DEEP LEARNING

FAKE NEWS DETECTION

Group – 7:

SAI KIRAN A

VAMSI ABHIRAM GURU

VIKAS REDDY K

University at Buffalo The State University of New York

The purpose – What and why

Fake news refers to false or misleading information presented as genuine news, often created and disseminated with the intent to deceive or manipulate public opinion. It can take various forms, including fabricated stories, misinformation, or misleading headlines, and is spread through traditional media, social media, or other online channels.

Why should we detect fake news?

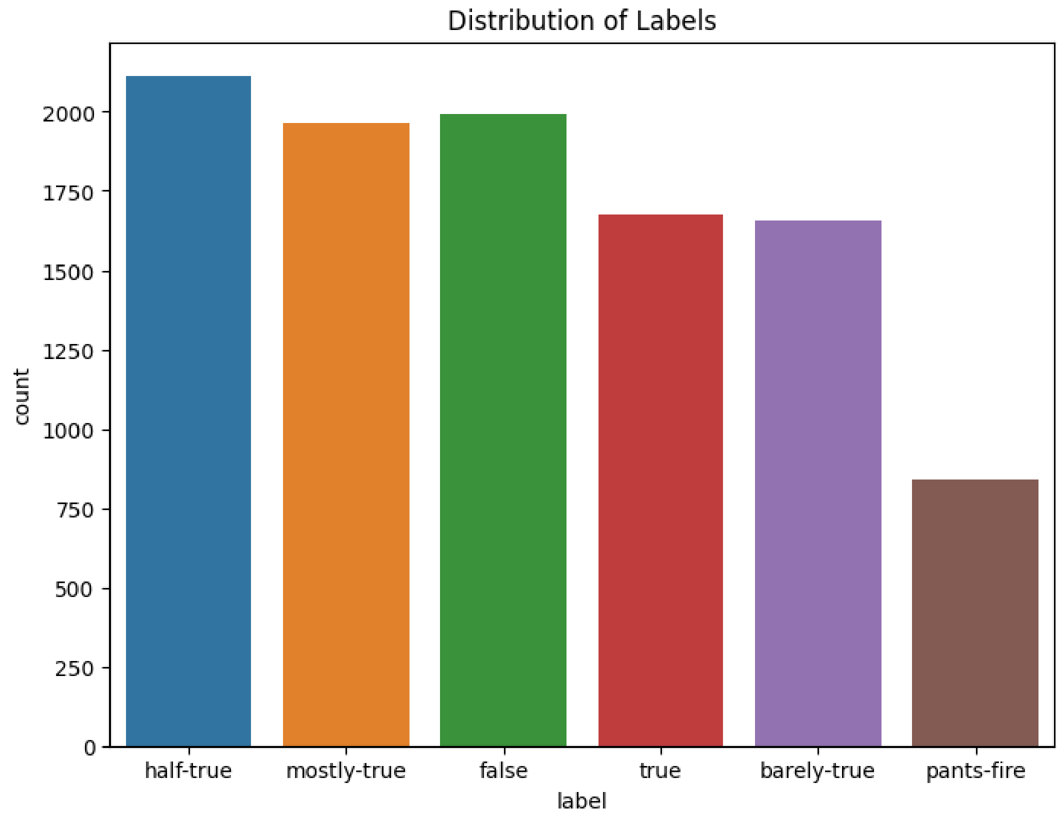
1. *People deserve the truth.*
2. *Fake news destroys people's credibility.*
3. *Real news benefits people.*

About the Dataset

- *The LIAR dataset, also known as "LIAR-PLUS," is a freely accessible dataset with a particular focus on political remarks. It is intended for use in fact-checking and the identification of false news.*
- *The LIAR dataset was developed to support investigations into the identification of deceit in political discourse and fact-checking. It attempts to offer a standard for creating and assessing machine learning models that can determine if comments made by politicians are true.10240 news reports total, each classified by category, are included in the collection.*
- *Natural language processing, deception detection, and false news detection research papers have all made extensive use of LIAR. It functions as a standard by which to measure how well machine learning models perform in the field of political fact-checking.*

University at Buffalo The State University of New York

EDA - Before going into models

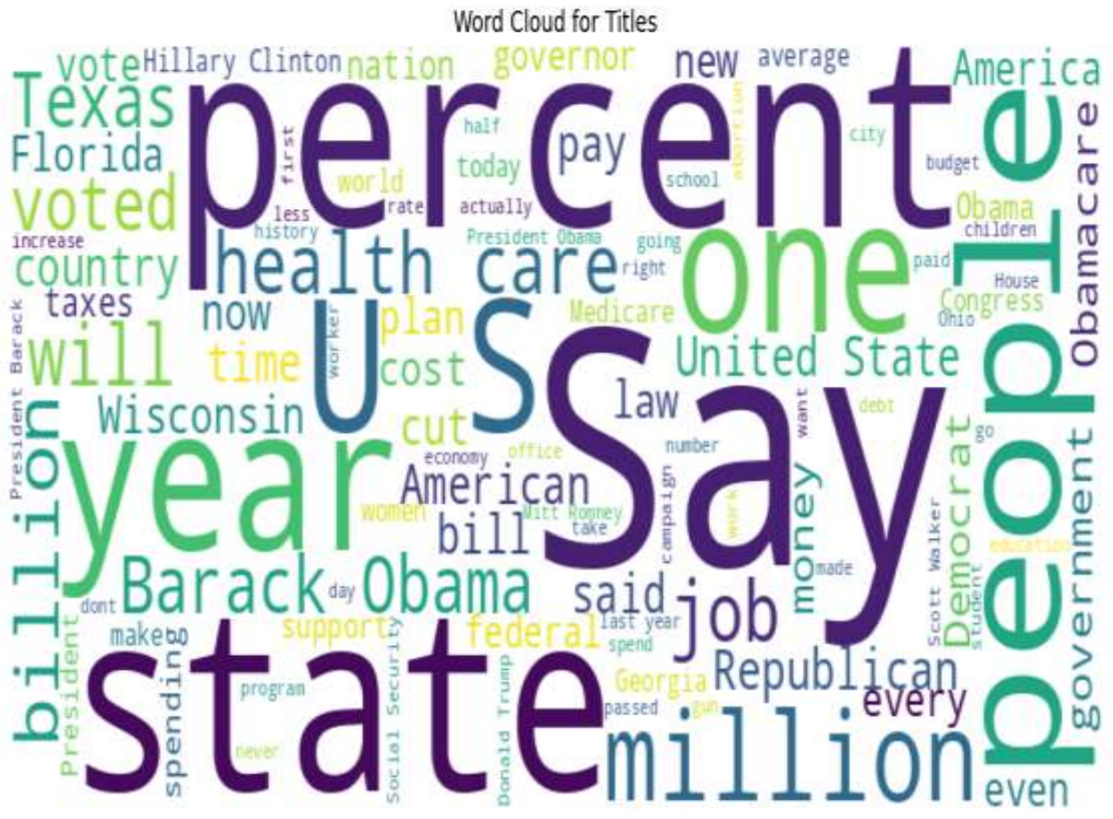


News are of 6 kinds in this data set:

1. *True*
2. *Mostly True*
3. *Half True*
4. *Barely True*
5. *False*
6. *Pants Fire*

This graph represents the counts of each kind of news in the LIAR dataset. Half true news are the highest in count so of course it’s quite hard to determine whether a news is True or not just by seeing the news.

EDA – Word Count



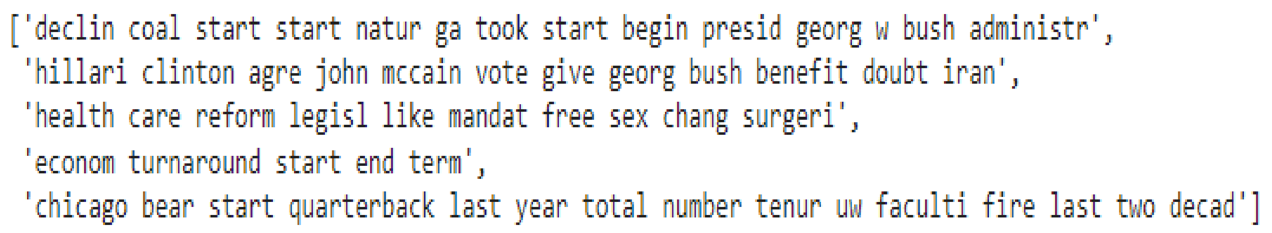
We did word cloud for the news titles. So what we did here is the word count analysis which is a fundamental step in detecting the fake news for reasons such as:

1. *Text representation.*
2. *Feature Engineering.*
3. *Preprocessing.*
4. *Cleaning.*
5. *Integration with other upcoming features.*
6. *Understanding content structure (pattern).*

Data Preprocessing

The steps we used for data preprocessing are:

1. *Role of stemming.*
2. *Dropping NaN and NULL cells.*
3. *Removing stop words.*
4. *Converting everything into lower case.*
5. *Replace unnecessary characters with spaces.*
6. *And finally, porter stemming.*



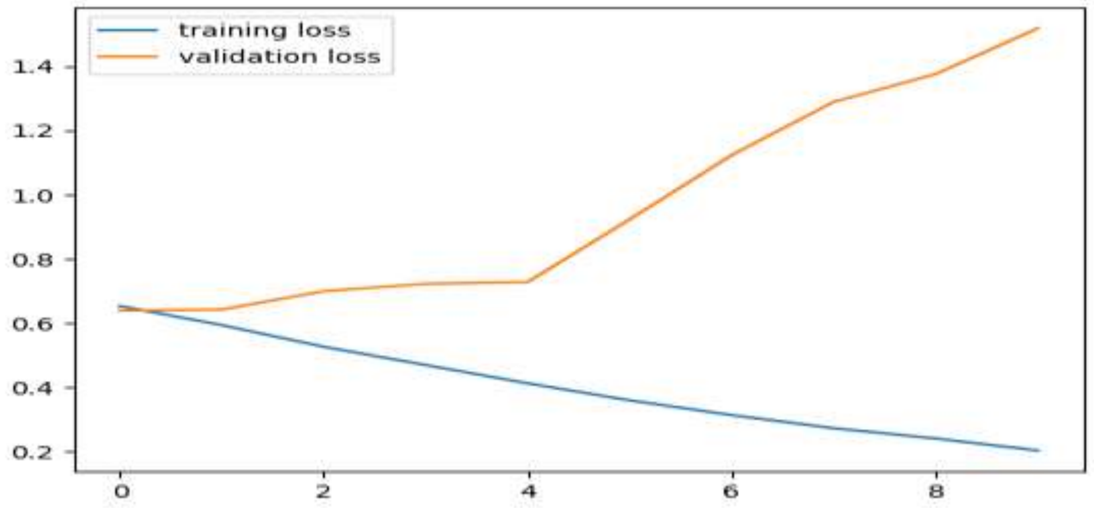
```
['declin coal start start natur ga took start begin presid georg w bush administr',
 'hillari clinton agre john mccain vote give georg bush benefit doubt iran',
 'health care reform legisl like mandat free sex chang surgeri',
 'econom turnaround start end term',
 'chicago bear start quarterback last year total number tenur uw faculti fire last two decad']
```

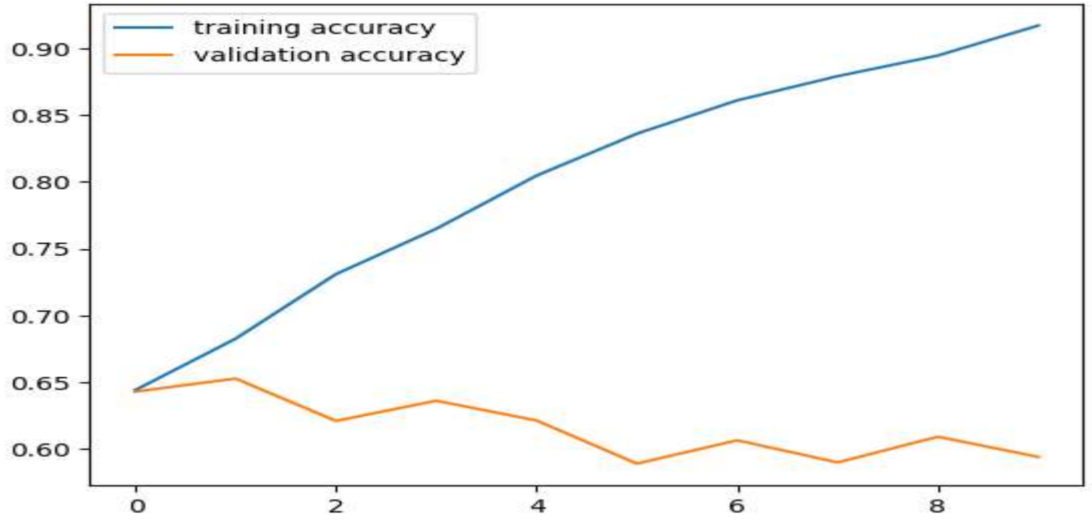
Models we used and Why

- ***LSTM and GRU:*** *We used Gated Recurrent Unit (GRU) and Long Short-Term Memory (LSTM) models because they are good at capturing sequential dependencies in textual data. Since fake news frequently demonstrates subtle patterns and nuanced language usage, these models are excellent at recognizing long-range relationships, which enables us to gather complex contextual data.*
- ***Transformer Model (BERT):*** *In terms of natural language processing tasks, the Transformer architecture—which is demonstrated by models such as BERT—offers cutting edge performance. Our ability to grasp the text's complicated relationships is made possible by its attention mechanisms, which facilitate bidirectional context interpretation.*
- ***Simple RNN :*** *Even though LSTMs and GRUs are thought to be more complex than Simple Recurrent Neural Networks (RNNs), we added RNNs for comparison and to investigate their potential in our setting. By using simple RNNs, we can also investigate how varying levels of model complexity affect the efficacy of false news detection. Simple RNNs may still be able to capture some patterns effectively.*

LSTM

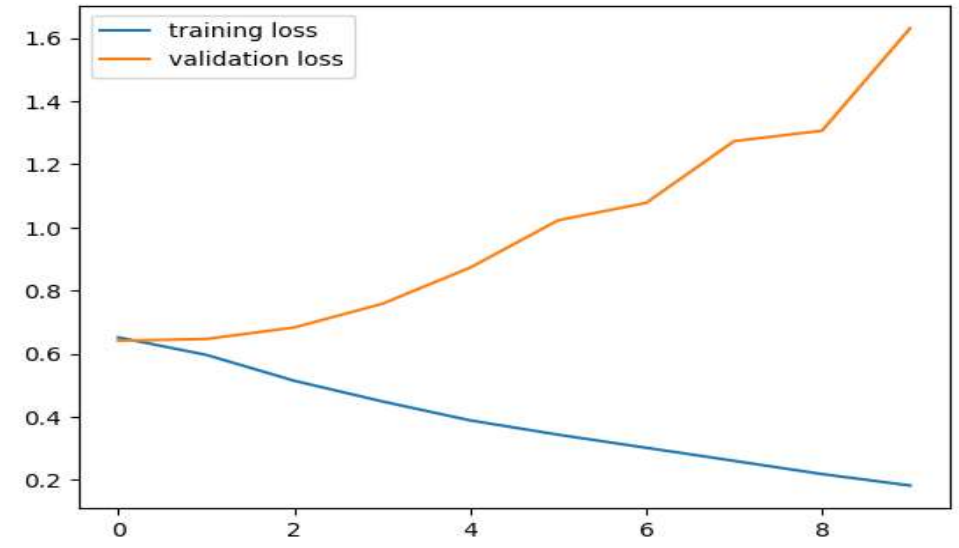
We created a fake news detection model using a Sequential model in Keras. The model consists of an Embedding layer with 40-dimensional vectors, followed by an LSTM layer with 100 neurons and a Dense layer with a sigmoid activation for binary classification. The model is compiled with binary cross entropy loss and the Adam optimizer. Training is performed on a dataset split into training and testing sets, and the model's performance is evaluated over 10 epochs. Finally, the trained model is saved to a file, and predictions are made on the test set, with a threshold of 0.5 for binary classification. Using LSTM, we attained the accuracy of 69%.

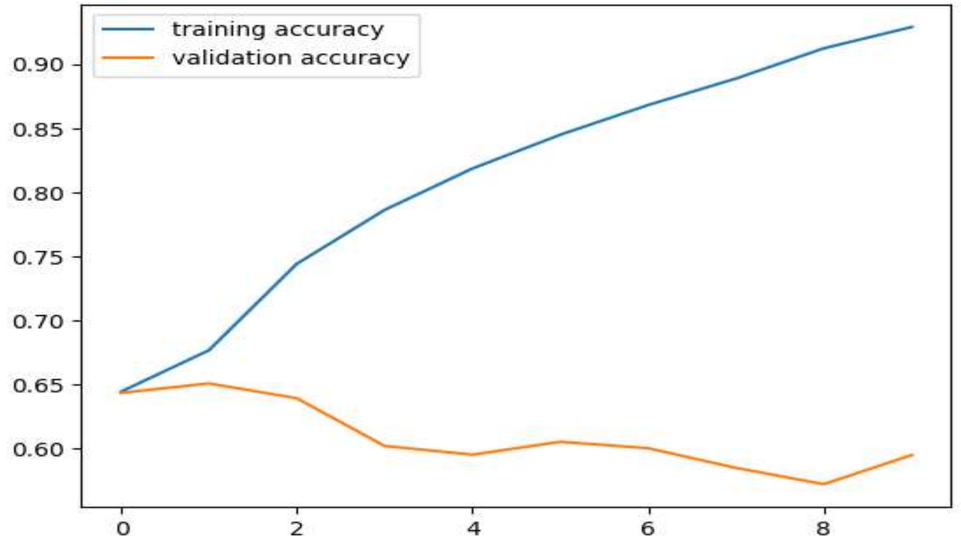




GRU

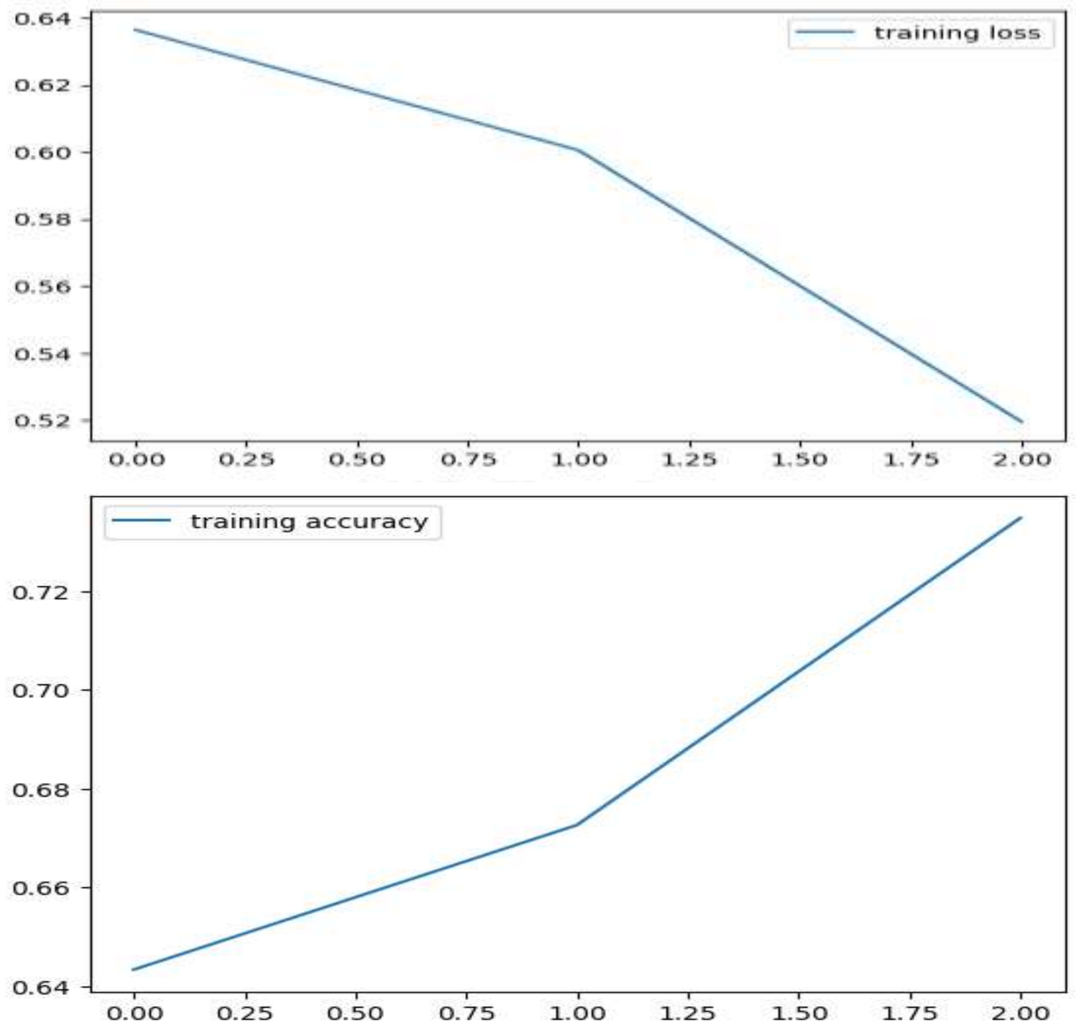
- *We constructed a fake news detection model using a Gated Recurrent Unit (GRU). The model architecture starts with an embedding layer, converting words into numerical vectors. Following that, a GRU layer with 100 neurons is incorporated to capture sequential patterns in the text efficiently. The model is configured for binary classification using binary crossentropy loss and the Adam optimizer. Training is executed on the provided datasets (X_train and y_train), with validation on a separate test set (X_test and y_test). The training spans 10 epochs with a batch size of 64. Using GRU we attained the accuracy of 68%.*





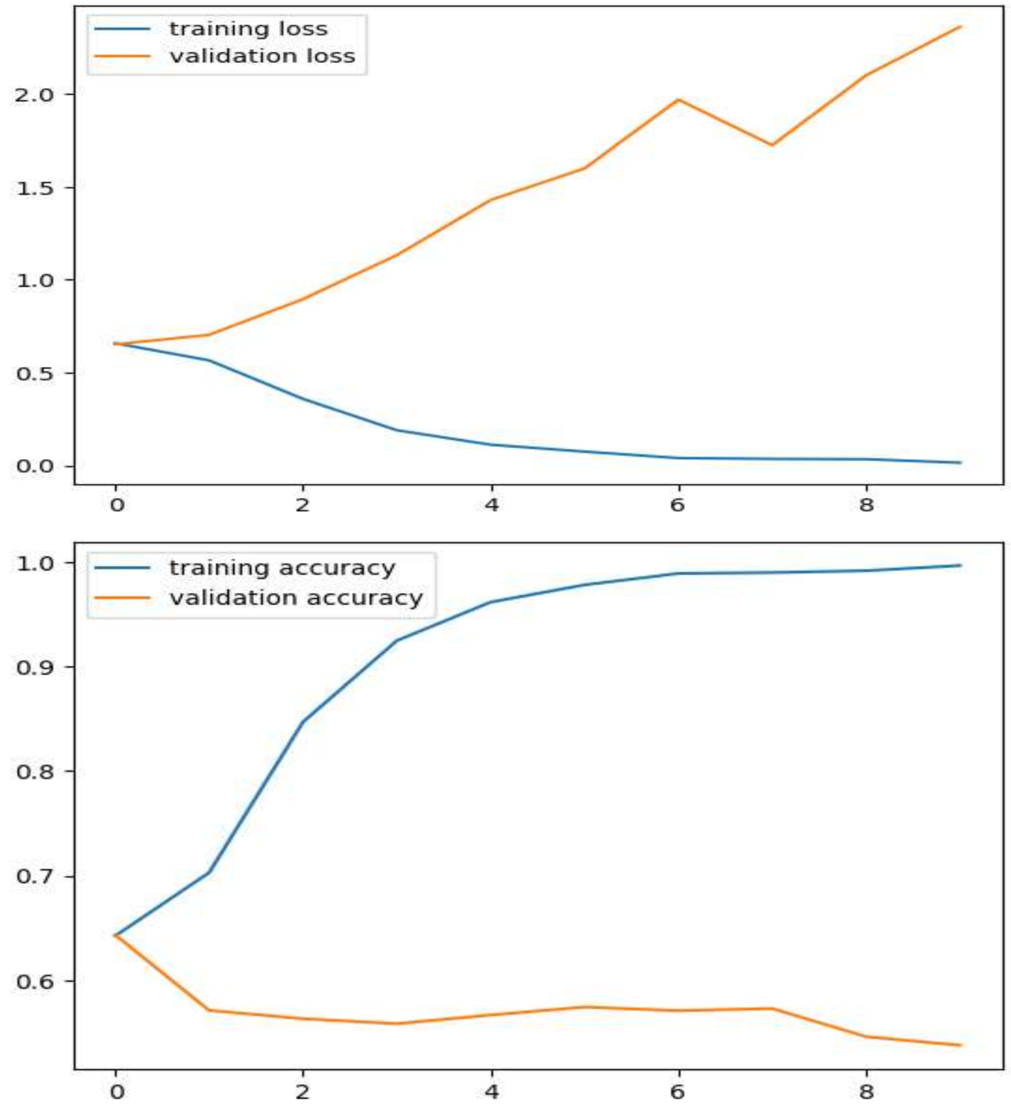
Transformer Model (BERT)

We are preprocessing a Data Frame column with titles using the Bert Tokenizer. To produce input tensors, the titles are tokenized, padded, and truncated. For sequence classification, we loaded a BERT model that had already been trained. The sparse categorical cross-entropy loss, accuracy as a metric, and an Adam optimizer are used to assemble this model. To classify titles into predetermined categories, we trained a BERT-based model. Using Transformer Model (BERT) we attained the training accuracy of 84% for 25% of data. We used this model's output logits to determine the predicted label and probabilities associated with each class. So, based on a threshold (0.5), our code tells whether the statement is considered mostly true or true, or false or barely true, according to the model's prediction. For BERT model we attained the accuracy of 65%.



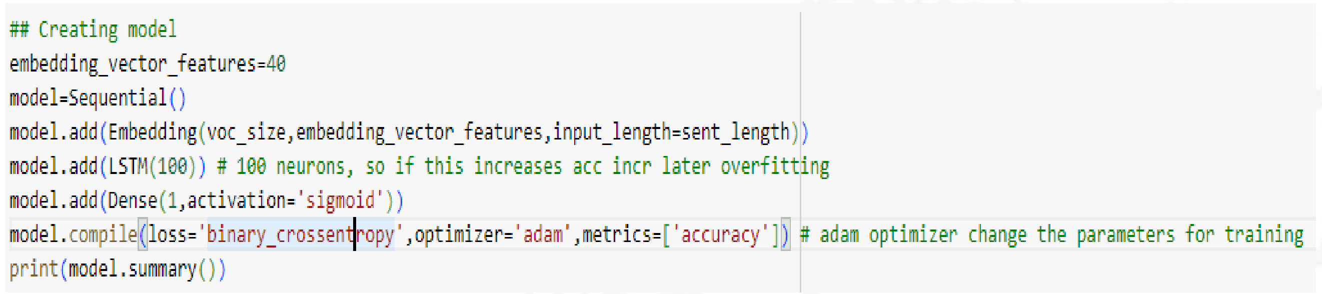
Simple RNN

We used Simple RNN model for fake news detection. The model starts with an embedding layer for converting words into numerical vectors, followed by a Simple RNN layer with 100 neurons to capture sequential patterns in the text. I used binary crossentropy loss and the Adam optimizer for efficient training. The summary of the model architecture is printed for clarity. The model is then trained on the provided datasets (X_train and y_train), validated on a separate test set (X_test and y_test), and runs for 10 epochs with a batch size of 64. The training history is stored for further analysis. Using RNN we got the accuracy 61%.

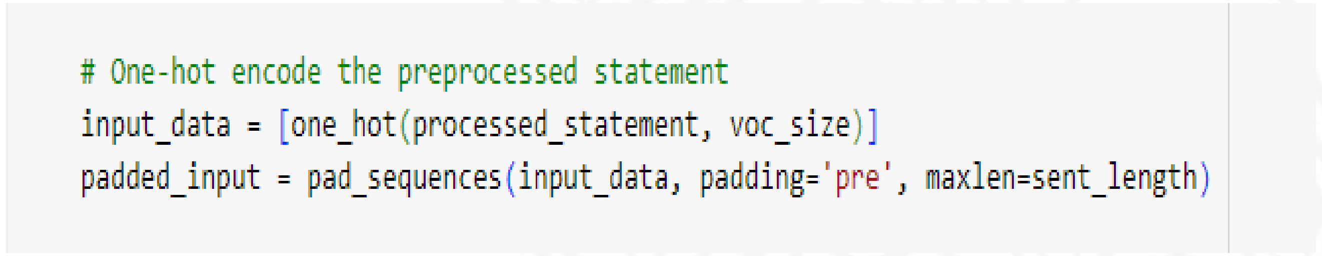


In the Code

- *We used one-hot encoding to convert the preprocessed statement into a format that can be input into a neural network for prediction.*
- *We used Adam optimizer with default parameters except for the BERT model we gave the variable learning rate.*
- *In all the models we used encoder as Embedding layer. Adam optimizer adapts the learning rate for each parameter individually based on their past gradients, efficiently navigating the optimization landscape.*
- *The encoding is implemented through embedding layer that is functioned as (Embedding(voc_size, embedding_vector_features, input_length=sent_length)) .*

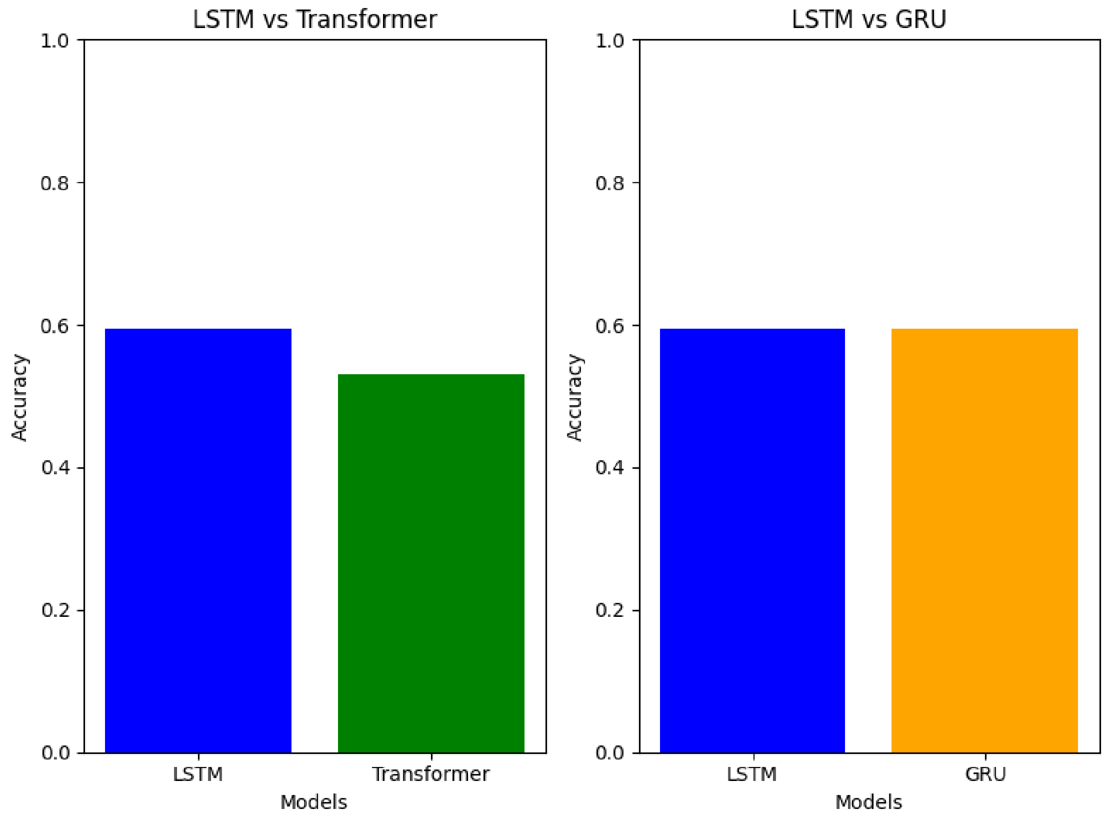


```
## Creating model
embedding_vector_features=40
model=Sequential()
model.add(Embedding(voc_size,embedding_vector_features,input_length=sent_length))
model.add(LSTM(100)) # 100 neurons, so if this increases acc incr later overfitting
model.add(Dense(1,activation='sigmoid'))
model.compile(loss='binary_crossentropy',optimizer='adam',metrics=['accuracy']) # adam optimizer change the parameters for training
print(model.summary())
```

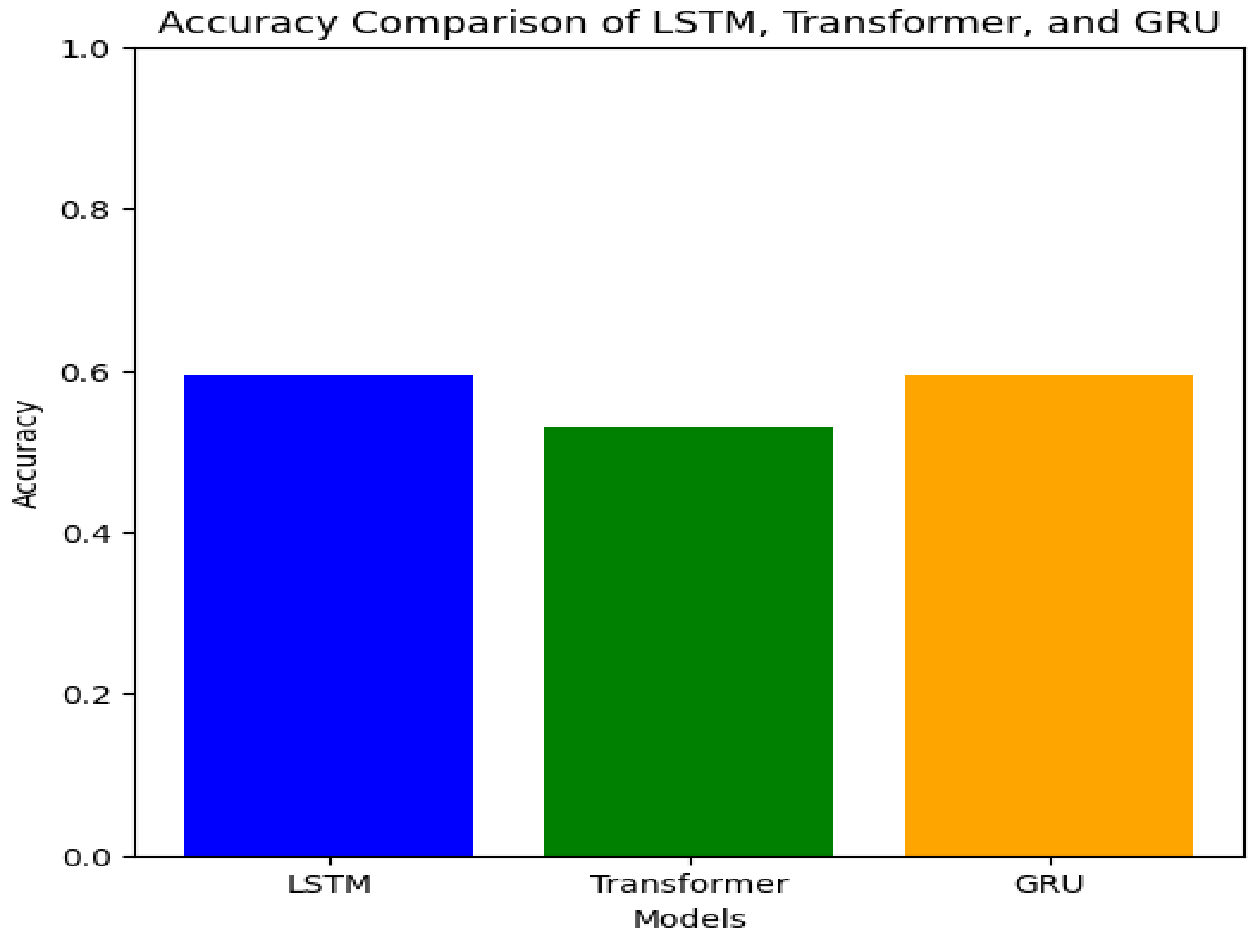


```
# One-hot encode the preprocessed statement
input_data = [one_hot(processed_statement, voc_size)]
padded_input = pad_sequences(input_data, padding='pre', maxlen=sent_length)
```

Comparison Among 2



Comparison


Accuracy Comparison of LSTM, Transformer, and GRU
1.0
0.8
0.6
0.4
0.2
0.0
Accuracy
LSTM
Transformer
GRU
Models

Differences

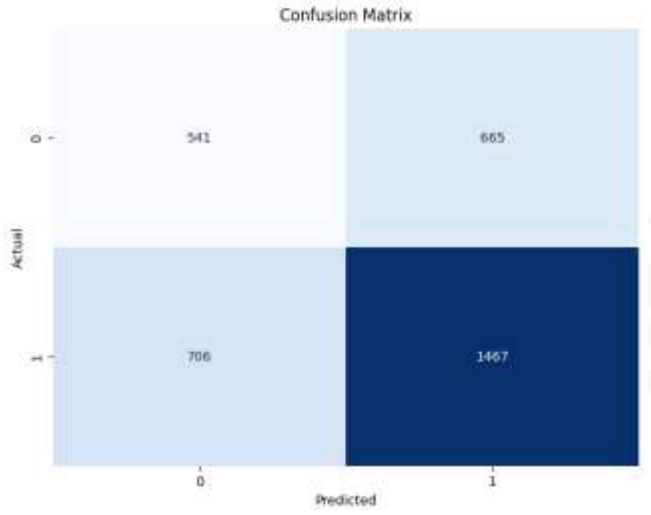
MODEL	Accuracy	Loss
LSTM	69%	1.51
Transformer	65%	1.58
RNN	61%	2.36
GRU	68%	1.60

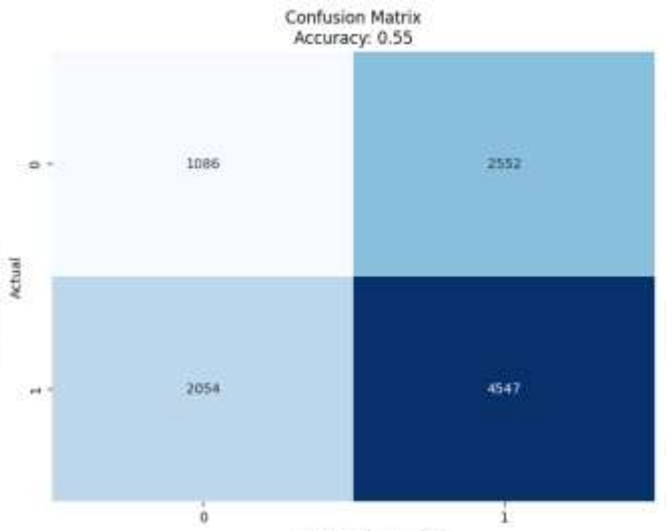
University at Buffalo The State University of New York

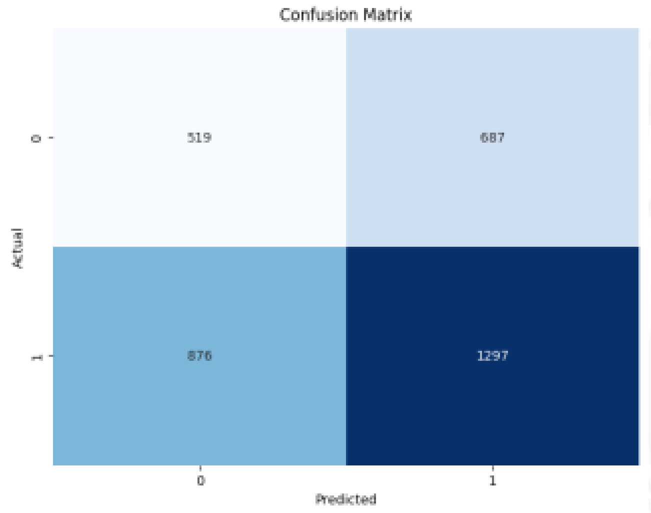
Analysis

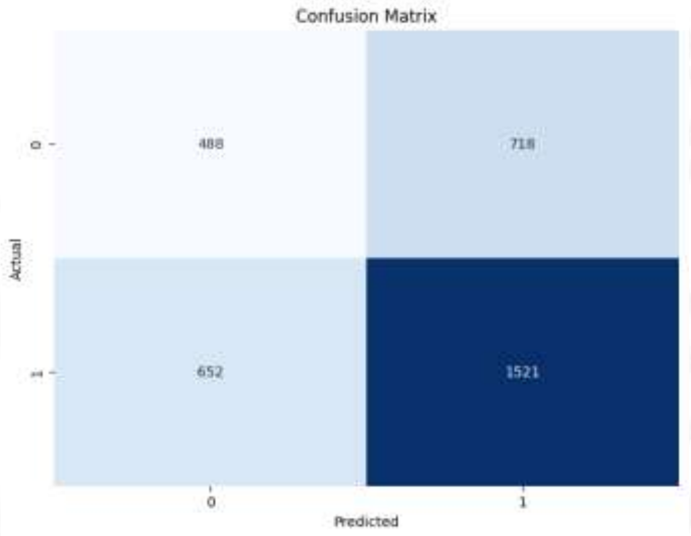
LSTMs can retain information over extended sequences, allowing them to understand the context of a statement within a larger context, which can be valuable in discerning fake news.

Transformers process input data in a bidirectional manner, considering both left and right context. This enables them to capture a wide range of contextual information. But LSTM model improvises in long range dependencies.

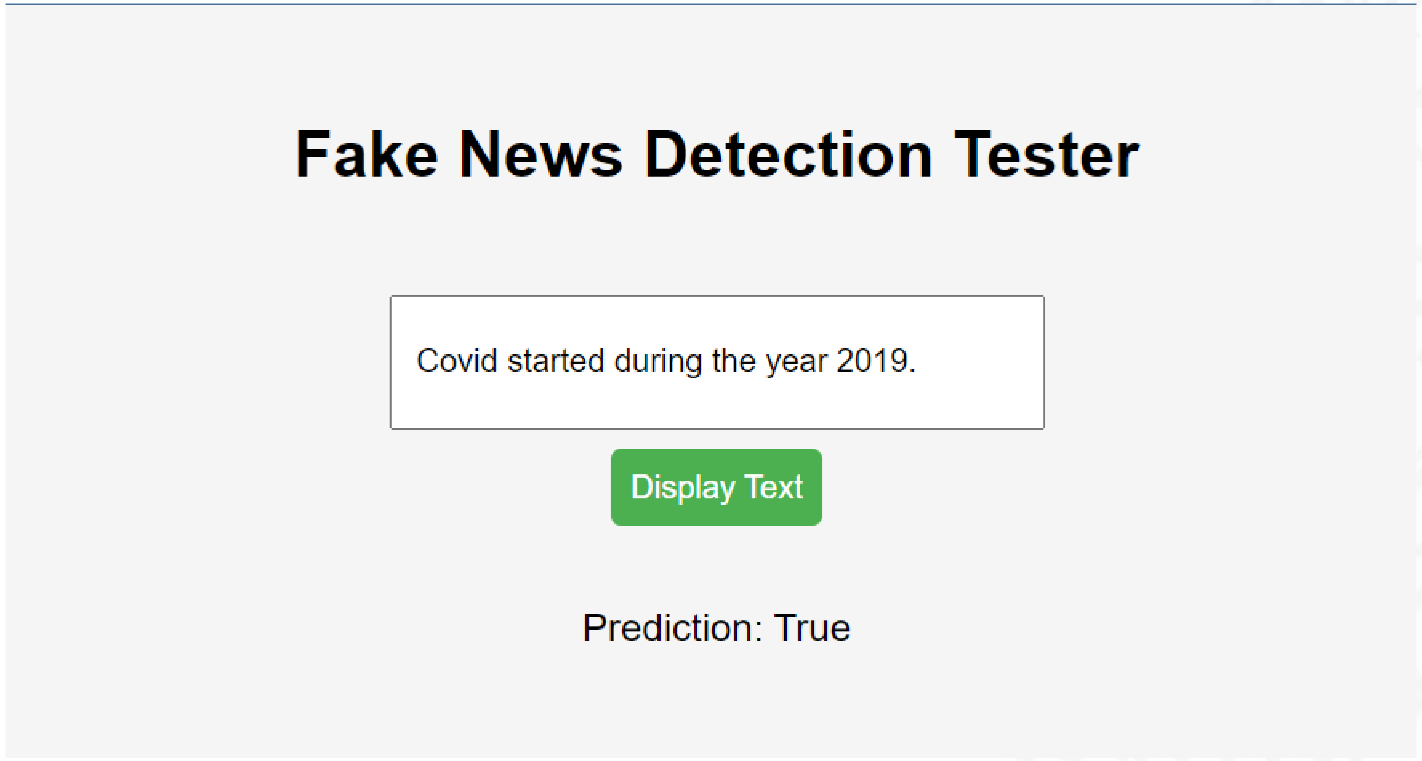








UI Demo - 1

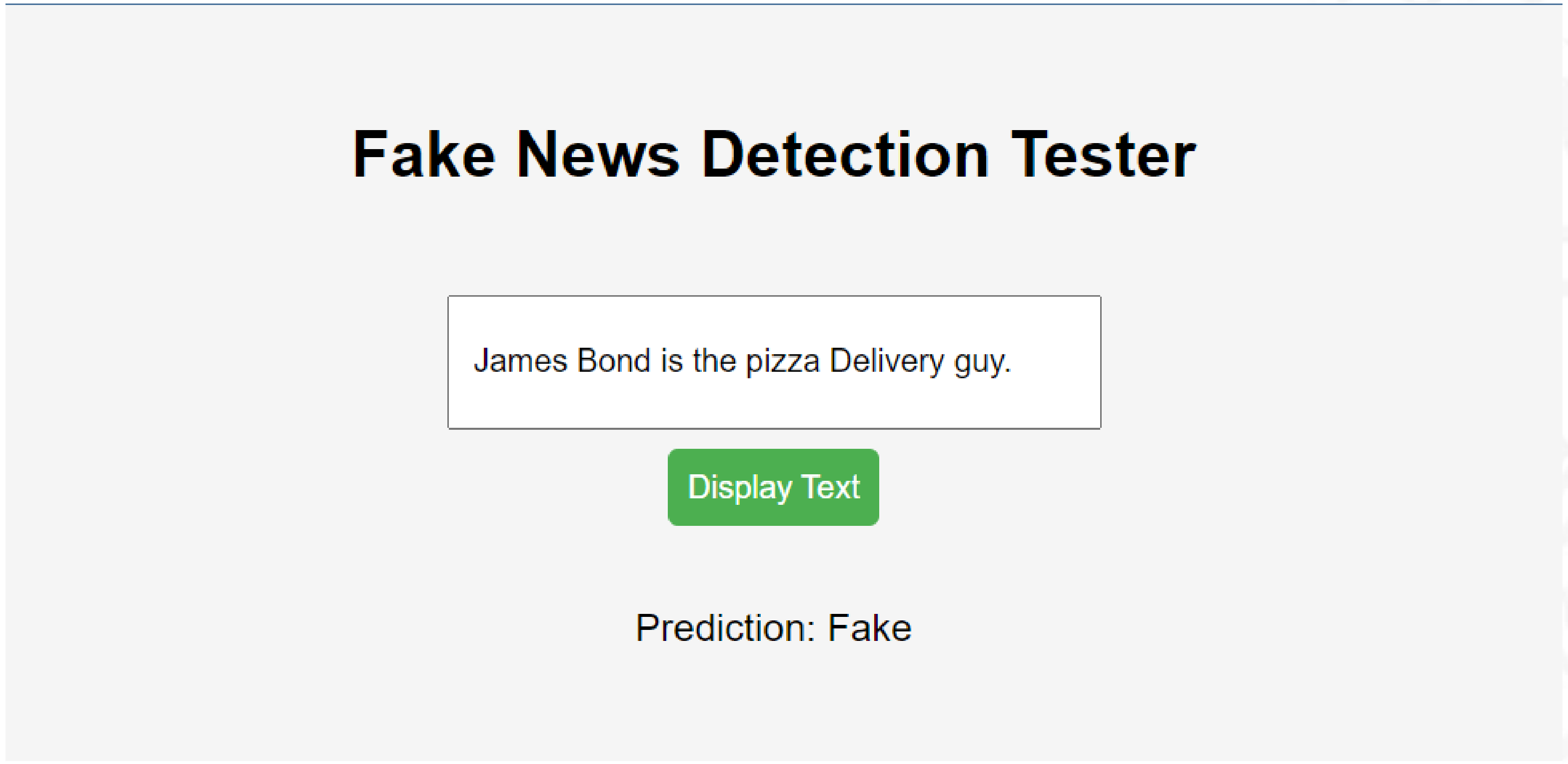


Fake News Detection Tester

Covid started during the year 2019.

Prediction: True

UI Demo 2

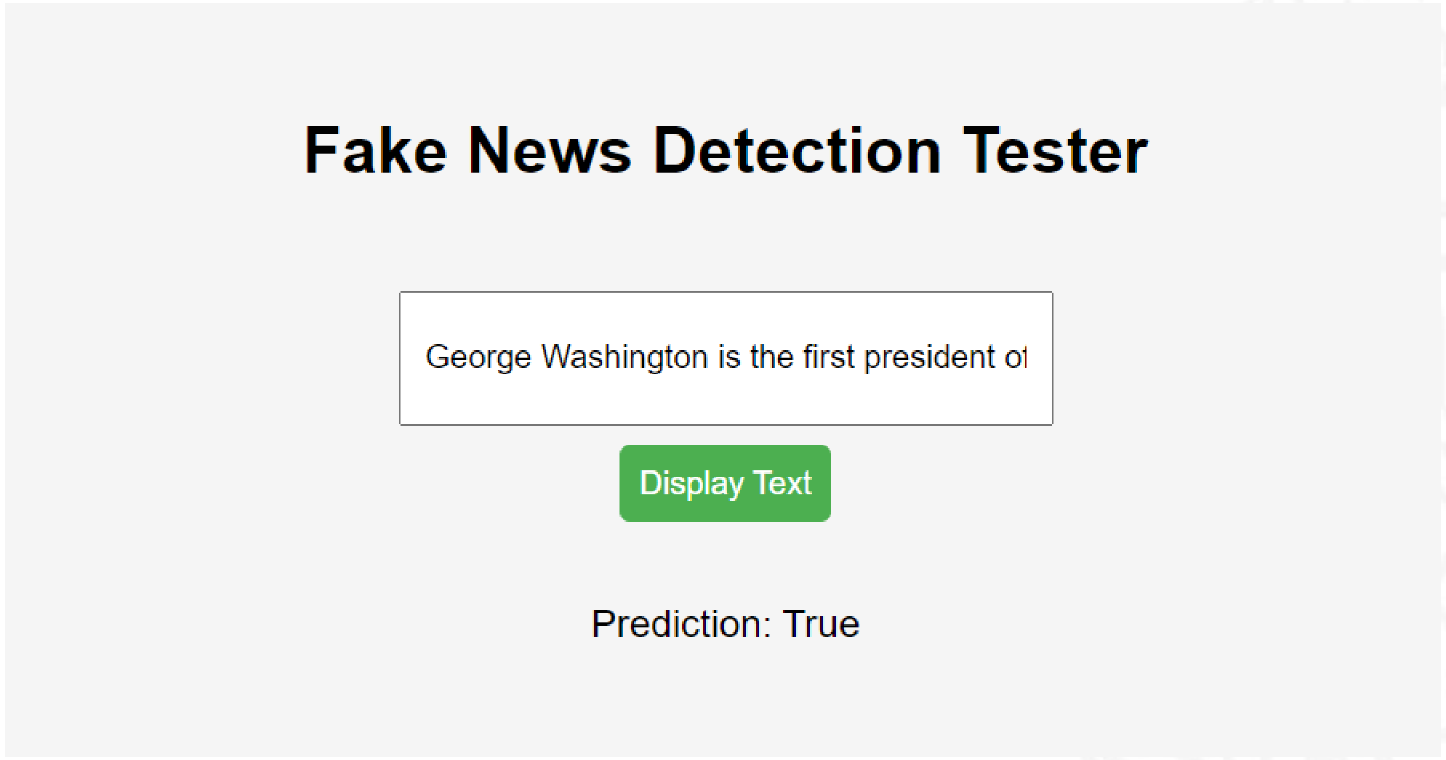


Fake News Detection Tester

James Bond is the pizza Delivery guy.

Prediction: Fake

UI Demo - 3

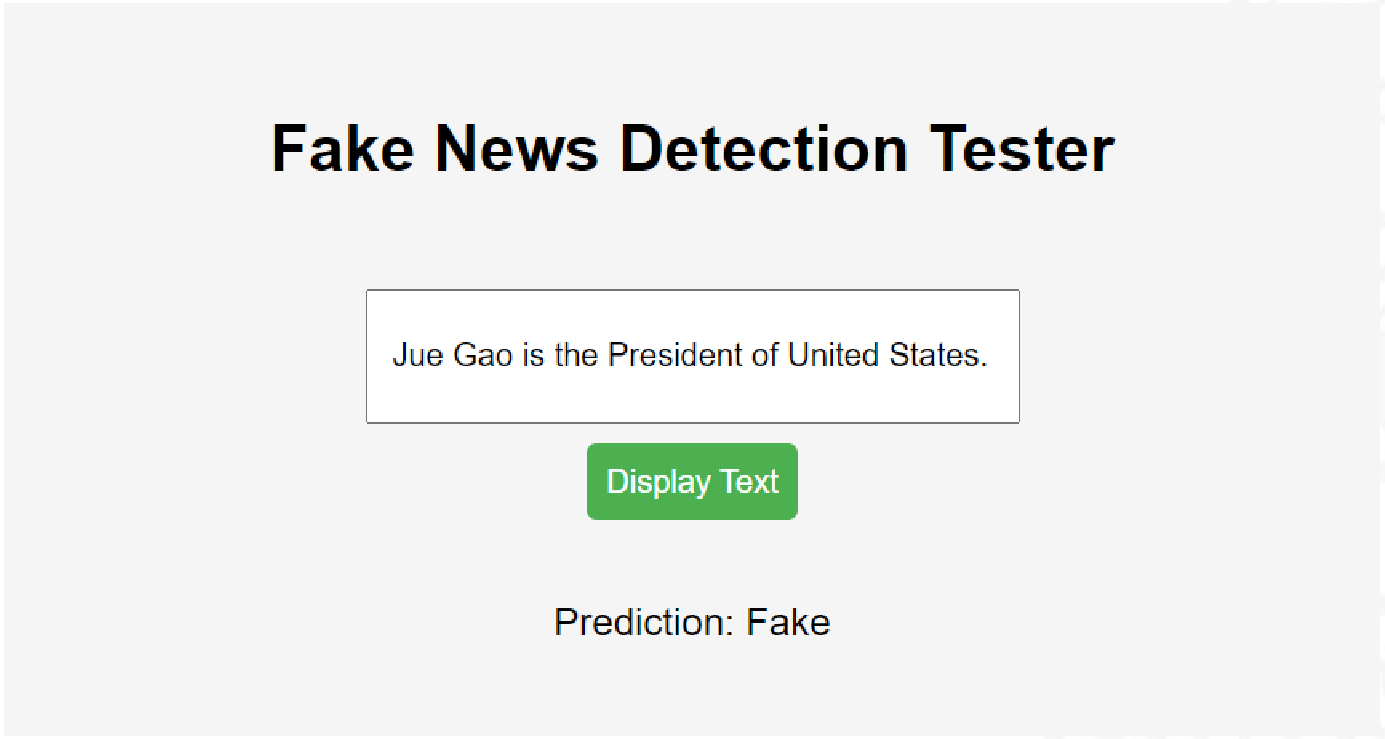


Fake News Detection Tester

George Washington is the first president of

Prediction: True

UI Demo - 4


Fake News Detection Tester

Jue Gao is the President of United States.

Prediction: Fake

University at Buffalo The State University of New York

Github Link

https://github.com/Saikiran-git/DeepLearning_Fake_News_Detection.git

THANK YOU