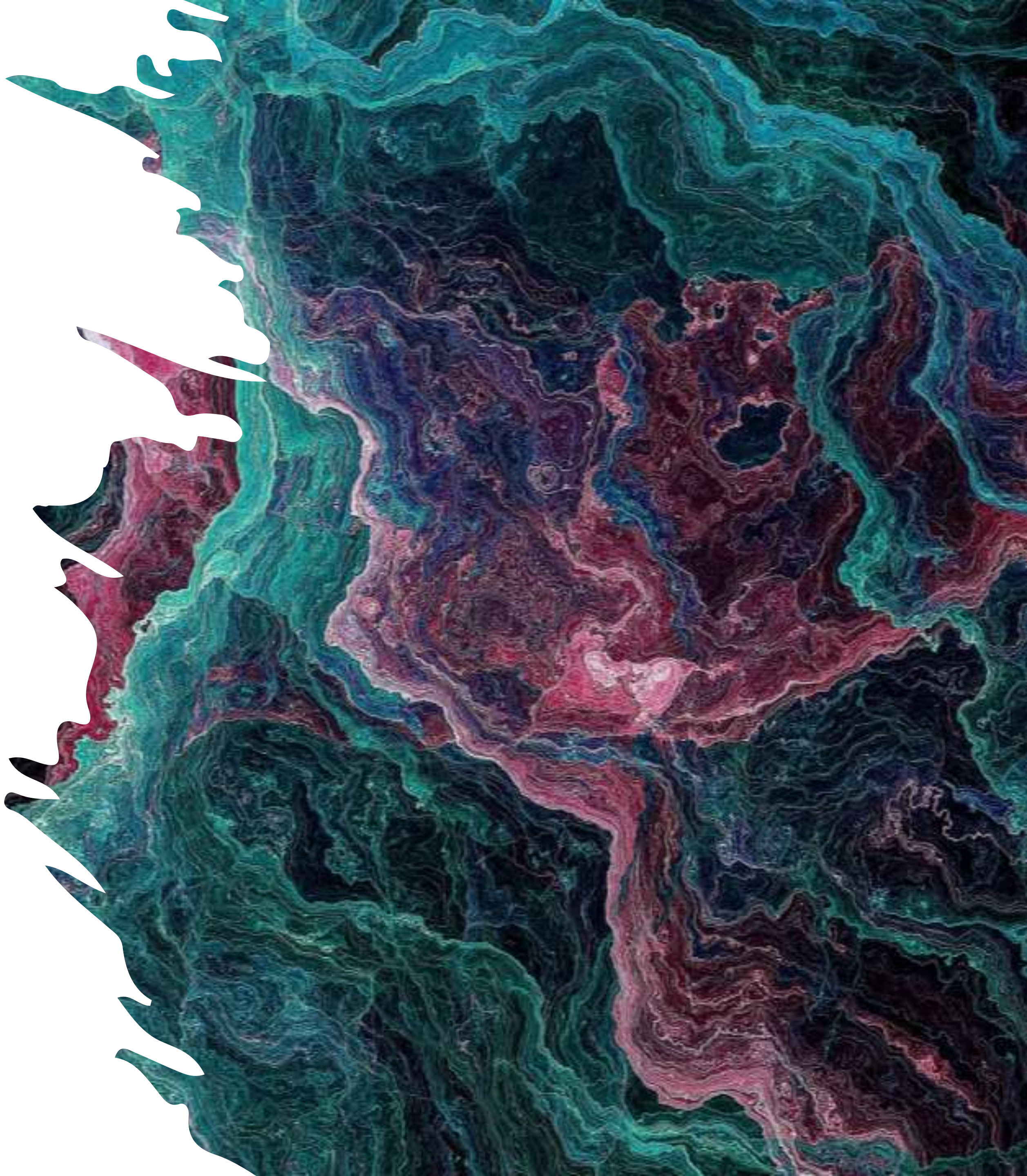
Fake News Detection using deep learning models

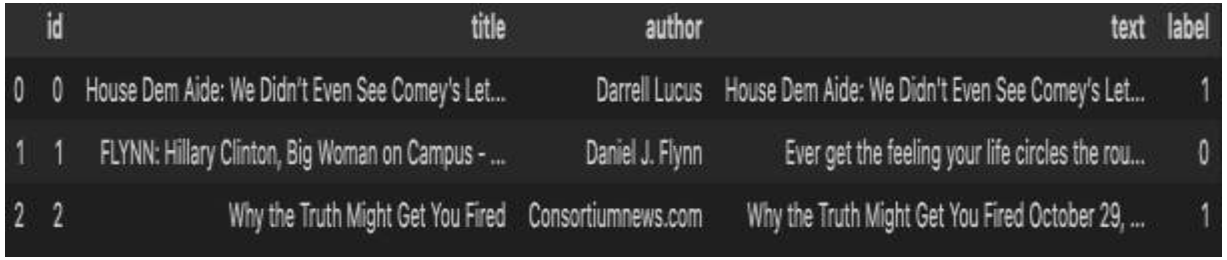
By- Mathu and Vijay Shrinivas Senthilnathan



About Our Dataset

Dataset Description:

Our dataset comprises online news articles covering various topics. These articles are used to evaluate different architectural setups for fake news detection, enabling us to assess their accuracy scores and effectiveness.

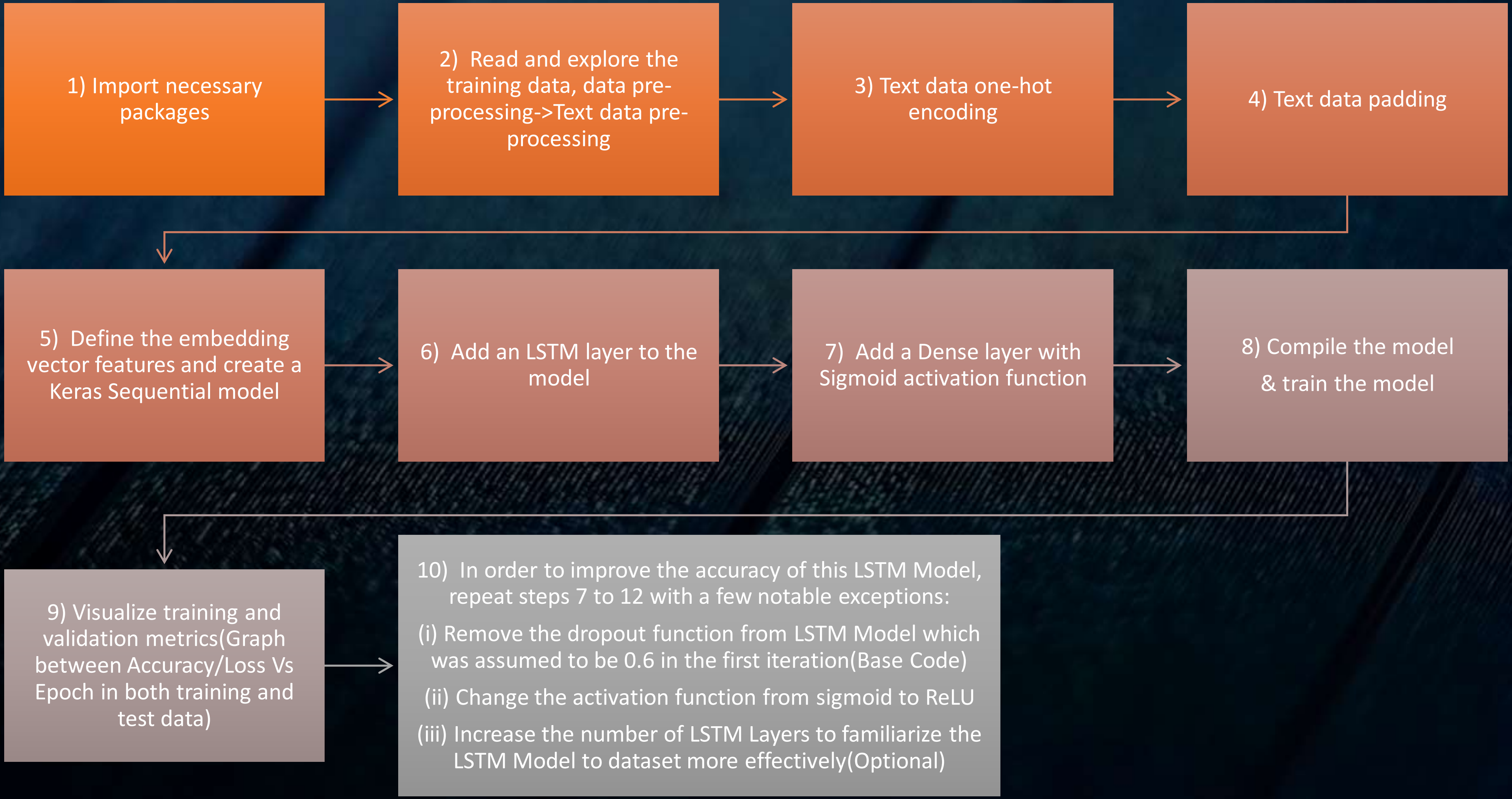


	id	title	author	text	label
0	0	House Dem Aide: We Didn't Even See Comey's Let...	Darrell Lucus	House Dem Aide: We Didn't Even See Comey's Let...	1
1	1	FLYNN: Hillary Clinton, Big Woman on Campus - ...	Daniel J. Flynn	Ever get the feeling your life circles the rou...	0
2	2	Why the Truth Might Get You Fired	Consortiumnews.com	Why the Truth Might Get You Fired October 29, ...	1

Dataset Link: https://github.com/TrilokiDA/Fake-News-Classifier-LSTM/blob/master/Fake%20News%20Classifier-LSTM.ipynb

(the file named train.csv in the github repository is the dataset used)

Basic summarization of code as an algorithm:



1) Import necessary packages

2) Read and explore the training data, data pre-processing->Text data pre-processing

3) Text data one-hot encoding

4) Text data padding

5) Define the embedding vector features and create a Keras Sequential model

6) Add an LSTM layer to the model

7) Add a Dense layer with Sigmoid activation function

8) Compile the model & train the model

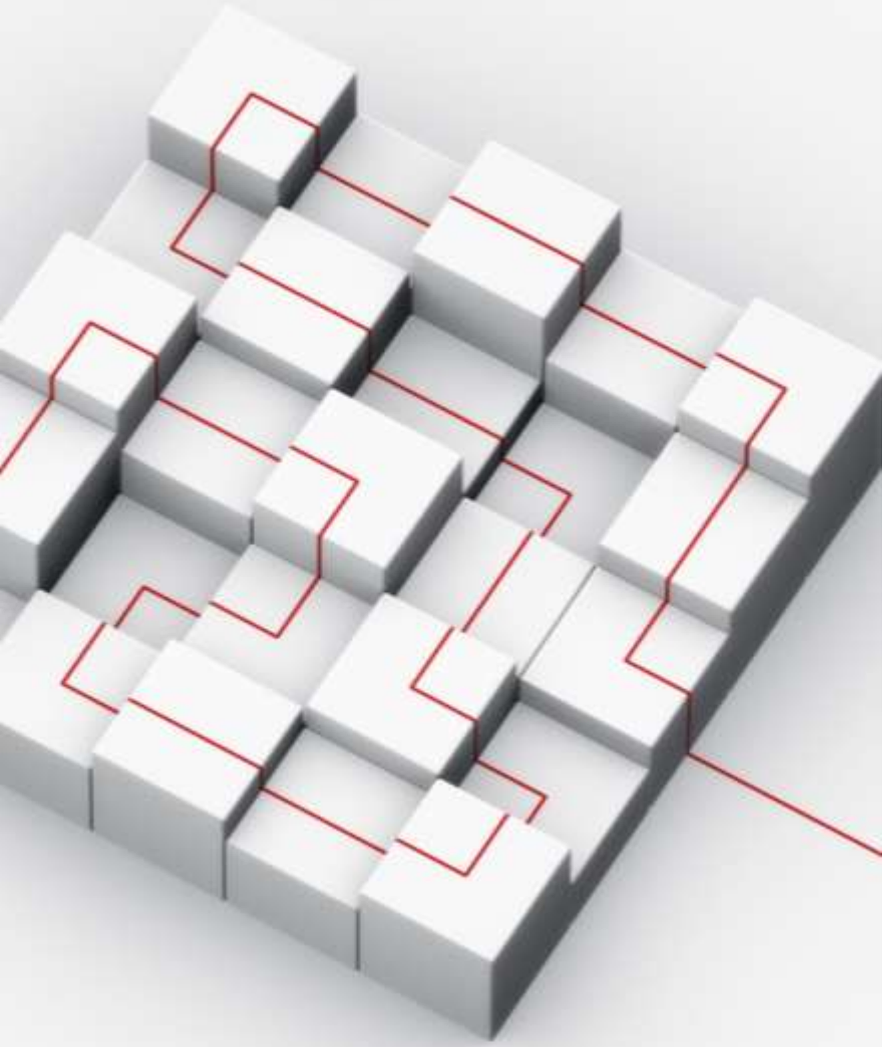
9) Visualize training and validation metrics(Graph between Accuracy/Loss Vs Epoch in both training and test data)

10) In order to improve the accuracy of this LSTM Model, repeat steps 7 to 12 with a few notable exceptions:

(i) Remove the dropout function from LSTM Model which was assumed to be 0.6 in the first iteration(Base Code)

(ii) Change the activation function from sigmoid to ReLU

(iii) Increase the number of LSTM Layers to familiarize the LSTM Model to dataset more effectively(Optional)



LSTM and Embedding

- **LSTM (Long Short-Term Memory):** LSTM is a type of recurrent neural network (RNN) architecture commonly used in deep learning models for tasks like fake news detection. LSTMs are designed to capture and model sequential data, making them well-suited for tasks involving natural language processing (NLP) where the order of words in a text is important.
  - LSTMs address the vanishing gradient problem of traditional RNNs, allowing them to capture long-range dependencies in text data. This is crucial in fake news detection because understanding the context and relationships between words and phrases is vital to differentiate between credible and non-credible content. LSTMs process input sequences step by step, updating their internal state at each time step and retaining information about past inputs.
  - In fake news detection, LSTMs can be used to analyse textual information, such as news articles or social media posts, by encoding the text into numerical representations and learning patterns of language use associated with fake news. These patterns can include sentence structures, grammatical anomalies, and the presence of biased or misleading language.
- **Embedding:** in the context of fake news detection using deep learning models, refers to the process of converting textual data into dense, continuous-valued vectors. These vectors represent words or phrases and are used as input to neural networks like LSTMs.
- **NLTK (Natural Language Toolkit):** is a Python library commonly used in the context of fake news detection models based on deep learning. It provides a set of tools and resources for natural language processing (NLP), allowing developers to perform tasks like text preprocessing, tokenization, and feature extraction, which are crucial for building effective models to detect fake news in textual content

Features:

- **PortStemmer**

A Porter Stemmer is a widely used algorithm in natural language processing (NLP) and information retrieval for reducing words to their base or root form. Developed by Martin Porter in 1980, it's designed to simplify words by removing common prefixes and suffixes, helping to standardize variations of words and improve text analysis, search, and information retrieval processes. The Porter Stemmer algorithm doesn't guarantee perfect word reduction but is effective in many cases, making it a valuable tool in text processing and NLP applications.

- **One-Hot Encoding:**
  1. One-Hot Encoding is a technique used to convert categorical data, such as words or labels, into a binary (0 or 1) format.
  2. In fake news detection, it can be applied to convert words or phrases in a news article or headline into a binary vector where each word corresponds to a position in the vector, and a 1 indicates the presence of that word while 0 represents its absence.
  3. This binary representation is useful for inputting text data into machine learning models, including neural networks, which require numerical input.
- **Dropout Layer:**
  1. Dropout is a regularization technique commonly used in neural networks, including those for text classification.
  2. It helps prevent overfitting, a common issue in machine learning, by randomly deactivating a fraction of neurons (nodes) during training.
  3. In the context of a fake news detection model, a dropout layer can be added to the neural network architecture to prevent it from relying too heavily on specific words or patterns in the training data. This promotes generalization and improves the model's ability to identify fake news in new, unseen data

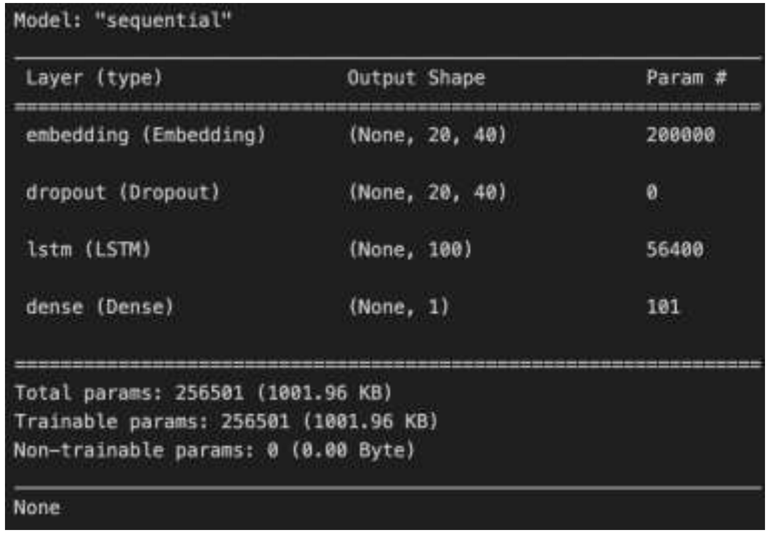
About the Model:

- **Sequential:**

1. In fake news detection, a "sequential" model typically refers to a type of neural network architecture used to process sequential data, such as text. Sequential models are designed to capture the order and context of words or tokens in a piece of text. They are well-suited for tasks like natural language processing (NLP).
2. One common example of a sequential model is the Recurrent Neural Network (RNN), which processes input data one element at a time while maintaining a hidden state that captures information from previous elements in the sequence. This allows it to model dependencies and context within the text data, which can be essential for identifying fake news that relies on nuanced language or subtle cues.

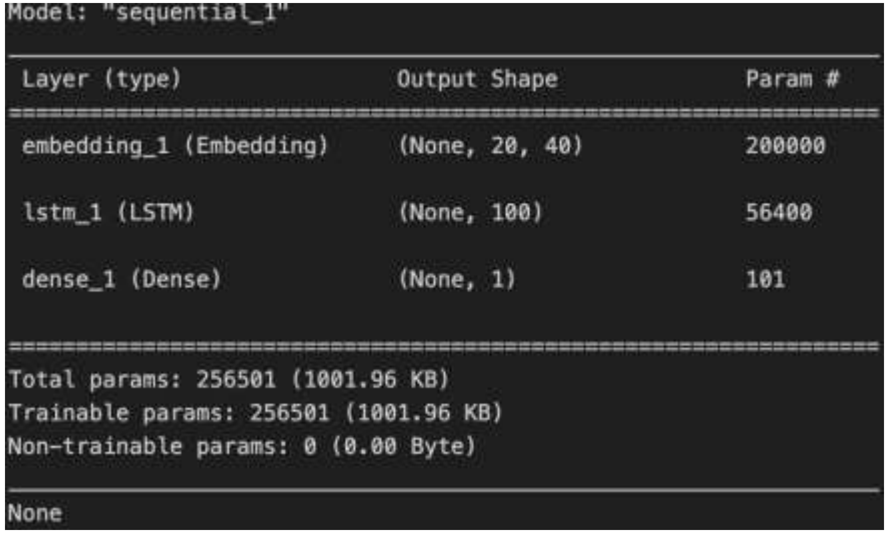
- **Dense:**

In the context of fake news detection, "dense" typically refers to dense layers, which are also known as fully connected layers. Dense layers are responsible for learning complex patterns and relationships within the data. They take the output from previous layers (such as sequential layers in an NLP model) and connect each neuron to every neuron in the subsequent layer. This interconnection allows the network to capture and represent high-level features and relationships in the data.



```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 embedding (Embedding)       (None, 20, 40)            200000

 dropout (Dropout)           (None, 20, 40)            0

 lstm (LSTM)                 (None, 100)               56400

 dense (Dense)               (None, 1)                 101

=================================================================
Total params: 256501 (1001.96 KB)
Trainable params: 256501 (1001.96 KB)
Non-trainable params: 0 (0.00 Byte)
_________________________________________________________________
None
```

The picture on the left is the model with the sigmoid activation function & dropout layer and the picture on the right is the one without the dropout layer and relu activation function.



```
Model: "sequential_1"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 embedding_1 (Embedding)     (None, 20, 40)            200000

 lstm_1 (LSTM)               (None, 100)               56400

 dense_1 (Dense)             (None, 1)                 101

=================================================================
Total params: 256501 (1001.96 KB)
Trainable params: 256501 (1001.96 KB)
Non-trainable params: 0 (0.00 Byte)
_________________________________________________________________
None
```

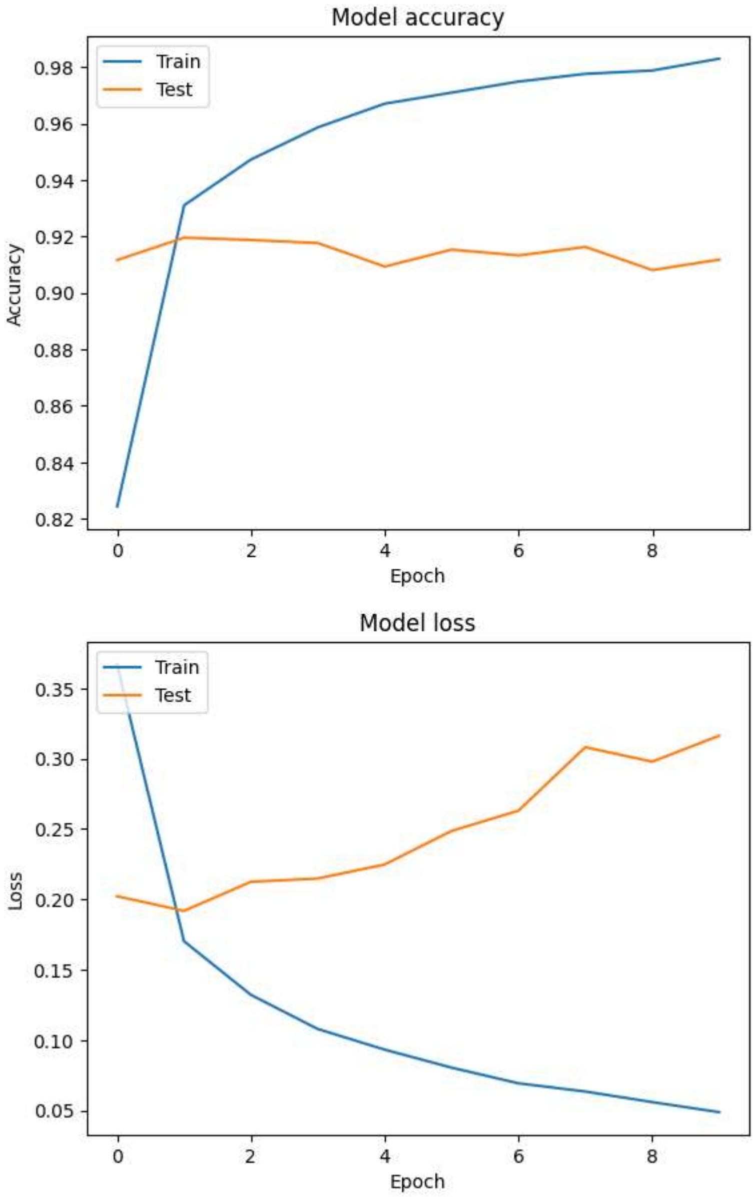
Changes made to base code:

- In the first LSTM iteration which is part of the base code I have added a dropout of 0.6 and have used sigmoid as my activation function. The accuracy rate is 98.53%

- In the second LSTM iteration which is part of my optimisation , I have removed dropout and have changed relu as my activation function. The accuracy rate is 99.72%

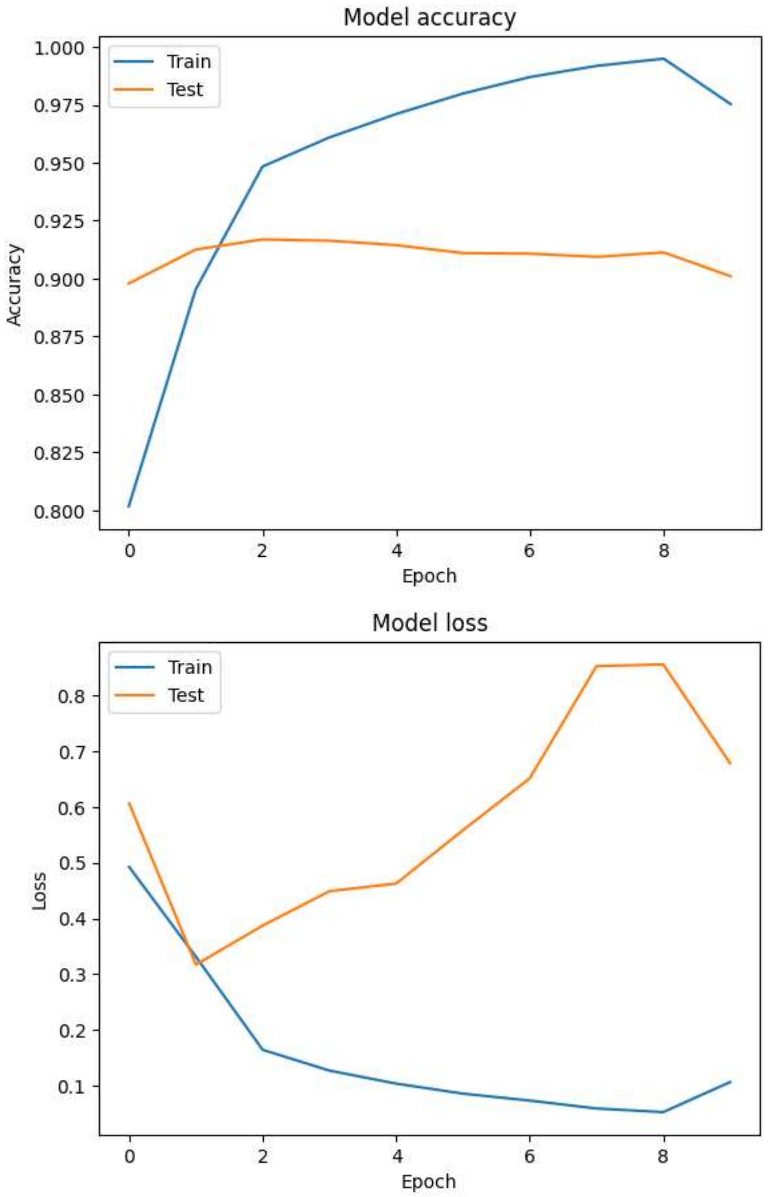
- The embedded features have been fixed at 40 at both instances

Results:



BASE
CODE

ACCURACY:
98.53%



OPTIMIZED
CODE

ACCURACY:
99.6%

REFERENCES:

- **Reference Paper-**Implementation of Fake News Detection Using Long Short Term Memory Method Base on Android
- **REFERENCE PAPER LINK:** https://ieeexplore.ieee.org/document/9768283
- **Dataset Link:** https://github.com/TrilokiDA/Fake-News-Classifier-LSTM/blob/master/Fake%20News%20Classifier-LSTM.ipynb

(the file named train.csv in the github repository is the dataset used)