Fake News Detection

Phase-2 Document Submission

Project: Fake News Detection using LSTM (Long short-Term Memory)

Team Members:

Logeshwaran M	- 110521106012 [Team leader]
Gunasegkaran S V	-110521106311
Karthi M	-110521106312
Shalini R	-110521106024

Overview:

Objectives:

The primary objective of this project is to develop an advanced AI-based solution for the automated detection of fake news articles from genuine ones. Fake news is a pervasive issue in today's information landscape, and this project seeks to contribute to the mitigation of its impact on society.

Methodology:

The project leverages the power of deep learning, specifically LSTM neural networks, for natural language processing and text classification. LSTM is well-suited for sequential data like text, making it an ideal choice for this task. The model is trained on a Label dataset of news articles, learning to distinguish between trustworthy and misleading content.

Key Components:

- **Data Collection:** A diverse and comprehensive dataset of news articles, containing both real and fake news, is gathered for training and testing the model.
- **Data Preprocessing:** The collected data undergoes preprocessing, including text cleaning, tokenization, and numerical vectorization to make it suitable for LSTM input.
- **LSTM Model:** The core of the project is the LSTM neural network architecture, which learns to capture sequential patterns and semantic context in the text.
- **Training and Evaluation:** The model is trained on the preprocessed data, and its performance is evaluated using Accuracy score.
- **Results:** The project provides insights into the model's effectiveness in detecting fake news, with a focus on achieving high accuracy and reducing false positives.

Challenges:

- Handling class imbalance in the dataset, as fake news articles are often outnumbered by genuine ones.
- Tuning hyperparameters and optimising the LSTM architecture for the best performance
- Addressing issues related to news articles with subtle misinformation, which might be challenging to detect.

Benefits:

- Fake news detection using LSTM offers a scalable and efficient solution to the ongoing problem of misinformation.
- It can be deployed in real-time to analyse news articles and social media content for authenticity, aiding journalists, fact-checkers, and platforms in content verification.

- The project contributes to promoting media literacy and responsible information consumption in society.
- In business and finance, fake news detection can prevent stock market manipulation, financial fraud, and other economic risks, potentially saving significant financial resources.

Datasets: https://www.kaggle.com/datasets/clmentbisaillon/fake-and-real-newsdataset

Steps:

Step-1: Data Collections and preparations:

- **Data Collection:** Describe the process of collecting a Labeled dataset of news articles, including fake and real labels.
- **Data Cleaning:** Explain how the collected data was cleaned, including removing irrelevant information, special characters, and stop words.
- **Tokenization:** Describe the tokenization process, converting text into individual words or subword units.
- **Data Splitting:** Explain how the dataset was split into training, validation, and test sets for model evaluation.

Step-2: Word Embeddings:

- **Word Embeddings:** Introduce the concept of word embeddings and explain the use of pre-trained embeddings like Word2Vec, GloVe, or FastText.
- **Embedding Layer:** Detail how words are mapped to their corresponding embedding vectors and used in the model.

Step-3: Sequence Padding:

- **Padding:** Explain the need for sequence padding to ensure that all input sequences have the same length.
- **Pad Sequences:** Describe how shorter sequences are padded with zeros, and longer sequences are truncated to a fixed length.

Step-4: Model Architecture:

- **LSTM Architecture:** Present the LSTM-based neural network architecture, including layers and units.
- **Embedding Layer:** Explain the embedding layer's role in understanding word semantics.
- **LSTM Layers:** Detail the LSTM layers that capture sequential patterns.
- **Final Dense Layer:** Describe the final dense layer for binary classification.

Step-5: Model Compilation:

- **Compile the Model:** Specify the model's optimizer (e.g., Adam), loss function (e.g., binary cross-entropy), and evaluation metric (e.g., accuracy).

Step-6: Model Training:

- **Training Process:** Explain how the model is trained using the training dataset.
- **Validation:** Describe the use of a validation set to monitor the model's performance during training.

Step-7: Model Evaluation:

- **Evaluation:** Discuss the evaluation metrics used to assess the model's performance, such as accuracy.

Step-8: Threshold Selection:

- **Threshold:** Discuss the selection of a threshold value for binary classification based on model output.
- **Default Value:**0.5 is a default threshold value.

Step-9:Testing:

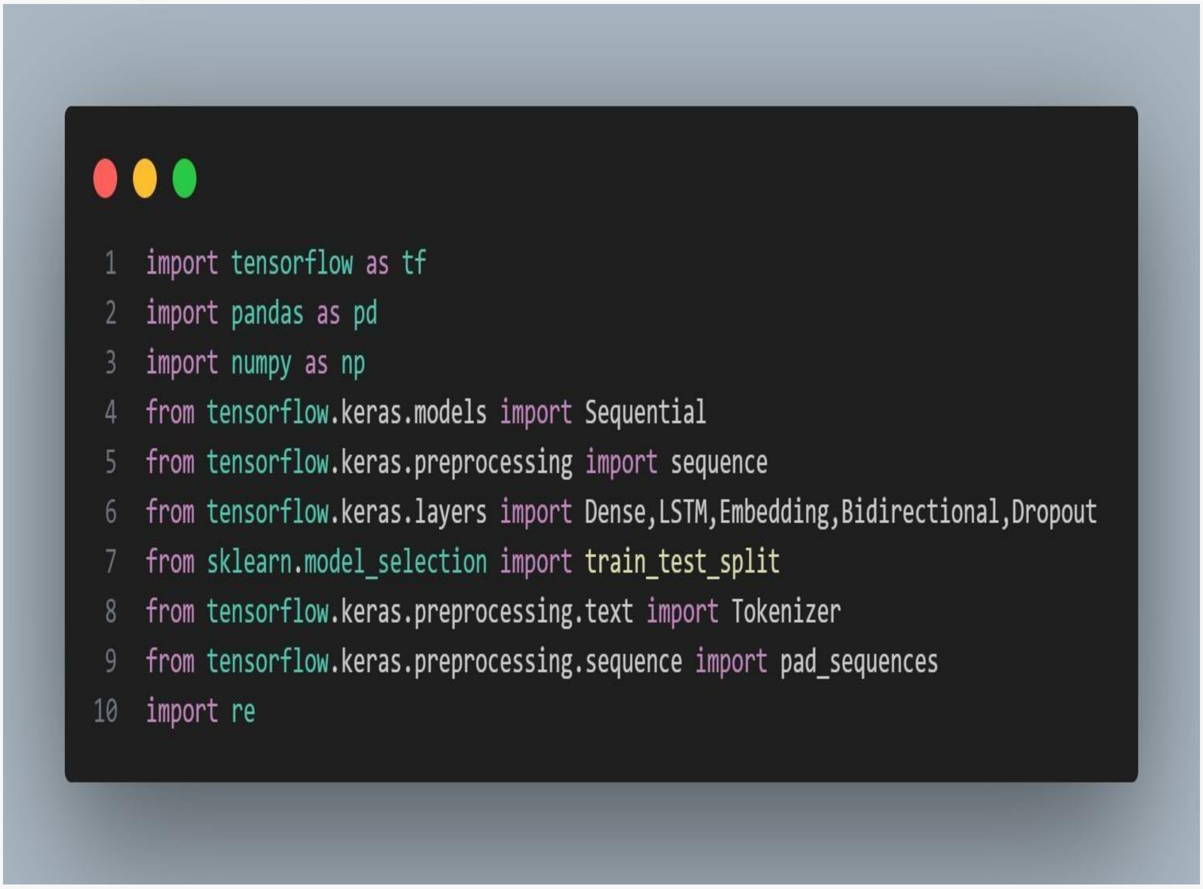
- **Input:** Getting the news as input from the users.
- **Tokenize:** Tokenize the text (split into tokens) and convert the text into numerical values.
- **padding:**Padding is often used when working with sequences of varying lengths, such as text data, to ensure that all sequences have the same length.

Step-10: Prediction:

- **Prediction:**After selection the threshold value based on our Datasets and Compare the predicted value with the threshold value and display the output.whether the Article contains Fake News or Genuine News.

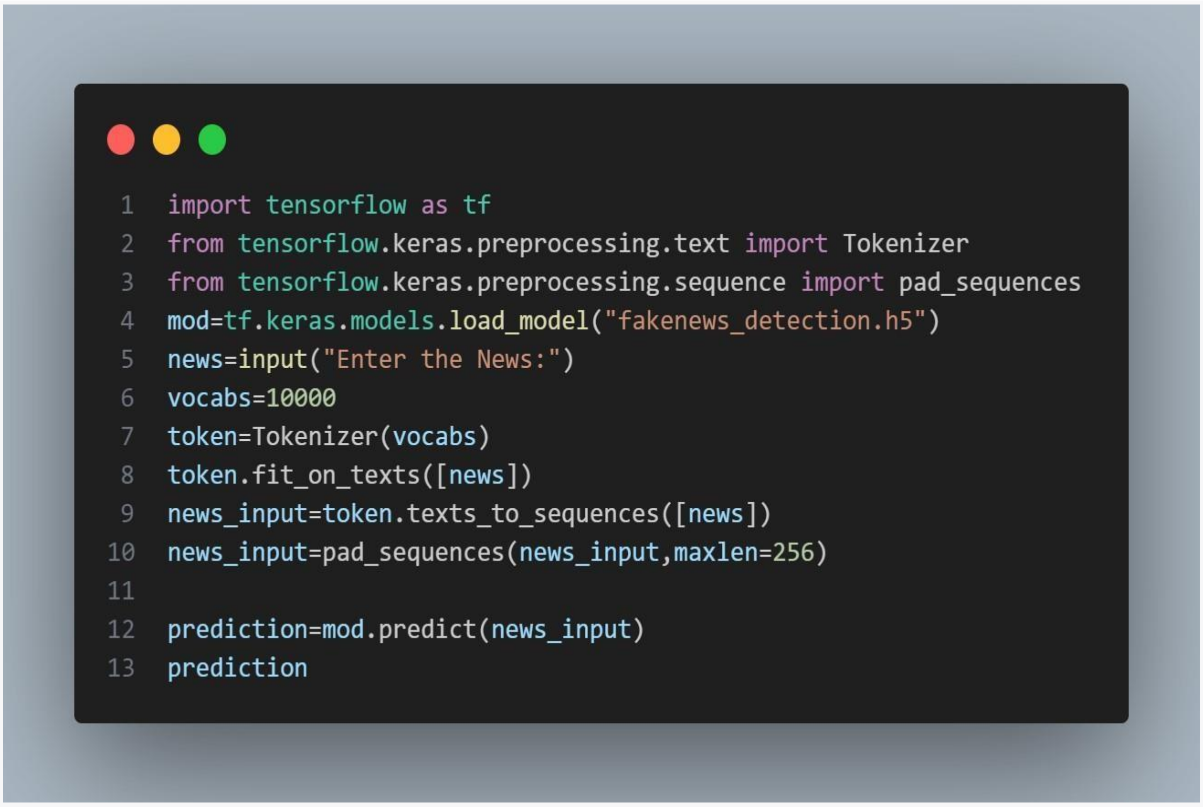
Program:

Importing Necessary Modules:



```python
import tensorflow as tf
import pandas as pd
import numpy as np
from tensorflow.keras.models import Sequential
from tensorflow.keras.preprocessing import sequence
from tensorflow.keras.layers import Dense,LSTM,Embedding,Bidirectional,Dropout
from sklearn.model_selection import train_test_split
from tensorflow.keras.preprocessing.text import Tokenizer
from tensorflow.keras.preprocessing.sequence import pad_sequences
import re
```

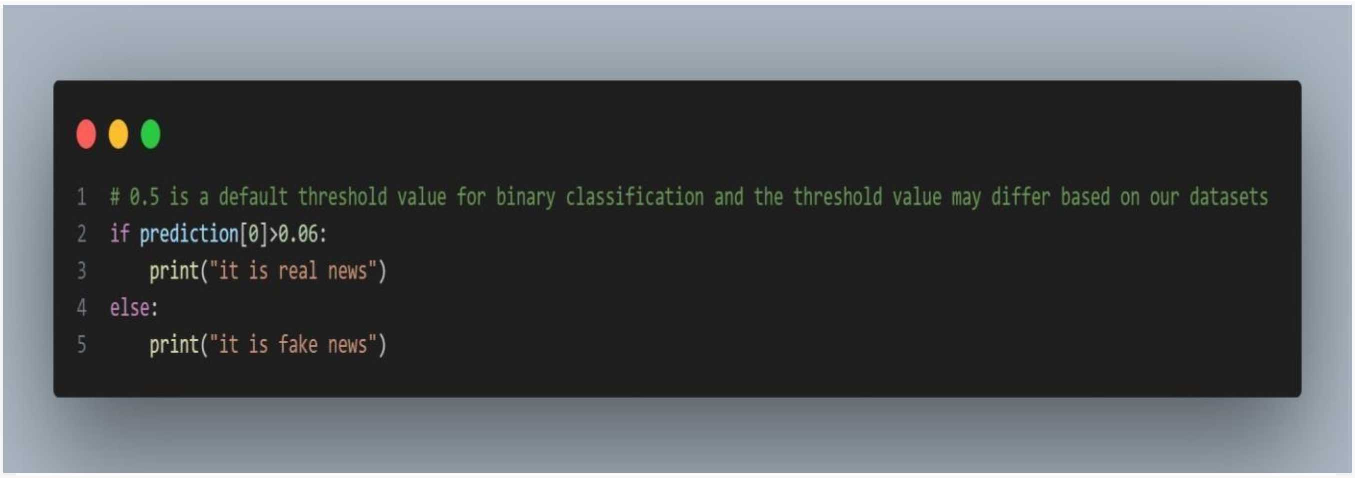
Testing the Trained Model:



```python
import tensorflow as tf
from tensorflow.keras.preprocessing.text import Tokenizer
from tensorflow.keras.preprocessing.sequence import pad_sequences
mod=tf.keras.models.load_model("fakenews_detection.h5")
news=input("Enter the News:")
vocabs=10000
token=Tokenizer(vocabs)
token.fit_on_texts([news])
news_input=token.texts_to_sequences([news])
news_input=pad_sequences(news_input,maxlen=256)

prediction=mod.predict(news_input)
prediction
```

Prediction:



```python
# 0.5 is a default threshold value for binary classification and the threshold value may differ based on our datasets
if prediction[0]>0.06:
    print("it is real news")
else:
    print("it is fake news")
```

Accuracy Details:

Accuracy:0.999375

Loss:0.003811

Conclusion:

In conclusion, the use of LSTM (Long Short-Term Memory) neural networks for fake news detection offers a powerful approach to identifying deceptive content in news articles. By preprocessing and tokenizing textual data, leveraging word embeddings, and constructing an LSTM-based model, we can effectively capture sequential patterns in the text. With appropriate training, evaluation, and threshold selection, LSTM models can provide valuable insights into classifying news articles as either fake or real.