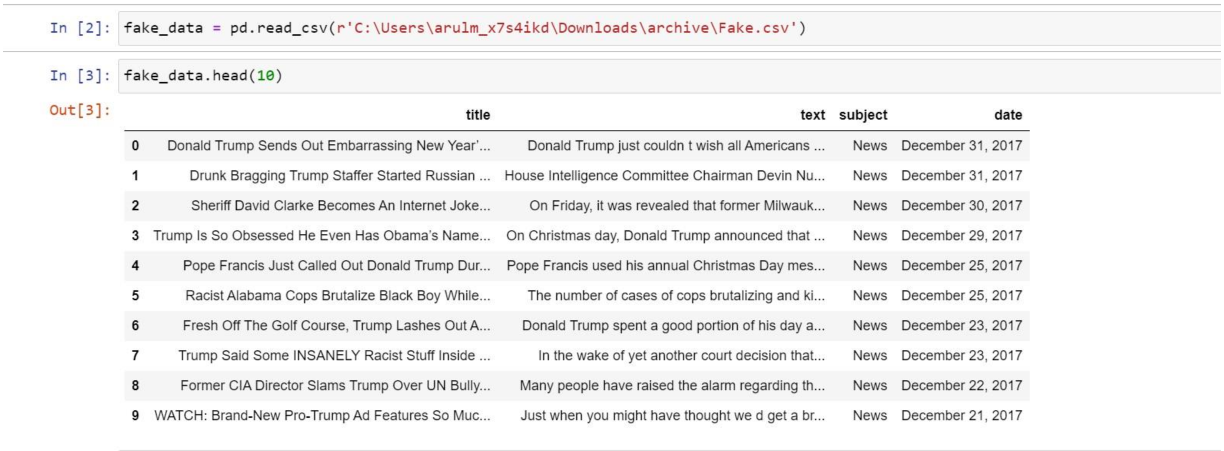


1	title	text	subject	date
2	Donald Trump Sends Out Embarrassing New Yearâ€™s Eve Message; This is Disturbing	Donald Trump just couldn t wish all Americans a Happy	News	December 31, 2017
3	Drunk Bragging Trump Staffer Started Russian Collusion Investigation	House Intelligence Committee Chairman Devin Nunes is	News	December 31, 2017
4	heriff David Clarke Becomes An Internet Joke For Threatening To Poke People â€˜In The Eyeâ€™	On Friday, it was revealed that former Milwaukee Sheri	News	December 30, 2017
5	Trump Is So Obsessed He Even Has Obamaâ€™s Name Coded Into His Website (IMAGES)	On Christmas day, Donald Trump announced that he w	News	December 29, 2017
6	Pope Francis Just Called Out Donald Trump During His Christmas Speech	Pope Francis used his annual Christmas Day message to	News	December 25, 2017
7	Racist Alabama Cops Brutalize Black Boy While He Is In Handcuffs (GRAPHIC IMAGES)	The number of cases of cops brutalizing and killing peo	News	December 25, 2017
8	Fresh Off The Golf Course, Trump Lashes Out At FBI Deputy Director And James Comey	Donald Trump spent a good portion of his day at his go	News	December 23, 2017
9	Trump Said Some INSANELY Racist Stuff Inside The Oval Office, And Witnesses Back It Up	In the wake of yet another court decision that derailed	News	December 23, 2017
10	r CIA Director Slams Trump Over UN Bullying, Openly Suggests Heâ€™s Acting Like A Dictator (T	Many people have raised the alarm regarding the fact t	News	December 22, 2017
11	WATCH: Brand-New Pro-Trump Ad Features So Much A** Kissing It Will Make You Sick	Just when you might have thought we d get a break fro	News	December 21, 2017
12	Papa Johnâ€™s Founder Retires, Figures Out Racism Is Bad For Business	A centerpiece of Donald Trump s campaign, and now hi	News	December 21, 2017
13	VATCH: Paul Ryan Just Told Us He Doesnâ€™t Care About Struggling Families Living In Blue State	that only someone who grew up in a wealthy family ca	News	December 21, 2017
14	Bad News For Trump â€” Mitch McConnell Says No To Repealing Obamacare In 2018	r could have done that in order to eradicate former Pres	News	December 21, 2017
15	CH: Lindsey Graham Trashes Media For Portraying Trump As â€˜Kooky,â€™ Forgets His Own We	, and coverage of the tax scam is no different. Coverage	News	December 20, 2017
16	Heiress To Disney Empire Knows GOP Scammed Us â€“ SHREDS Them For Tax Bill	natically and trickle down economics has turned out to	News	December 20, 2017
17	Tone Deaf Trump: Congrats Rep. Scalise On Losing Weight After You Almost Died	ed squatting in the White House almost a year ago. Than	News	December 20, 2017
18	The Internet Brutally Mocks Disneyâ€™s New Trump Robot At Hall Of Presidents	in the Hall of Presidents looks like a 71-year-old Chucky	News	December 19, 2017
19	Mueller Spokesman Just F-cked Up Donald Trumpâ€™s Christmas	owners never received notification of the request and h	News	December 17, 2017
20	SNL Hilariously Mocks Accused Child Molester Roy Moore For Losing AL Senate Race (VIDEO)	oved as Chief Justice of the Alabama Supreme Court not	News	December 17, 2017
21	Republican Senator Gets Dragged For Going After Robert Mueller	will not be tolerated. Cornyn retweeted Holder to say,	News	December 16, 2017
22	In A Heartless Rebuke To Victims, Trump Invites NRA To Xmas Party On Sandy Hook Anniversar	worse than we had expected.After 11 months of Donald	News	December 16, 2017

Features: [List key features such as Title, text , subject , date]

Data Preprocessing:

There are a number of libraries available that can help with data preprocessing tasks, such as handling missing values, encoding categorical variables, and scaling the features.



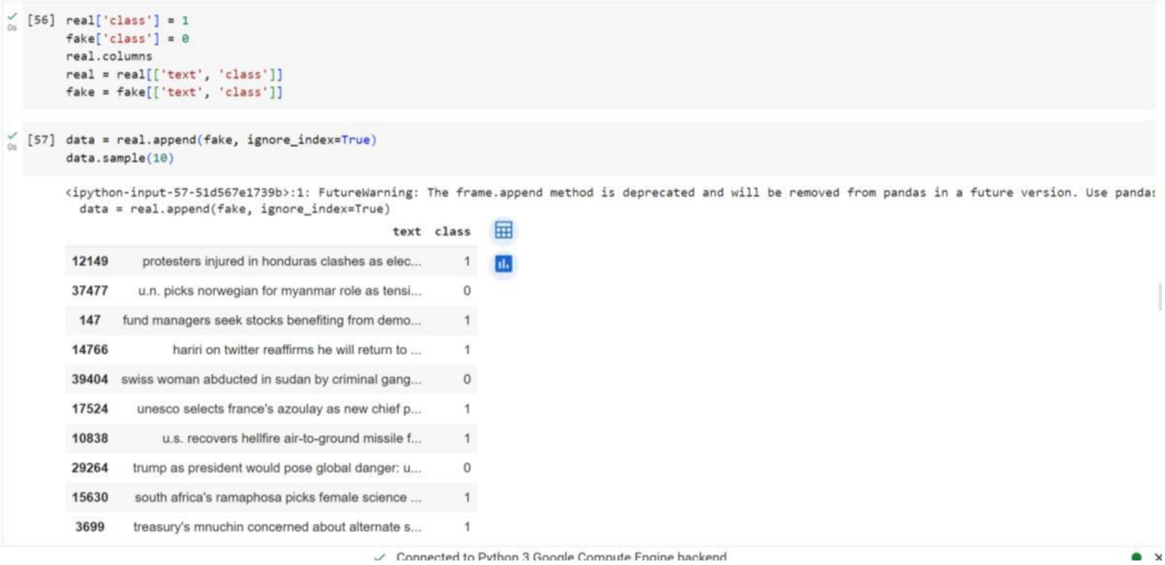
```
In [2]: fake_data = pd.read_csv(r'C:\Users\arulm_x7s4ikd\Downloads\archive\Fake.csv')
```

```
In [3]: fake_data.head(10)
```

Out[3]:

	title	text	subject	date
0	Donald Trump Sends Out Embarrassing New Year'...	Donald Trump just couldn t wish all Americans ...	News	December 31, 2017
1	Drunk Bragging Trump Staffer Started Russian ...	House Intelligence Committee Chairman Devin Nu...	News	December 31, 2017
2	Sheriff David Clarke Becomes An Internet Joke...	On Friday, it was revealed that former Milwauk...	News	December 30, 2017
3	Trump Is So Obsessed He Even Has Obama's Name...	On Christmas day, Donald Trump announced that ...	News	December 29, 2017
4	Pope Francis Just Called Out Donald Trump Dur...	Pope Francis used his annual Christmas Day mes...	News	December 25, 2017
5	Racist Alabama Cops Brutalize Black Boy While...	The number of cases of cops brutalizing and ki...	News	December 25, 2017
6	Fresh Off The Golf Course, Trump Lashes Out A...	Donald Trump spent a good portion of his day a...	News	December 23, 2017
7	Trump Said Some INSANELY Racist Stuff Inside ...	In the wake of yet another court decision that...	News	December 23, 2017
8	Former CIA Director Slams Trump Over UN Bully...	Many people have raised the alarm regarding th...	News	December 22, 2017
9	WATCH: Brand-New Pro-Trump Ad Features So Muc...	Just when you might have thought we d get a br...	News	December 21, 2017

[PREPROCESSING]



```
[56] real['class'] = 1
     fake['class'] = 0
     real.columns
     real = real[['text', 'class']]
     fake = fake[['text', 'class']]
```

```
[57] data = real.append(fake, ignore_index=True)
     data.sample(10)
```

```
<ipython-input-57-51d567e1739b>:1: FutureWarning: The frame.append method is deprecated and will be removed from pandas in a future version. Use panda:
  data = real.append(fake, ignore_index=True)
```

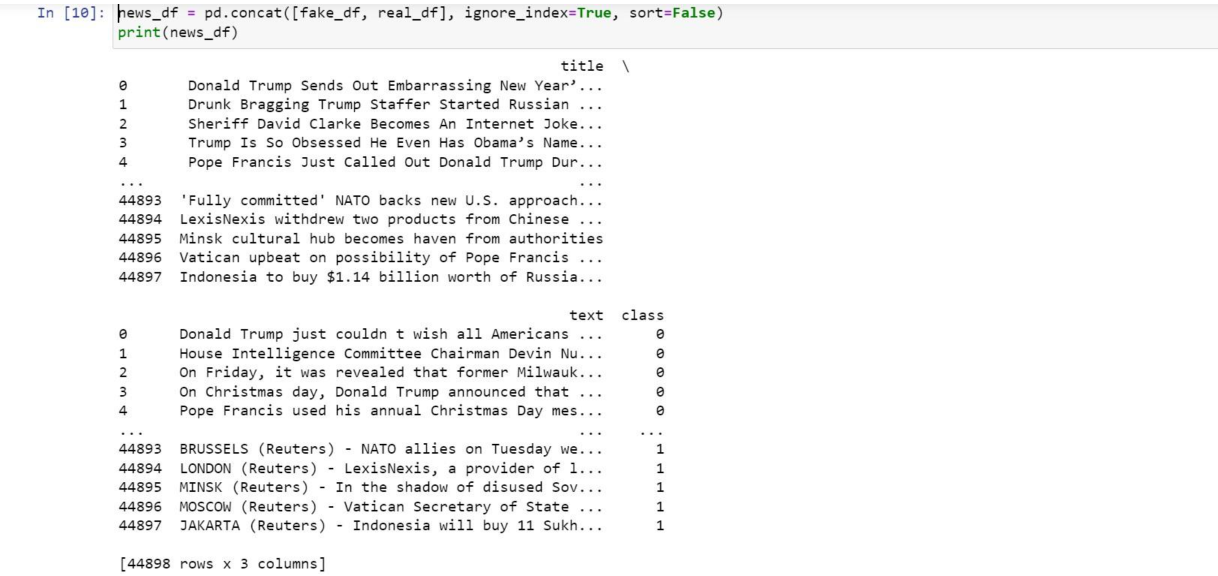
	text	class
12149	protesters injured in honduras clashes as elec...	1
37477	u.n. picks norwegian for myanmar role as tensi...	0
147	fund managers seek stocks benefiting from demo...	1
14766	hariri on twitter reaffirms he will return to ...	1
39404	swiss woman abducted in sudan by criminal gang...	0
17524	unesco selects france's azoulay as new chief p...	1
10838	u.s. recovers hellfire air-to-ground missile f...	1
29264	trump as president would pose global danger: u...	0
15630	south africa's ramaphosa picks female science ...	1
3699	treasury's mnuchin concerned about alternate s...	1

Outlier Detection and Treatment:

Explain the approach to identifying and addressing outliers.

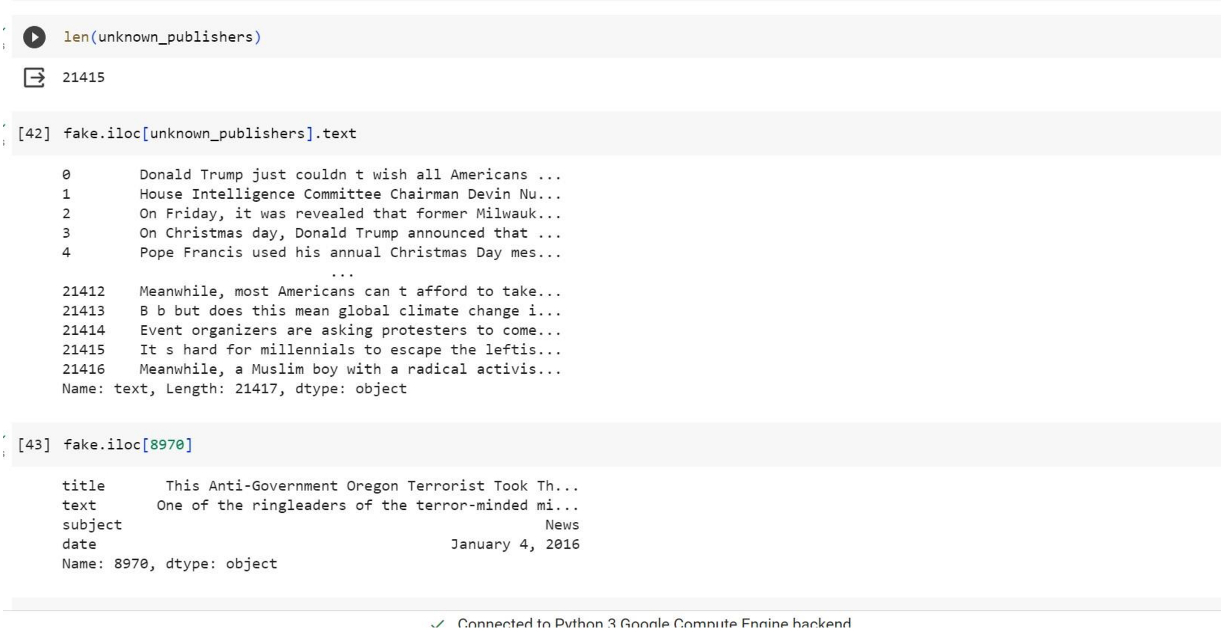
Feature Selection

Unknown publishers is the text ,object Feature Engineering;



```
In [10]: news_df = pd.concat([fake_df, real_df], ignore_index=True, sort=False)
         print(news_df)

                                                      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...
         ...                                                  ...
         44893  'Fully committed' NATO backs new U.S. approach...
         44894  LexisNexis withdrew two products from Chinese ...
         44895  Minsk cultural hub becomes haven from authorities
         44896  Vatican upbeat on possibility of Pope Francis ...
         44897  Indonesia to buy $1.14 billion worth of Russia...

                                                       text  class
         0      Donald Trump just couldn t wish all Americans ...      0
         1      House Intelligence Committee Chairman Devin Nu...      0
         2      On Friday, it was revealed that former Milwauk...      0
         3      On Christmas day, Donald Trump announced that ...      0
         4      Pope Francis used his annual Christmas Day mes...      0
         ...                                                  ...    ...
         44893  BRUSSELS (Reuters) - NATO allies on Tuesday we...      1
         44894  LONDON (Reuters) - LexisNexis, a provider of l...      1
         44895  MINSK (Reuters) - In the shadow of disused Sov...      1
         44896  MOSCOW (Reuters) - Vatican Secretary of State ...      1
         44897  JAKARTA (Reuters) - Indonesia will buy 11 Sukh...      1

         [44898 rows x 3 columns]
```

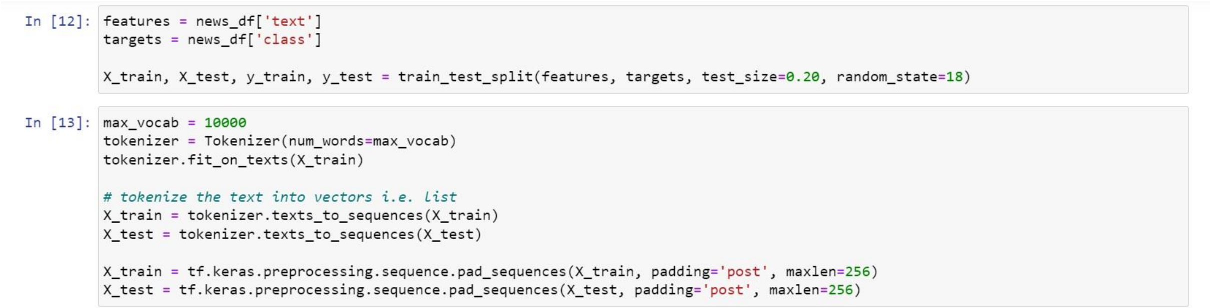
Selecting a future direction for fake news detection using NLP involves considering emerging trends, challenges, and opportunities in the field of natural language processing and misinformation detection.



```
len(unknown_publishers)

21415

[42] fake.iloc[unknown_publishers].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...
                               ...
21412    Meanwhile, most Americans can t afford to take...
21413    B b but does this mean global climate change i...
21414    Event organizers are asking protesters to come...
21415    It s hard for millennials to escape the leftis...
21416    Meanwhile, a Muslim boy with a radical activis...
Name: text, Length: 21417, dtype: object

[43] fake.iloc[8970]

title      This Anti-Government Oregon Terrorist Took Th...
text     One of the ringleaders of the terror-minded mi...
subject                                                News
date                                    January 4, 2016
Name: 8970, dtype: object
```

Machine Learning Algorithm:

Chosen algorithm ; state the Machine learning algorithms used (eg.. LSTM)

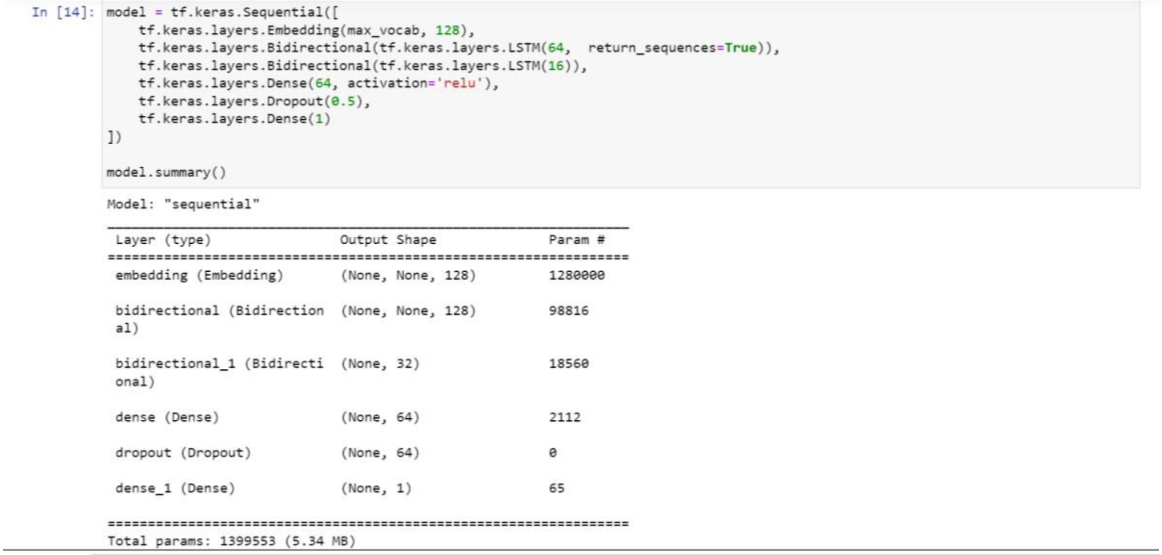


```
In [12]: features = news_df['text']
         targets = news_df['class']

         X_train, X_test, y_train, y_test = train_test_split(features, targets, test_size=0.20, random_state=18)
```

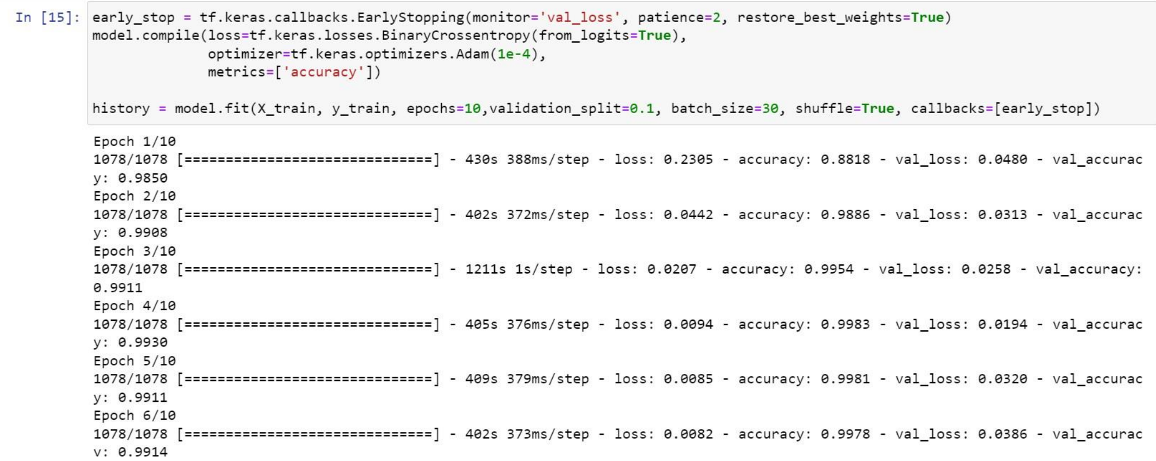
```
In [13]: max_vocab = 10000
         tokenizer = Tokenizer(num_words=max_vocab)
         tokenizer.fit_on_texts(X_train)

         # tokenize the text into vectors i.e. list
         X_train = tokenizer.texts_to_sequences(X_train)
         X_test = tokenizer.texts_to_sequences(X_test)

         X_train = tf.keras.preprocessing.sequence.pad_sequences(X_train, padding='post', maxlen=256)
         X_test = tf.keras.preprocessing.sequence.pad_sequences(X_test, padding='post', maxlen=256)
```

Machine learning algorithms used (eg.. LSTM)

- First load in the data. The preprocessing only consist of normalization and the creation of windows.
- Creation of the LSTM model
- Training the LSTM model



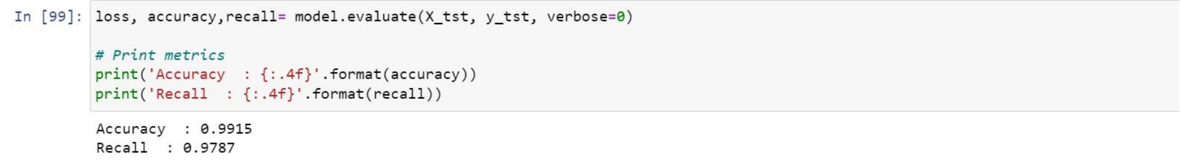
```
In [14]: model = tf.keras.Sequential([
             tf.keras.layers.Embedding(max_vocab, 128),
             tf.keras.layers.Bidirectional(tf.keras.layers.LSTM(64,  return_sequences=True)),
             tf.keras.layers.Bidirectional(tf.keras.layers.LSTM(16)),
             tf.keras.layers.Dense(64, activation='relu'),
             tf.keras.layers.Dropout(0.5),
             tf.keras.layers.Dense(1)
         ])

         model.summary()

         Model: "sequential"
         _________________________________________________________________
          Layer (type)                Output Shape              Param #
         =================================================================
          embedding (Embedding)       (None, None, 128)         1280000

          bidirectional (Bidirection  (None, None, 128)         98816
          al)

          bidirectional_1 (Bidirecti  (None, 32)                18560
          onal)

          dense (Dense)               (None, 64)                2112

          dropout (Dropout)           (None, 64)                0

          dense_1 (Dense)             (None, 1)                 65

         =================================================================
         Total params: 1399553 (5.34 MB)
```



```
In [15]: early_stop = tf.keras.callbacks.EarlyStopping(monitor='val_loss', patience=2, restore_best_weights=True)
         model.compile(loss=tf.keras.losses.BinaryCrossentropy(from_logits=True),
                       optimizer=tf.keras.optimizers.Adam(1e-4),
                       metrics=['accuracy'])

         history = model.fit(X_train, y_train, epochs=10,validation_split=0.1, batch_size=30, shuffle=True, callbacks=[early_stop])

         Epoch 1/10
         1078/1078 [==============================] - 430s 388ms/step - loss: 0.2305 - accuracy: 0.8818 - val_loss: 0.0480 - val_accurac
         y: 0.9850
         Epoch 2/10
         1078/1078 [==============================] - 402s 372ms/step - loss: 0.0442 - accuracy: 0.9886 - val_loss: 0.0313 - val_accurac
         y: 0.9908
         Epoch 3/10
         1078/1078 [==============================] - 1211s 1s/step - loss: 0.0207 - accuracy: 0.9954 - val_loss: 0.0258 - val_accuracy:
         0.9911
         Epoch 4/10
         1078/1078 [==============================] - 405s 376ms/step - loss: 0.0094 - accuracy: 0.9983 - val_loss: 0.0194 - val_accurac
         y: 0.9930
         Epoch 5/10
         1078/1078 [==============================] - 409s 379ms/step - loss: 0.0085 - accuracy: 0.9981 - val_loss: 0.0320 - val_accurac
         y: 0.9911
         Epoch 6/10
         1078/1078 [==============================] - 402s 373ms/step - loss: 0.0082 - accuracy: 0.9978 - val_loss: 0.0386 - val_accurac
         v: 0.9914
```

Evaluation Metrics:

Metrics used : define evaluation metrics such as accuracy , recall , F1- score explain the choice of metrics and how project goals.



```
In [99]: loss, accuracy,recall= model.evaluate(X_tst, y_tst, verbose=0)

         # Print metrics
         print('Accuracy  : {:.4f}'.format(accuracy))
         print('Recall  : {:.4f}'.format(recall))

         Accuracy  : 0.9915
         Recall  : 0.9787
```

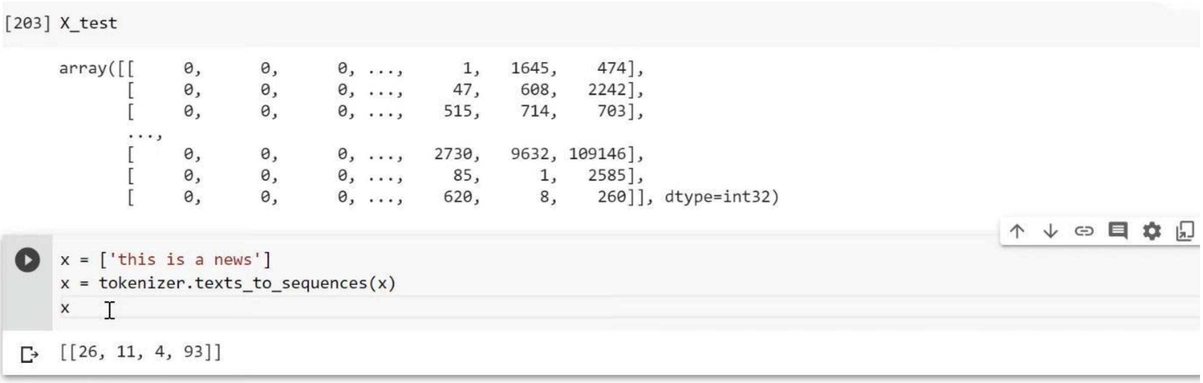
Project Documentation and reporting:

Separate from the report, create documentation for your Fake news detection, including instructions for usage, system architecture, and any code-related documentation.

Remember to use clear and concise python language in your documentation and reporting.

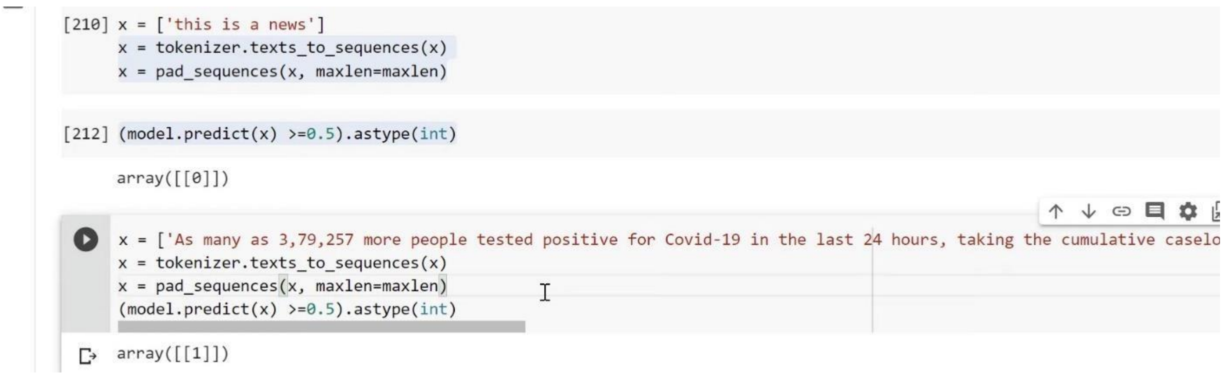
Final outcome:

We showed our final outcome for this project:



```
[203] X_test

    array([[     0,      0,      0, ...,      1,   1645,    474],
           [     0,      0,      0, ...,     47,    608,   2242],
           [     0,      0,      0, ...,    515,    714,    703],
           ...,
           [     0,      0,      0, ...,   2730,   9632, 109146],
           [     0,      0,      0, ...,     85,      1,   2585],
           [     0,      0,      0, ...,    620,      8,    260]], dtype=int32)
```

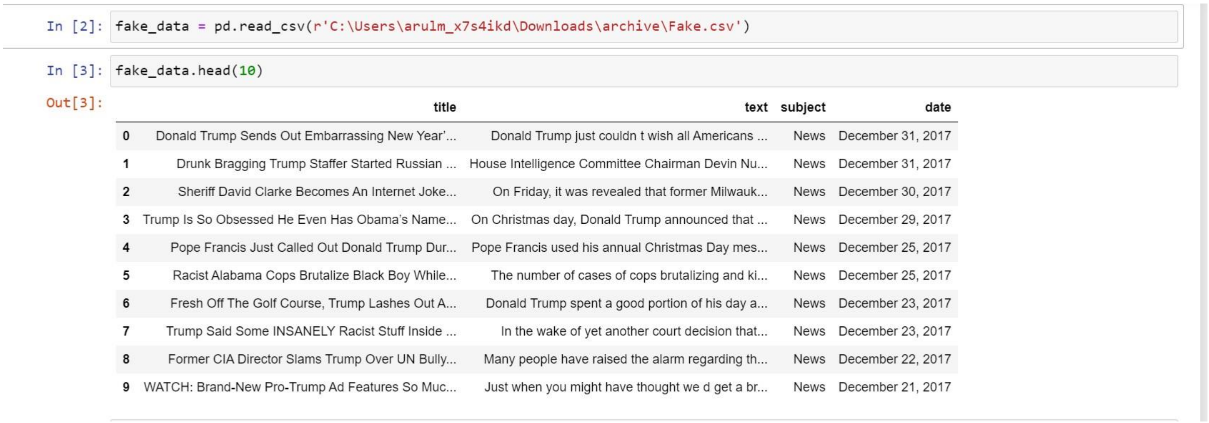
```
x = ['this is a news']
x = tokenizer.texts_to_sequences(x)
x

[[26, 11, 4, 93]]
```



```
[210] x = ['this is a news']
      x = tokenizer.texts_to_sequences(x)
      x = pad_sequences(x, maxlen=maxlen)

[212] (model.predict(x) >=0.5).astype(int)

      array([[0]])
```

```
x = ['As many as 3,79,257 more people tested positive for Covid-19 in the last 24 hours, taking the cumulative caselo
x = tokenizer.texts_to_sequences(x)
x = pad_sequences(x, maxlen=maxlen)
(model.predict(x) >=0.5).astype(int)

array([[1]])
```

FAKE NEWS :-



```
In [2]: fake_data = pd.read_csv(r'C:\Users\arulm_x7s4ikd\Downloads\archive\Fake.csv')
```

```
In [3]: fake_data.head(10)
```

Out[3]:

	title	text	subject	date
0	Donald Trump Sends Out Embarrassing New Year’...	Donald Trump just couldn t wish all Americans ...	News	December 31, 2017
1	Drunk Bragging Trump Staffer Started Russian ...	House Intelligence Committee Chairman Devin Nu...	News	December 31, 2017
2	Sheriff David Clarke Becomes An Internet Joke...	On Friday, it was revealed that former Milwauk...	News	December 30, 2017
3	Trump Is So Obsessed He Even Has Obama’s Name...	On Christmas day, Donald Trump announced that ...	News	December 29, 2017
4	Pope Francis Just Called Out Donald Trump Dur...	Pope Francis used his annual Christmas Day mes...	News	December 25, 2017
5	Racist Alabama Cops Brutalize Black Boy While...	The number of cases of cops brutalizing and ki...	News	December 25, 2017
6	Fresh Off The Golf Course, Trump Lashes Out A...	Donald Trump spent a good portion of his day a...	News	December 23, 2017
7	Trump Said Some INSANELY Racist Stuff Inside ...	In the wake of yet another court decision that...	News	December 23, 2017
8	Former CIA Director Slams Trump Over UN Bully...	Many people have raised the alarm regarding th...	News	December 22, 2017
9	WATCH: Brand-New Pro-Trump Ad Features So Muc...	Just when you might have thought we d get a br...	News	December 21, 2017

Conclusion:

We have classified our news data using three classification models. We have analysed the performance of the models using accuracy and confusion matrix. But this is only a beginning point for the problem.

There are advanced techniques like BERT, GloVe and ELMo which are popularly used in the field of NLP. If you are interested in NLP, you can work forward with these techniques