Project Report

Project Title : News Classification

1. Project Description (in brief):

   We have a dataset where we have a news description and it is labeled as true news or fake news and the classification is to be done on the basis of true and fake news by training the model.
   So various natural language processing steps are performed to extract the needed data for training.
   News classification using logistic regression is a common machine learning task in natural language processing. The goal of news classification is to assign a predefined category or label to a given news article based on its content. Logistic regression is a supervised learning algorithm that is commonly used for binary classification problems but can be extended to multi-class classification problems as well.

   The passive aggressive classifier is a machine learning algorithm that is commonly used for binary classification tasks, such as spam detection or sentiment analysis. It belongs to the family of online learning algorithms and is known for its simplicity, speed, and ability to handle large-scale datasets.

   The passive aggressive classifier works by adjusting the weights of the model iteratively based on the training examples, with the goal of minimizing the classification error. Specifically, the algorithm updates the weights of the model in a "passive aggressive" manner, meaning that it adjusts the weights in proportion to the error made by the model while also trying to minimize the magnitude of the weight update.

2. Type of problem: (supervised\Unsupervised), (Prediction\Classification) Why the problem falls into a particular category?

   It is Supervised learning method as we already know the output of the given data as it is labeled data.
   It is a classification problem as we need to classify new between true and fake and we need to classify between two different categories making it categorical output.

3. Python Libraries used:

NLTK

NLTK (Natural Language Toolkit) is a Python library for working with human language data. It provides a suite of libraries and programs for tasks such as tokenization, parsing, semantic analysis, classification, and machine learning. NLTK was developed at the University of Pennsylvania and is open source software.

Some of the key features of NLTK are:

- Text Processing: NLTK provides various tools for processing text data, such as tokenization, stemming, lemmatization, and stop word removal.

- Corpora: NLTK includes a wide range of corpora, or large collections of text, that can be used for language modeling and analysis. These corpora include the Brown Corpus, the Gutenberg Corpus, and the WordNet Corpus.

- Language Models: NLTK includes tools for building and evaluating language models, such as n-gram models, Markov models, and Hidden Markov Models.

- Machine Learning: NLTK includes various machine learning algorithms for tasks such as classification, clustering, and regression. These algorithms include decision trees, Naive Bayes, and Maximum Entropy.

- Visualization: NLTK includes tools for visualizing text data, such as frequency distributions, collocations, and concordances.

Sklearn

Scikit-learn, also known as sklearn, is a popular Python library for machine learning. It is built on top of NumPy, SciPy, and matplotlib, and provides a simple and efficient set of tools for data mining and analysis.

Pandas

Pandas is an open-source Python library for data manipulation and analysis. It provides a set of powerful tools for data preprocessing, cleaning, analysis, and visualization.

4. Data set details

There are two datasets named “True.csv” and “Fake.csv”. Both have the columns like title, text, subject, date published. Based on the text we need to classify into true or fake and all the data present in true csv are true news and for false news the data is in false csv and we need to append the labels in the datasets and combine both dataset.

True :
Rows,cols = (21417, 4)
Features : 'title', 'text', 'subject', 'date'
Title has title or heading of the news, text has the actual news, subject is the topic of the news and date is when the news was published.
Out of these only text is a feature which will be used during the training as rest all can be avoided as date is irrelevant and subject and title are too generic and not useful.

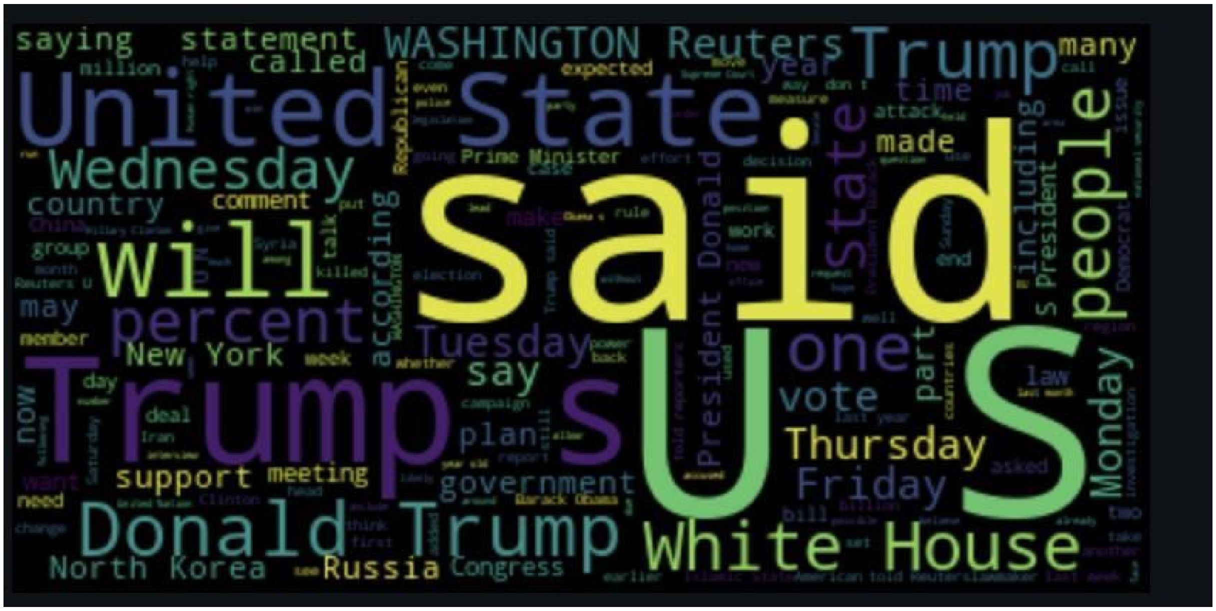
Fake :
Rows,cols = (23481, 4)
Features : 'title', 'text', 'subject', 'date'
Title has title or heading of the news, text has the actual news, subject is the topic of the news and date is when the news was published.
Out of these only text is a feature which will be used during the training as rest all can be avoided as date is irrelevant and subject and title are too generic and not useful.

Dataset source :
https://www.kaggle.com/datasets/clmentbisaillon/fake-and-real-news-dataset Dataset is around 5-6 years old.

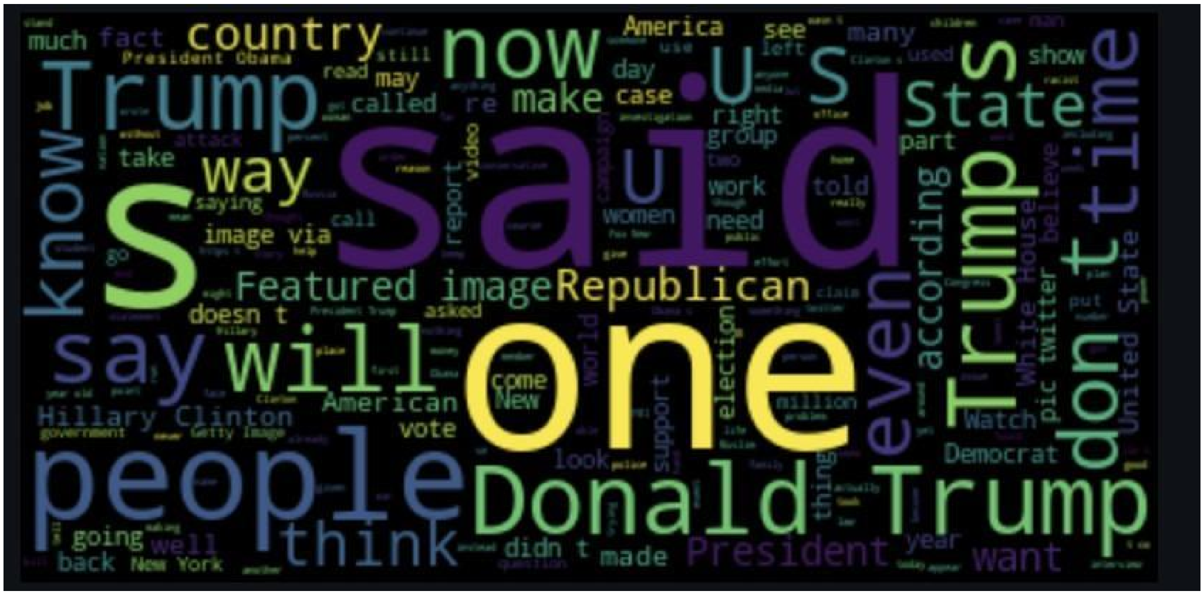
5. Data set visualization and inference

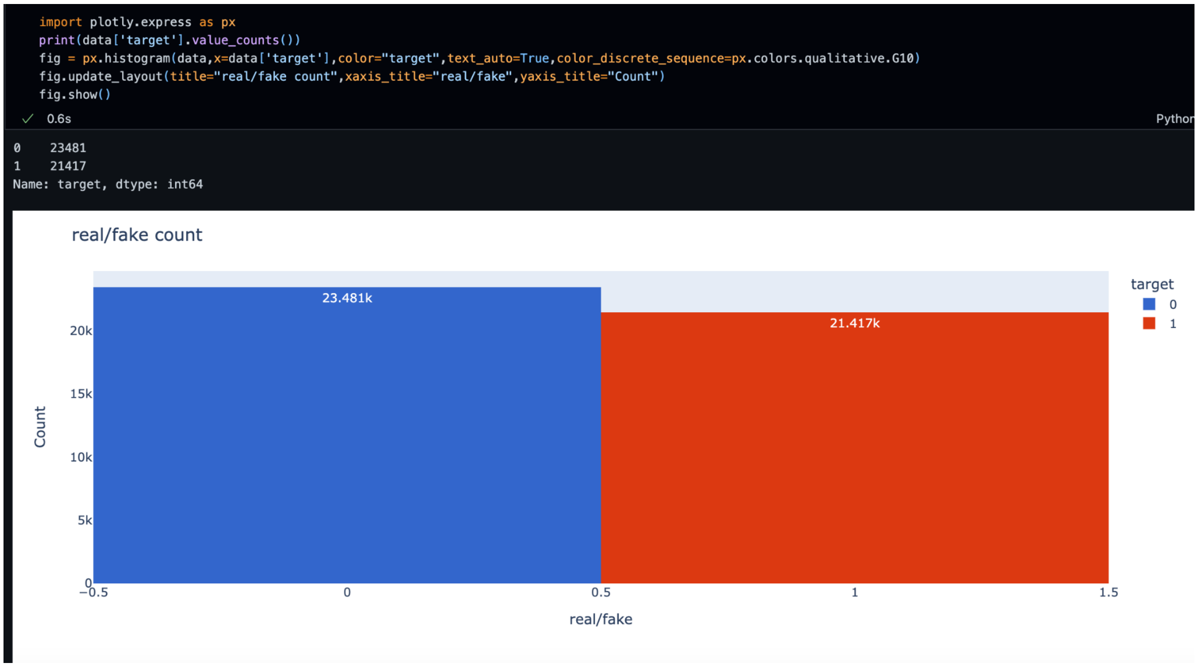
Visualization can be done using the wordcloud by getting to know the frequency and importance of words.

True :



Fake:





```
import plotly.express as px
print(data['target'].value_counts())
fig = px.histogram(data,x=data['target'],color="target",text_auto=True,color_discrete_sequence=px.colors.qualitative.G10)
fig.update_layout(title="real/fake count",xaxis_title="real/fake",yaxis_title="Count")
fig.show()
```

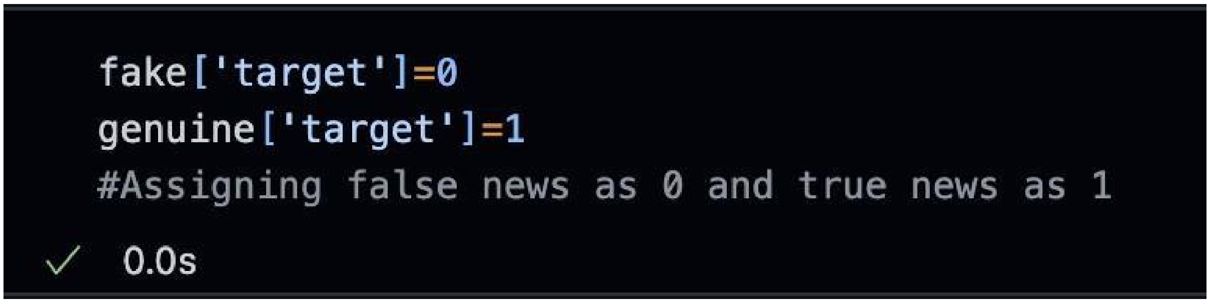
```
0    23481
1    21417
Name: target, dtype: int64
```

6. Data Cleaning steps

The data was already cleaned.

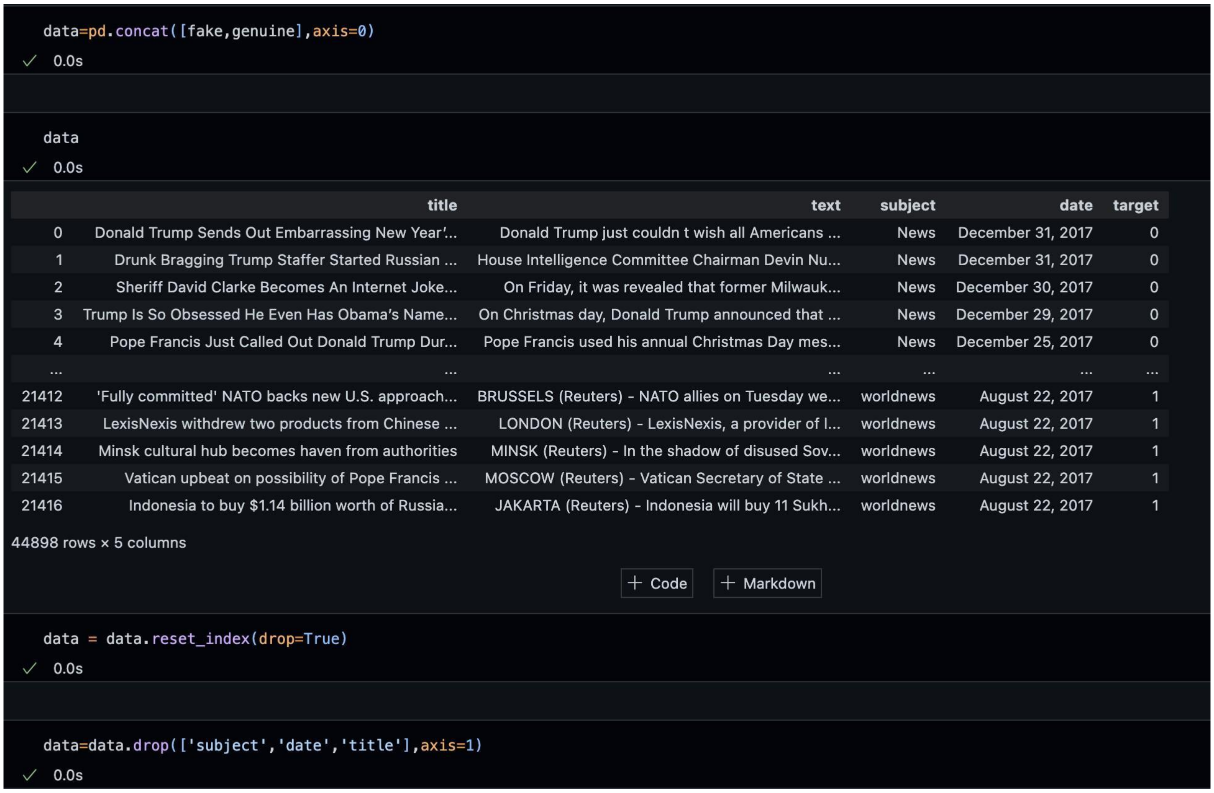
7. Data Preprocessing steps

Adding output columns in both the files.



```
fake['target']=0
genuine['target']=1
#Assigning false news as 0 and true news as 1
```

✓ 0.0s



```
data=pd.concat([fake,genuine],axis=0)
```

✓ 0.0s

```
data
```

✓ 0.0s

	title	text	subject	date	target
0	Donald Trump Sends Out Embarrassing New Year'...	Donald Trump just couldn t wish all Americans ...	News	December 31, 2017	0
1	Drunk Bragging Trump Staffer Started Russian ...	House Intelligence Committee Chairman Devin Nu...	News	December 31, 2017	0
2	Sheriff David Clarke Becomes An Internet Joke...	On Friday, it was revealed that former Milwauk...	News	December 30, 2017	0
3	Trump Is So Obsessed He Even Has Obama's Name...	On Christmas day, Donald Trump announced that ...	News	December 29, 2017	0
4	Pope Francis Just Called Out Donald Trump Dur...	Pope Francis used his annual Christmas Day mes...	News	December 25, 2017	0
...	...	...	...	...	...
21412	'Fully committed' NATO backs new U.S. approach...	BRUSSELS (Reuters) - NATO allies on Tuesday we...	worldnews	August 22, 2017	1
21413	LexisNexis withdrew two products from Chinese ...	LONDON (Reuters) - LexisNexis, a provider of l...	worldnews	August 22, 2017	1
21414	Minsk cultural hub becomes haven from authorities	MINSK (Reuters) - In the shadow of disused Sov...	worldnews	August 22, 2017	1
21415	Vatican upbeat on possibility of Pope Francis ...	MOSCOW (Reuters) - Vatican Secretary of State ...	worldnews	August 22, 2017	1
21416	Indonesia to buy $1.14 billion worth of Russia...	JAKARTA (Reuters) - Indonesia will buy 11 Sukh...	worldnews	August 22, 2017	1

44898 rows × 5 columns

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

✓ 0.0s

```
data=data.drop(['subject','date','title'],axis=1)
```

✓ 0.0s

Merging both the files and randomly sorting the rows and dropping unnecessary columns.

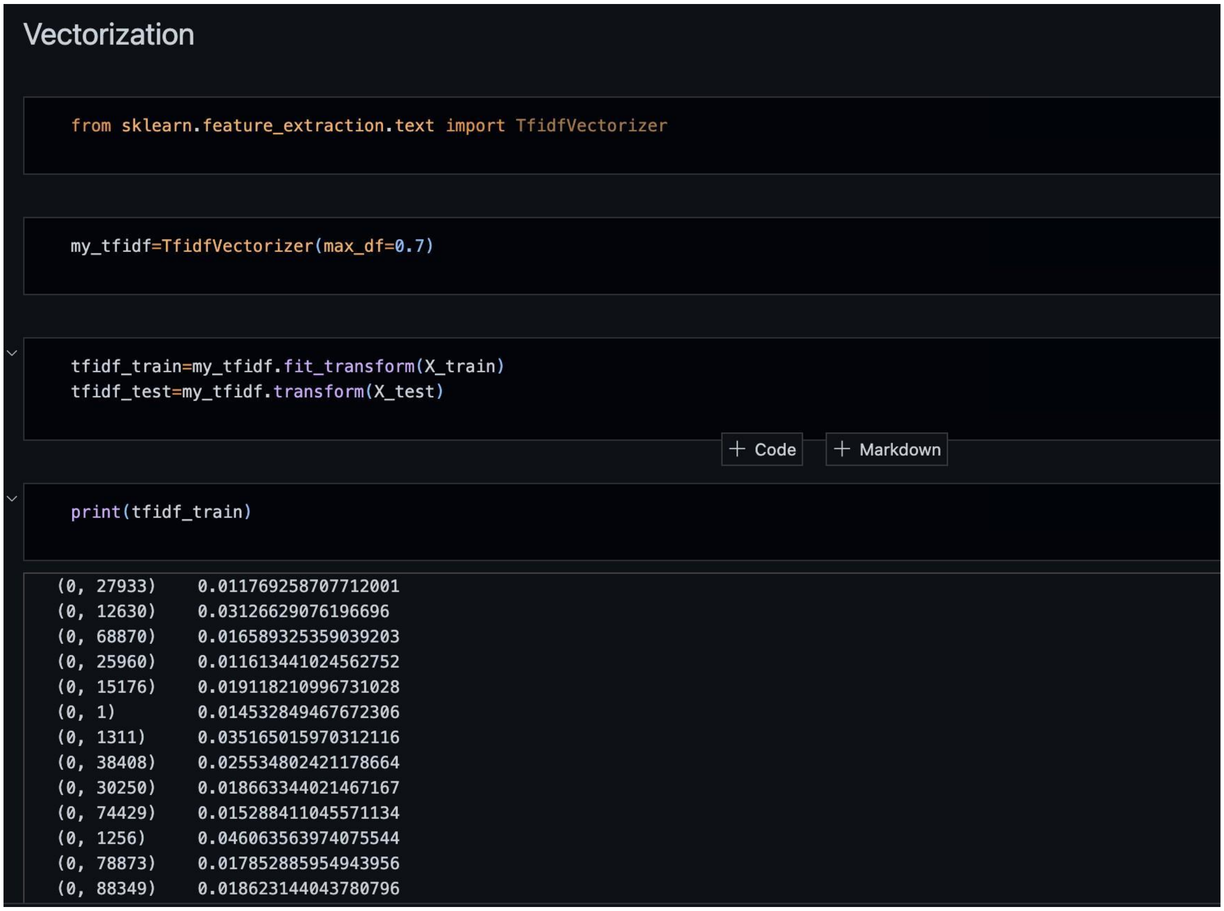
8. Feature scaling\Normalization (if applied) – Which technique is implemented why?

There was no feature scaling but there was application of vectorization.
In Natural Language Processing (NLP), vectorization is the process of transforming text into numerical vectors or arrays of numbers that can be understood and processed by machine learning models. This is necessary because machine learning algorithms typically operate on numerical data, while textual data is represented as strings of characters.

TF-IDF (Term Frequency-Inverse Document Frequency) is a popular technique for vectorizing text data in Natural Language Processing (NLP). It is used to convert text documents into numerical feature vectors that can be used in machine learning models for various tasks such as document classification, information retrieval, and text mining.

TF-IDF works by assigning a weight to each term (word or phrase) in a document, based on how frequently it appears in the document (term frequency) and how common it is across all documents in a corpus (inverse document frequency).

The term frequency (TF) of a term t in a document d is calculated as the number of times t appears in d divided by the total number of terms in d. This value is then multiplied by the inverse document frequency (IDF) of the term t, which is calculated as the logarithm of the total number of documents in the corpus divided by the number of documents that contain t. The formula for TF-IDF is:



Vectorization

```
from sklearn.feature_extraction.text import TfidfVectorizer
```

```
my_tfidf=TfidfVectorizer(max_df=0.7)
```

```
tfidf_train=my_tfidf.fit_transform(X_train)
tfidf_test=my_tfidf.transform(X_test)
```

```
print(tfidf_train)
```

```
  (0, 27933)	0.011769258707712001
  (0, 12630)	0.03126629076196696
  (0, 68870)	0.016589325359039203
  (0, 25960)	0.011613441024562752
  (0, 15176)	0.019118210996731028
  (0, 1)	0.014532849467672306
  (0, 1311)	0.035165015970312116
  (0, 38408)	0.025534802421178664
  (0, 30250)	0.018663344021467167
  (0, 74429)	0.015288411045571134
  (0, 1256)	0.046063563974075544
  (0, 78873)	0.017852885954943956
  (0, 88349)	0.018623144043780796
```

TF-IDF(t, d) = TF(t, d) * IDF(t)

The TF-IDF score gives a high weight to terms that are frequent in a particular document, but rare in the corpus as a whole. This is because such terms are considered to be more important for characterizing the content of the document. On the other hand, terms that are common across all documents in the corpus are given a low weight, as they are less useful for distinguishing one document from another.

9. Model building- Describe in detail what all models were build, train and test split size. If required put sample code for model building

News classification using logistic regression is a common machine learning task in natural language processing. The goal of news classification is to assign a predefined category or label to a given news article based on its content. Logistic regression is a supervised learning algorithm that is commonly used for binary classification problems but can be extended to multi-class classification problems as well.Here are the basic steps for building a news classification system using logistic regression:

- Data preparation: The first step is to gather and preprocess the data. This involves collecting news articles from various sources, cleaning and preprocessing the data, and splitting it into training and testing sets.

- Feature extraction: The next step is to convert the text data into a numerical representation that can be used as input to the logistic regression model. Commonly used techniques include bag-of-words, TF-IDF, and word embeddings.

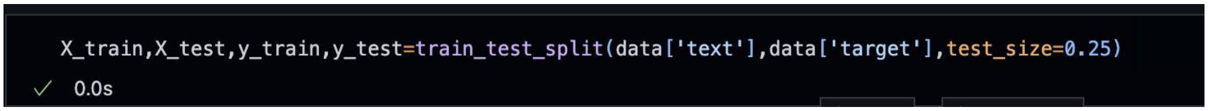
- Model training: Once the data is preprocessed and features are extracted, the logistic regression model can be trained on the training set. The goal of training is to learn the optimal parameters that can classify the news articles into their respective categories.

- Model evaluation: After the model is trained, it needs to be evaluated on the testing set to measure its accuracy and performance. Various metrics can be used to evaluate the model, such as precision, recall, F1 score, and accuracy.

- Model deployment: Once the model is evaluated and validated, it can be deployed to classify new news articles into their respective categories.

Overall, logistic regression is a simple yet effective algorithm for news classification tasks. However, its performance may be limited by the quality and quantity of the training data, the choice of features, and the hyperparameters of the model. Therefore, it is important to experiment with different techniques and parameters to achieve the best possible performance.

Train and test split size :

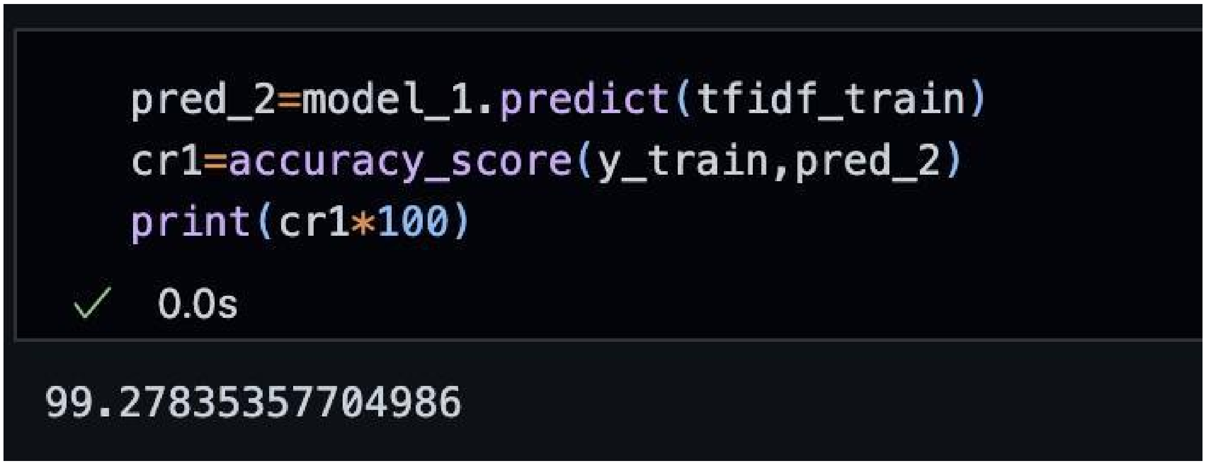


```
X_train,X_test,y_train,y_test=train_test_split(data['text'],data['target'],test_size=0.25)
```

✓ 0.0s

10.Model evaluation – Comparative description regarding training and testing accuracy depending

Training accuracy :



```
pred_2=model_1.predict(tfidf_train)
cr1=accuracy_score(y_train,pred_2)
print(cr1*100)
```

```
✓  0.0s
```

```
99.27835357704986
```

11. Conclusion :

In conclusion, to classify news articles into true or fake using Python, you need to collect a labeled dataset of news articles, preprocess the data, convert it into numerical features, split the data into training and testing sets, train a machine learning model, and evaluate its performance using metrics like accuracy, precision, recall, and F1 score. Creating a high-quality dataset is essential for building an accurate model, and it's crucial to continually monitor and update the model to account for changes in the types of fake news articles being produced. By following these steps, you can build an effective news classification system that can help identify false information and promote accurate reporting.