

The Daily News

Thursday 09/11/2023

News

Edition: 001

Guardians of Authencity: AI Edition - Distinguishing True News from False Narratives



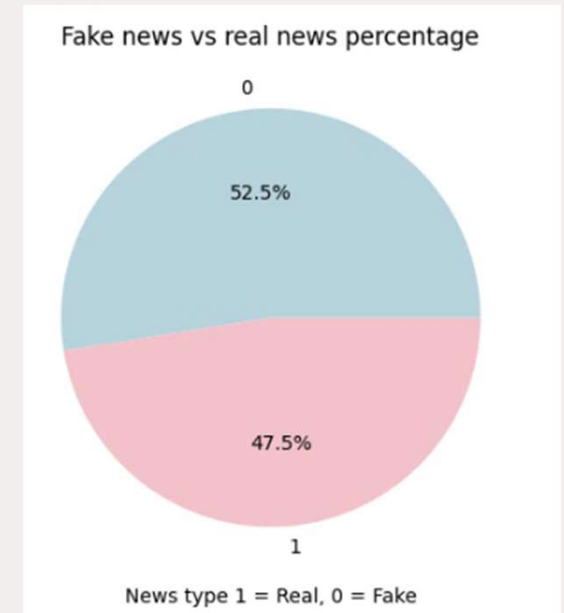
By Jyotsna, Priya, Taryn, Asha

Front page news

Thursday 09/11/2023

Edition: 001

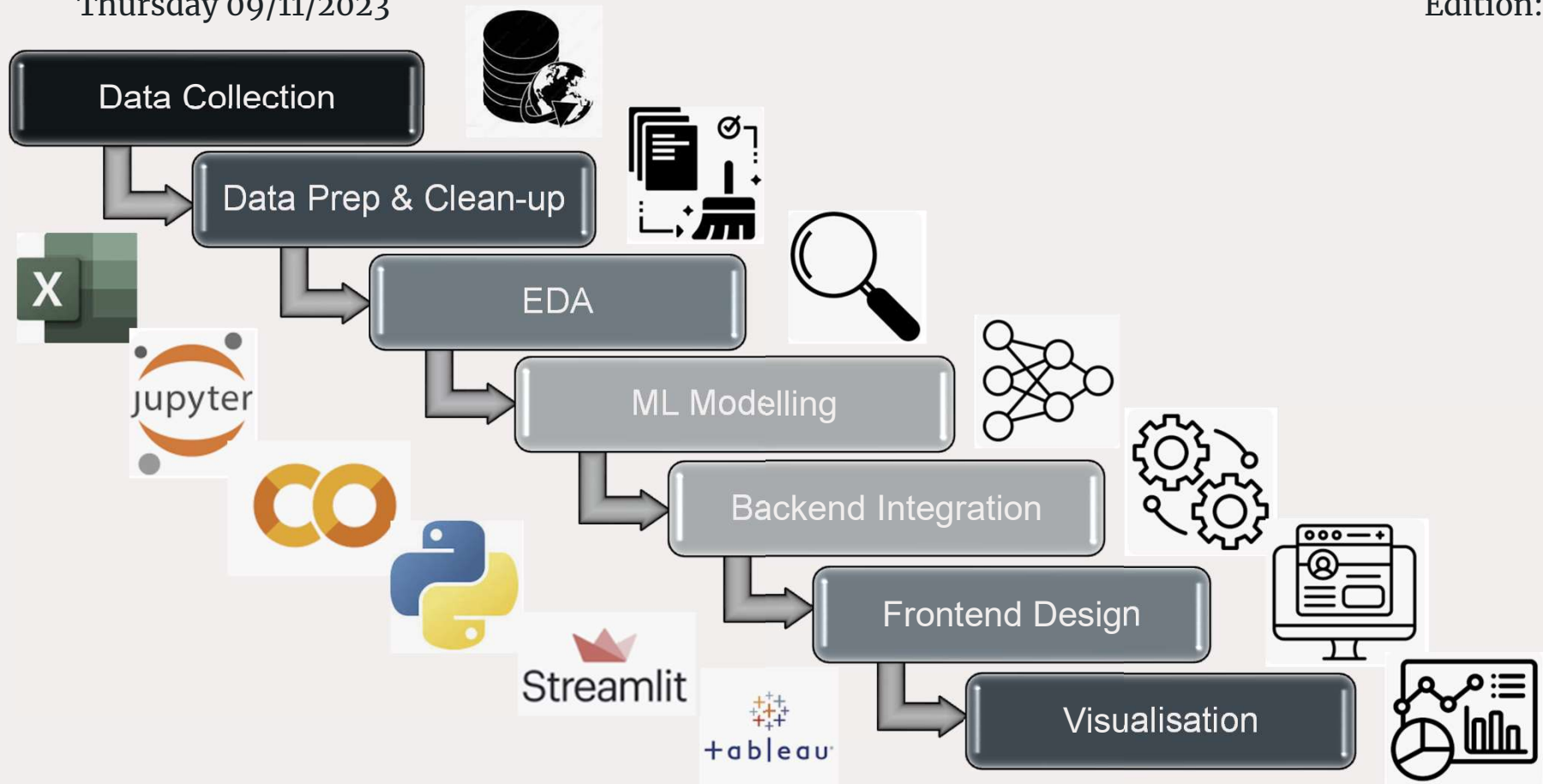
- **The problem**
The challenge of differentiating between real and fake news
- **The objective**
To create a tool that can classify news articles as real or fake using machine learning techniques
- **The data**
 - Fake news articles: 23,478
 - Real news articles: 21,209
 - Primary source from Reuters
 - Over 90% articles from the USA



Process Overview

Thursday 09/11/2023

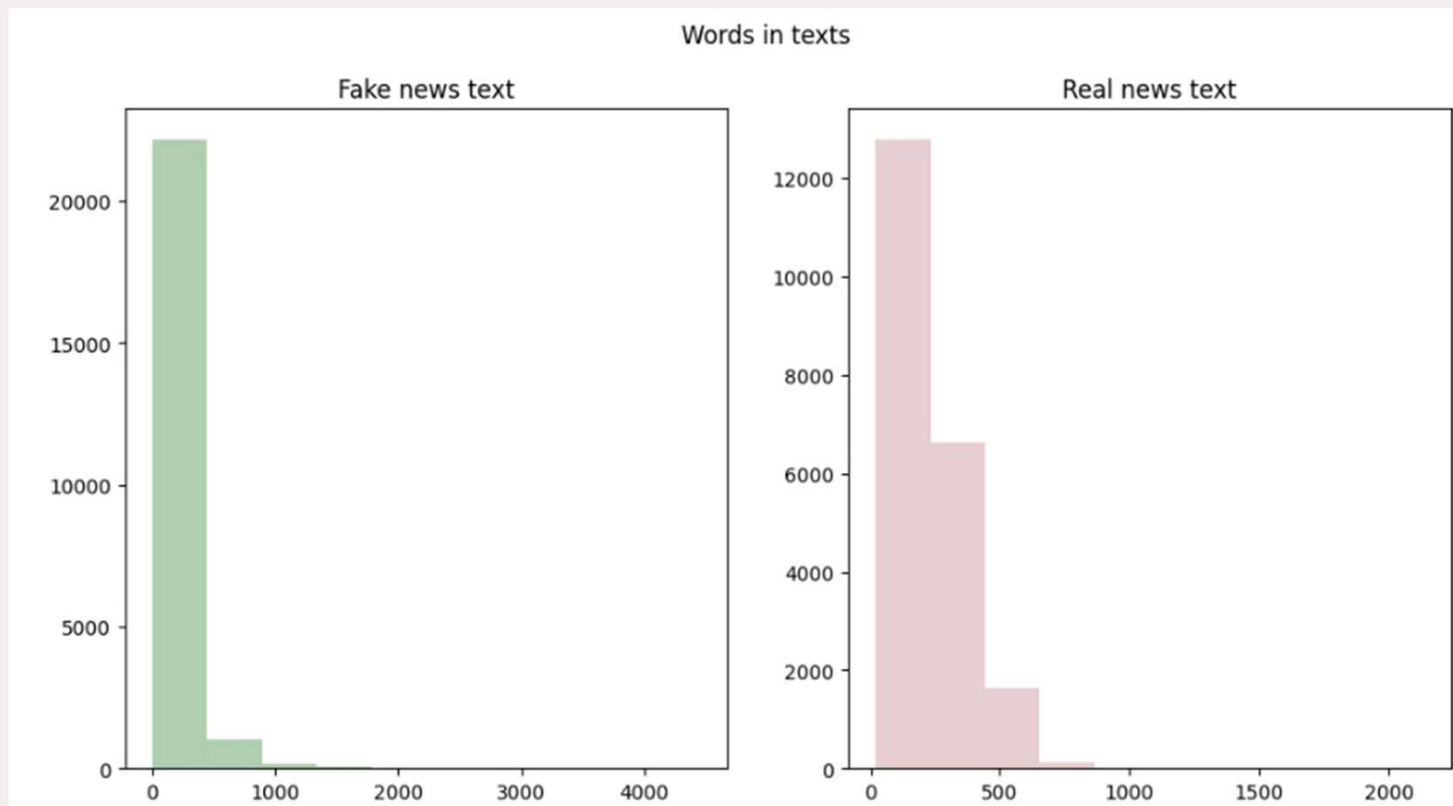
Edition: 001



Words Count

Thursday 09/11/2023

Edition: 001



Natural Language Processing

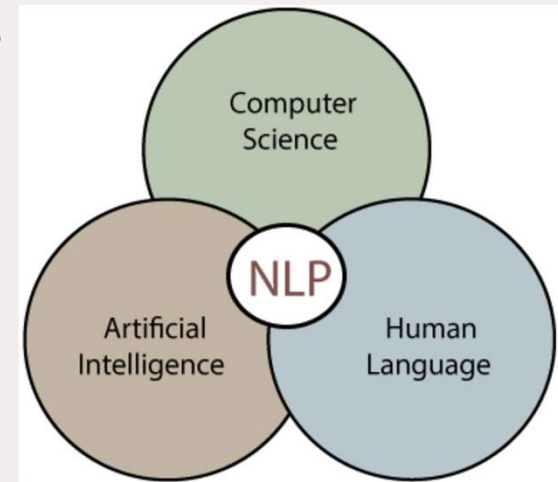
Thursday 09/11/2023

Edition: 001

NLP: Making sense of human language

Key Processes Explained:

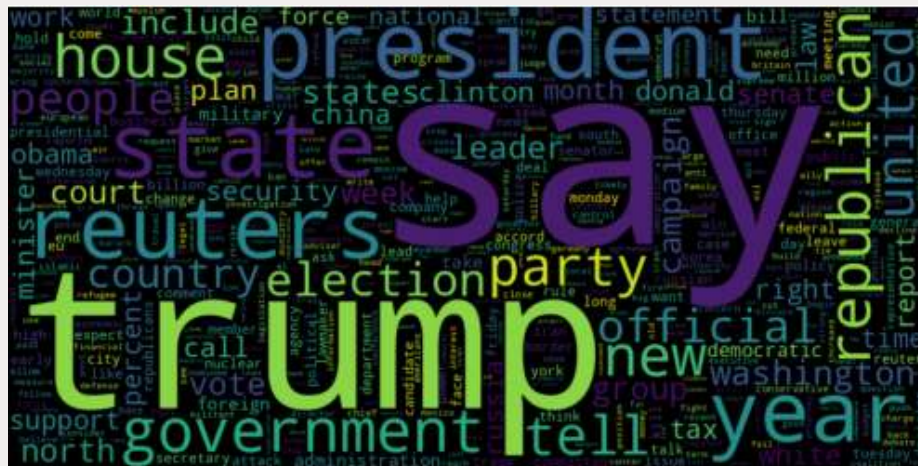
- Tokenization: Breaking text into words
- Cleaning: Removing special characters and punctuation
- Lemmatization: Condensing words to their root forms



Most Common Words

Thursday 09/11/2023

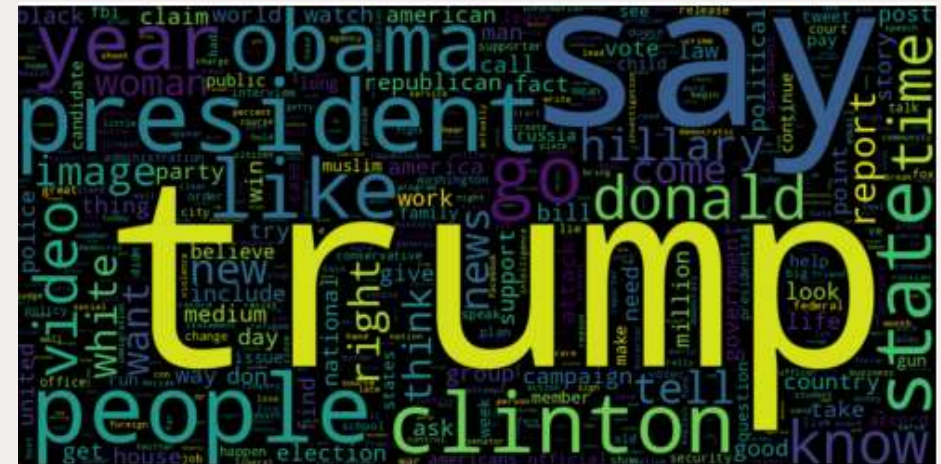
Edition: 001



Fake News WordCloud



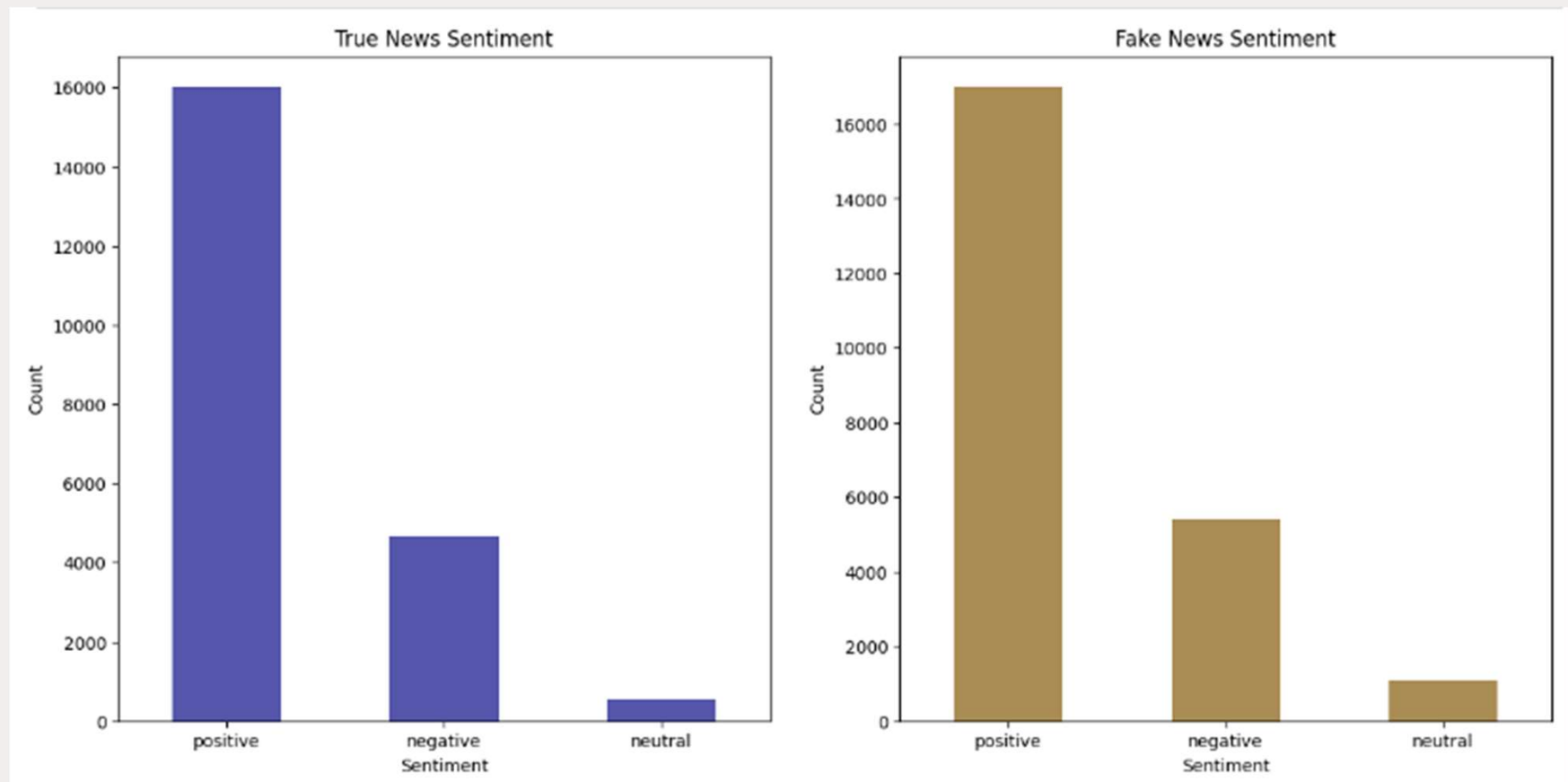
Real News WordCloud



Sentiment Analysis

Thursday 09/11/2023

Edition: 001



Machine Learning and Recurrent Neural Network Modelling

Thursday 09/11/2023

Machine Learning Models

Edition: 001

Naïve Bayes Classifier

- Based on the popular Bayes' probability theorem.
- Known for creating simple yet well performing models, especially in the fields of document classification and disease prediction.
- Ability to handle large feature spaces.

SVM Classifier

- Capability to handle High dimensionality
- Effective decision boundaries in the form of hyperplanes

Machine Learning Models

Thursday 09/11/2023

Edition: 001

Data Preparation:

- *Target:* “class” column values 0 or 1 depending on Fake or True news.
- *Features:* “text” column values with news text.
- Data split: Train vs Test => 3:1 ratio using .
- Already Tokenized, Lemmatized and stop words removed data vectorized using SKLearn’s TfidfVectorizer.

Machine Learning Models

Naïve Bayes classifier

Multinomial Naïve Bayes classifier used:

Naive = naive_bayes.MultinomialNB()

Classification Report:

	precision	recall	f1-score	support
Fake	0.59	0.89	0.71	5819
Real	0.74	0.33	0.46	5351
accuracy			0.62	11170
macro avg	0.66	0.61	0.58	11170
weighted avg	0.66	0.62	0.59	11170

SVM Classifier

Linear kernel to increase the computational efficiency of SVM used

SVM = svm.SVC(C=1.0, kernel='liner',degrees=3)

Classification Report:

	precision	recall	f1-score	support
Fake	0.65	0.97	0.78	5819
Real	0.93	0.43	0.59	5351
accuracy			0.71	11170
macro avg	0.79	0.70	0.68	11170
weighted avg	0.79	0.71	0.69	11170

Recurrent Neural Network Modelling

Thursday 09/11/2023

Edition: 001

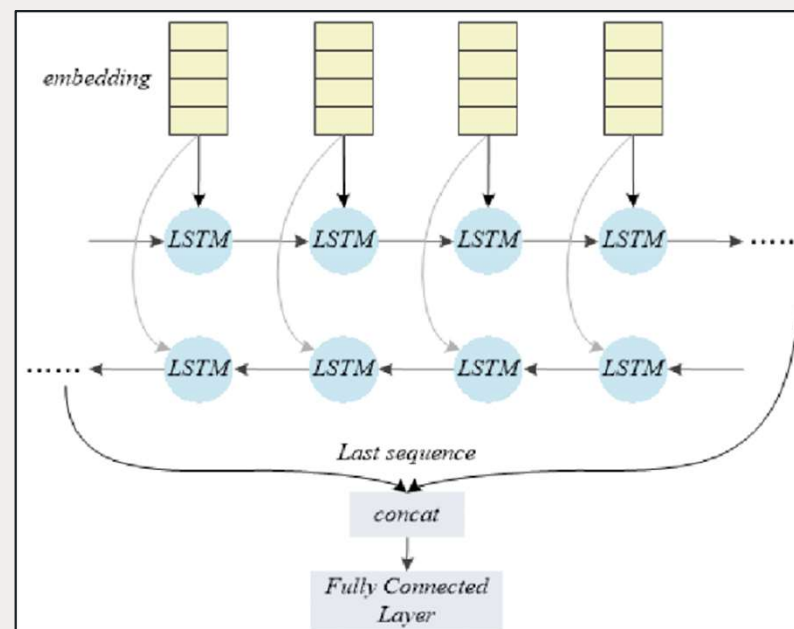
Neural Network Models

Long Short-Term Memory (LSTM)

- An advanced version of RNN designed to model chronological sequences
- Remembers patterns over time

Bidirectional LSTM(BiLSTM)

- Scans in both directions of the sequence, allowing simultaneous access to both forward and backward contexts.
- Provides additional training to the model to improve predictions



BiLSTM – Recurrent Neural Network Model

Thursday 09/11/2023

Edition: 001

Model layers and hyperparameters:

```
# Building the LSTM Model
model1 = Sequential()
model1.add(Embedding(all_words, output_dim = 256))
model1.add(Bidirectional(LSTM(128)))
model1.add(Dense(64, activation = 'relu'))
model1.add(Dense(1, activation = 'sigmoid'))
```

```
# Compiling the model
model1.compile(loss="binary_crossentropy", optimizer="adam", metrics=["accuracy"])
```

BiLSTM – Recurrent Neural Network Model

Model Training:

```
# Model training
history = model1.fit(pad_train, y_train, validation_split=0.2, epochs=2, batch_size=200, shuffle=True, verbose = 1)
```

```
Epoch 1/3
419/419 [=====] - 240s 551ms/step - loss: 0.0411 - accuracy: 0.9821 - val_loss: 0.0067 - val_accuracy: 0.9982
Epoch 2/3
419/419 [=====] - 217s 518ms/step - loss: 8.4818e-04 - accuracy: 0.9999 - val_loss: 0.0054 - val_accuracy: 0.9987
Epoch 3/3
419/419 [=====] - 203s 484ms/step - loss: 9.2755e-05 - accuracy: 1.0000 - val_loss: 0.0057 - val_accuracy: 0.9987
```

Evaluation of Model Loss and Accuracy :

```
350/350 - 39s - loss: 0.0063 - accuracy: 0.9985 - 39s/epoch - 110ms/step
Loss: 0.0062997424975037575, Accuracy: 0.9984780550003052
```

The Model is 99.8% Accurate

Front End – Streamlit

- Streamlit is an open-source app framework in Python language.
- It helps us create web apps for data science and machine learning.
- Streamlit is an all-in-one tool that encompasses web serving as well as data analysis
- It is compatible with major Python libraries such as scikit-learn, Keras, NumPy, pandas, Matplotlib etc.

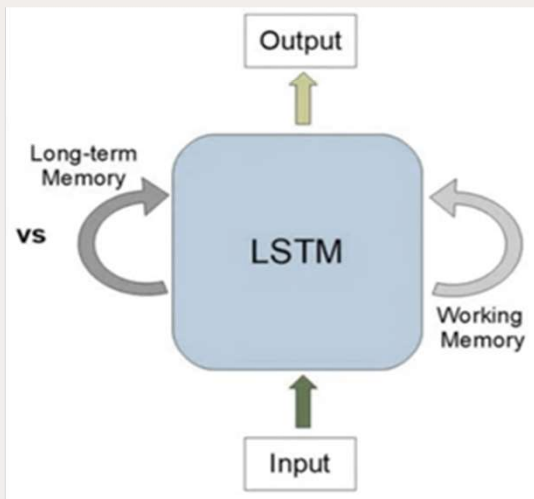
Installation : **pip install streamlit**

Execution : <Code Terminal> **streamlit run <your_script.py>**

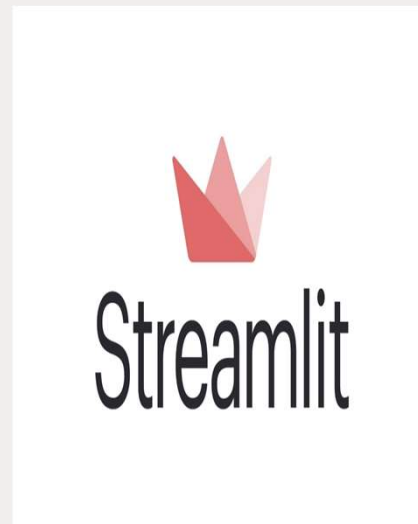
Runs an interactive application on your local computer at <http://localhost:8501>.

Development Workflow

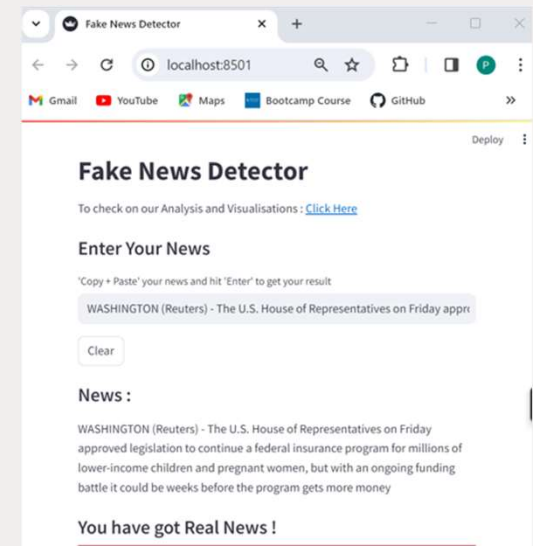
Machine Learning Model



Python Library



Interactive Web Page



THANK YOU

