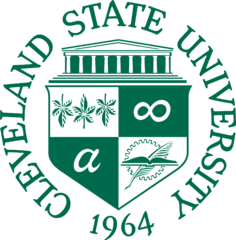
Detecting Fake vs Real News using Machine Learning



**Project Report**

Anubhuti Dayal – 2824826

Oyindoubra Timi – 2822556

Project Report

# Introduction

These days use of social media or other online platforms are getting used as digital weapons. As millions of users are there on Face book & Instagram (over 2billions), Twitter (claims 290.5 million monthly active users) spreading rumors & counterfeit news as a part of yellow journalism are very common practice these days. It is trouble-free to mess up the image of any person or organization using false allegations, mostly during elections so with help of machine learning we decided to develop algorithm to classify fake and real news.

We are currently living in a world where each single aspect is revolving around the social media, online platforms, Internet etc. Even for the news we check all the time social media. Ease of use has made social media a gigantic digital dumping yard, where everyone is posting all kind of information. And it is hard to distinguish between real and fake news. Rumors, Incorrect information, false allegations are easy to impose to rupture the image of any person, country or organization in few seconds with trillions of users. It is really significant to check and verify the truth behind it. Machine learning helps us in training the model and then tests it. My project will help to spot and segregate true /false news.

Developing machine learning algorithm to identify real vs. fake news from a data set. With this project using ‘python’ language, we developed the model using training dataset to analyze it and finally we tested it using our own developed model. As a part of Data Mining we have to go through with several steps to clean the data before using it so first of all we preprocessed it, remove all null values, drop irrelevant information from the data set, and then implement logics to recognize real vs fake information. Finally at the end we tested model with the test set. We have used python inbuilt functions, libraries to implement business logic.

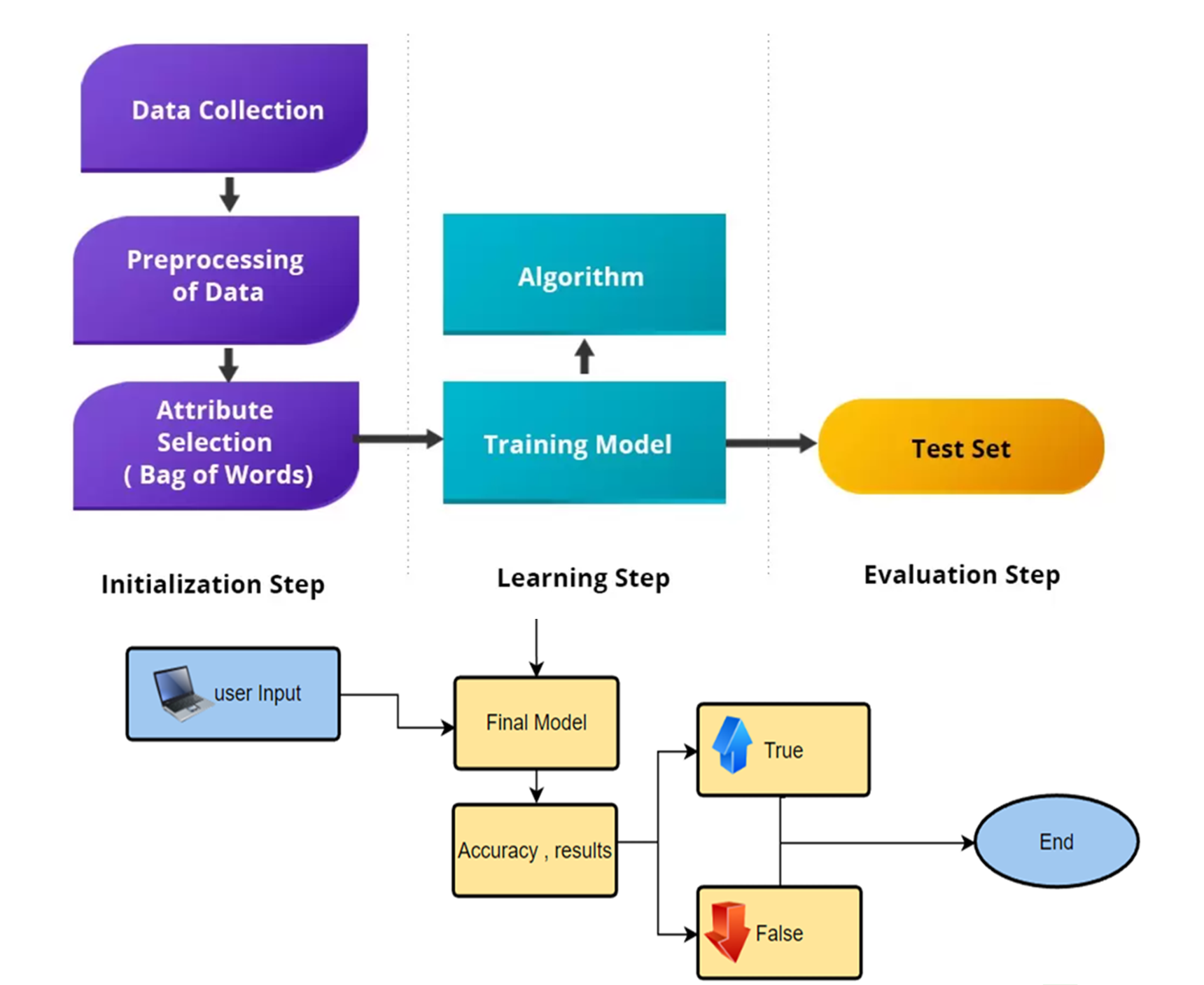
Our goal was, the final output should be able to interpret fake news correctly.

We all know that Data mining shows most impactful results if we implement & deploy it strategically to get the proper solution to a problem. Here we are planning to categorize entire data into two sets with labels (0 for fake news and 1 for real news).

**High-level Data mining steps:**

* **Data Collection** : As we mentioned in above section , first we downloaded the data
* **Environment set up**: code is written in python language, we used jupyter notebook and pycharm
* **Data Preparation**: First we cleaned the data for preprocessing, kept only relevant data to get the results.
* **Developing Model**: At this step we tried to select the appropriate modeling techniques for the downloaded dataset. There are various techniques available including Naïve-Bayes classification, clustering, Regression classifier, BERT, predictive models, classification, estimation, or a combination.
* **Test Mode**l: After generating model at previous step we tested model to measure the success, calculating accuracy, precision and verifying if it is able to detect fake news correctly or not.
* **Results**: We have tuned the model , changed the data set , merged multiple data set and ran again and again to get promising results accuracy. Finally summarized the results created report and presentation for the demonstration of the project and final submission.

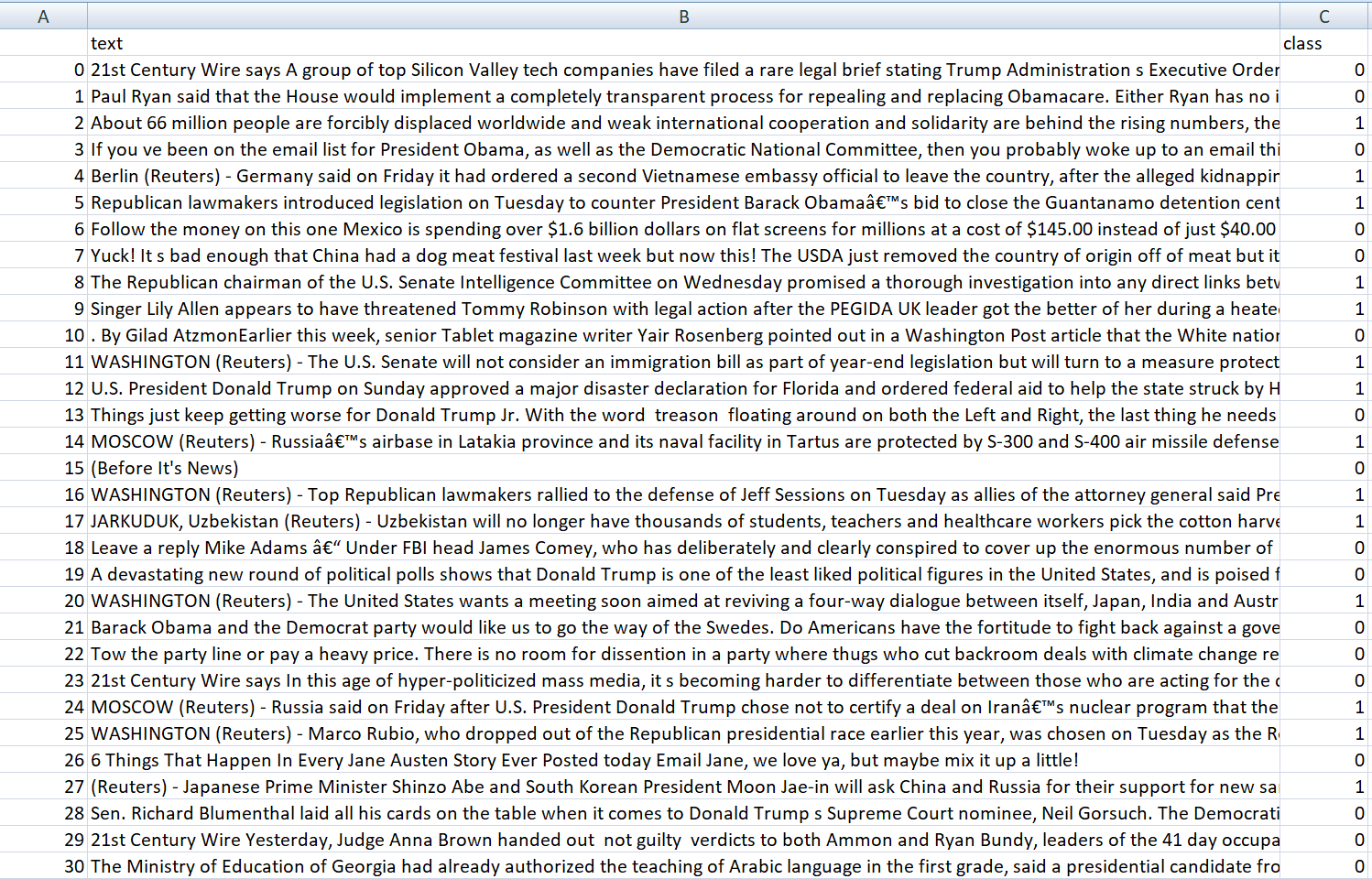
**Flow chart diagram for the steps involved in generating model**



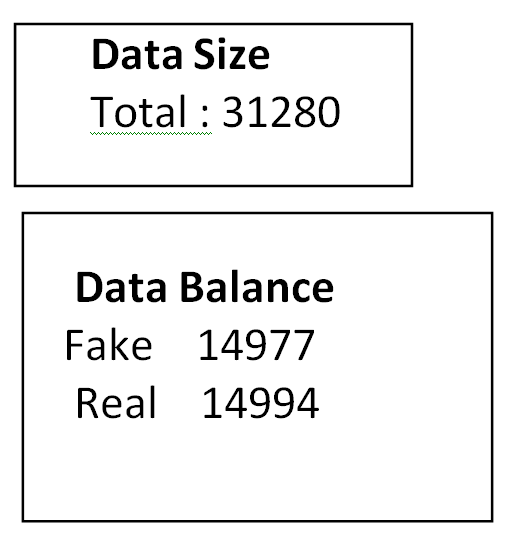
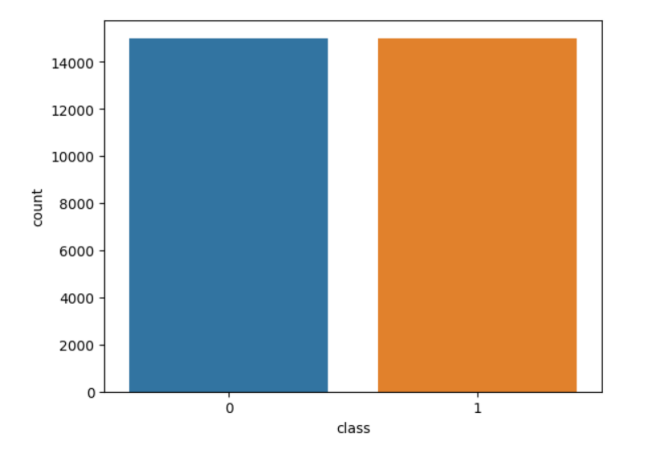
Data Contents, Selected Feature Description

***Data Contents***

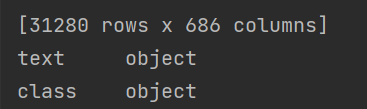
Researched multiple sites to get the correct sample going to use Kaggle.com data for the final interpretation. <https://www.kaggle.com/datasets/>

******

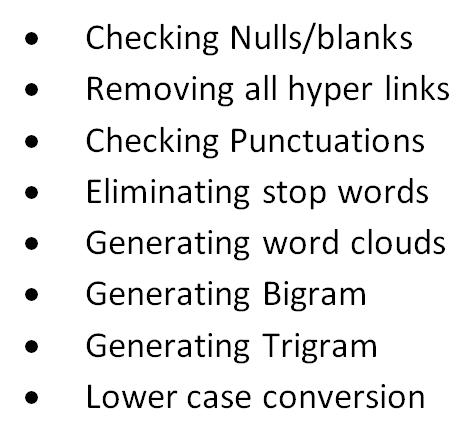
Merged multiple data sets from Kaggel.

******

**Selected Features:** text and class



Data Preprocessing Steps



**Python code**



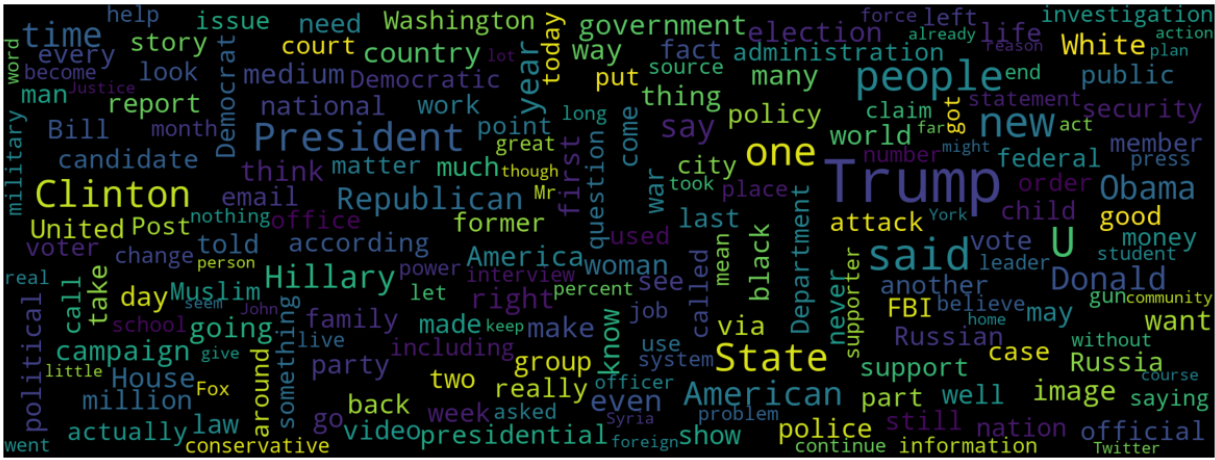
Generating Word clouds –

For True and Fake news

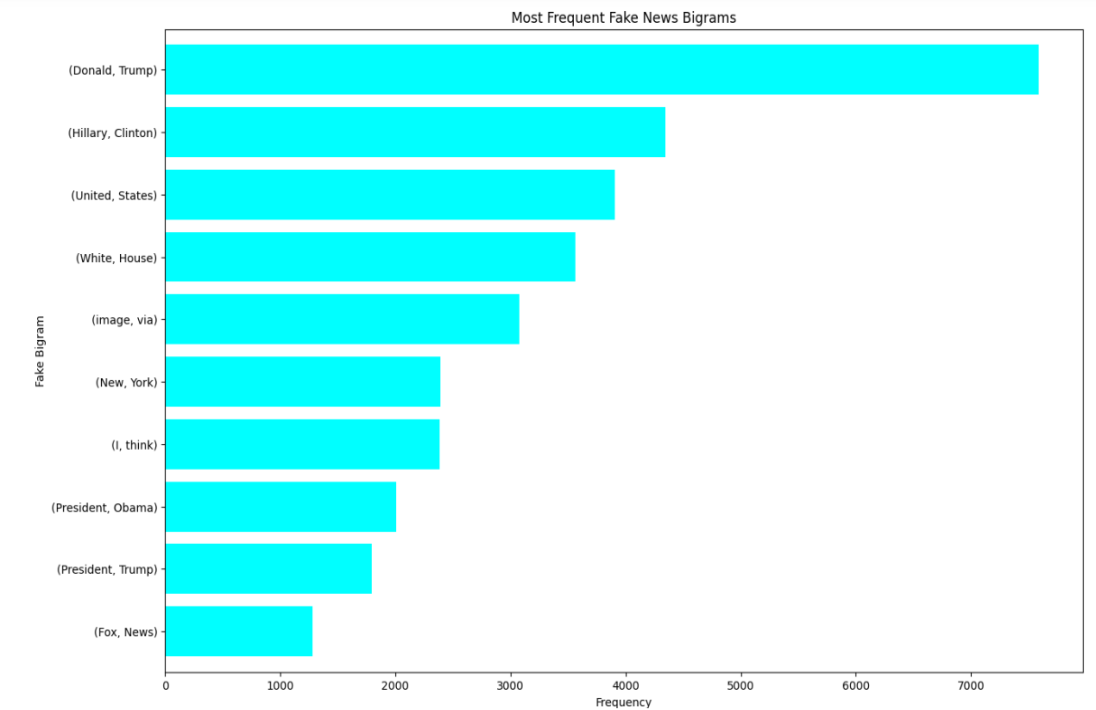
True Case

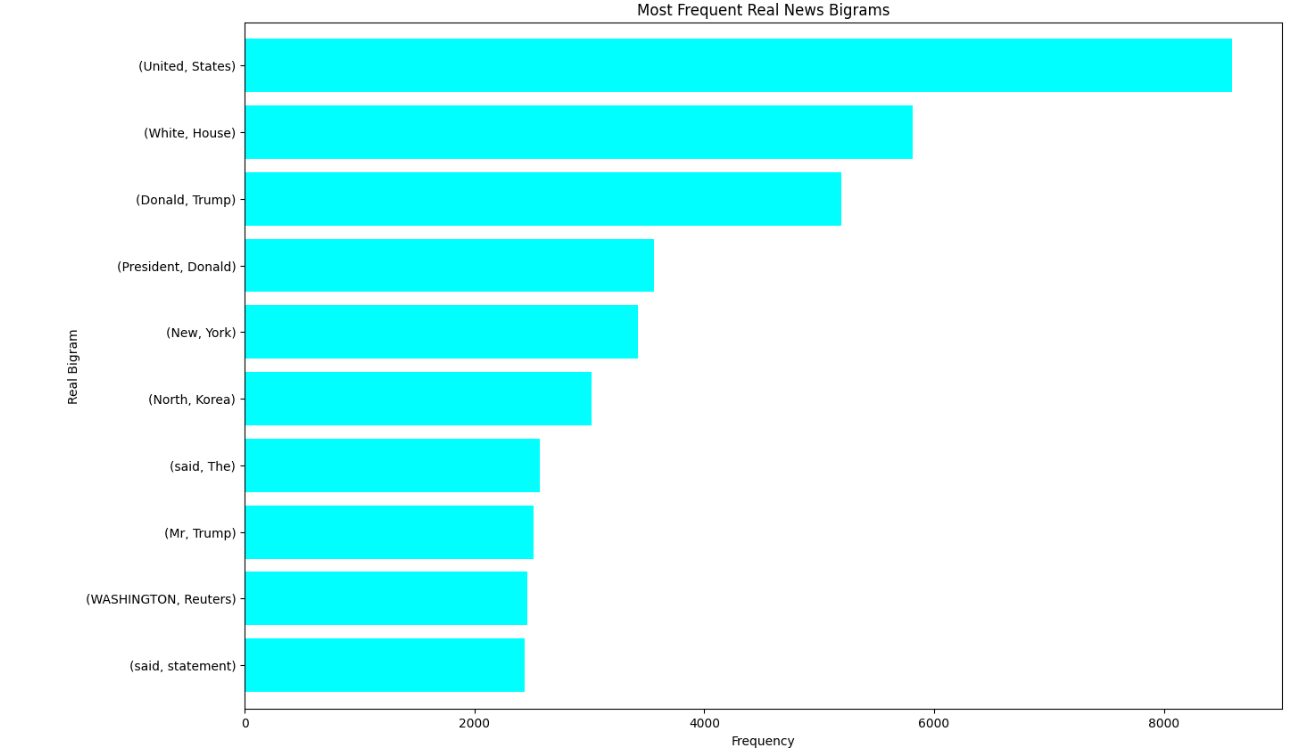


Fake News



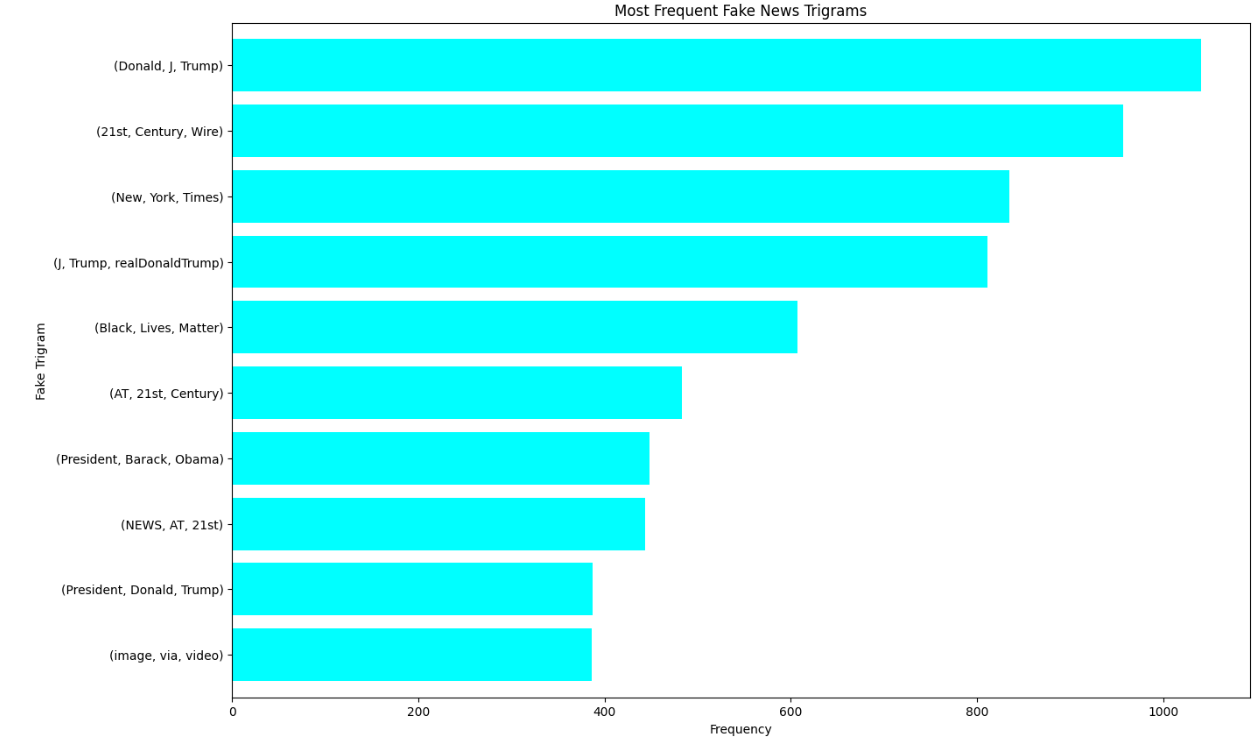
**Generating Bigram**:  A **bigram** is a sequence of two adjacent elements from a string , like “please turn”, “turn your”, or ”your homework”. We ran the code to see the counts in graph for our data set.

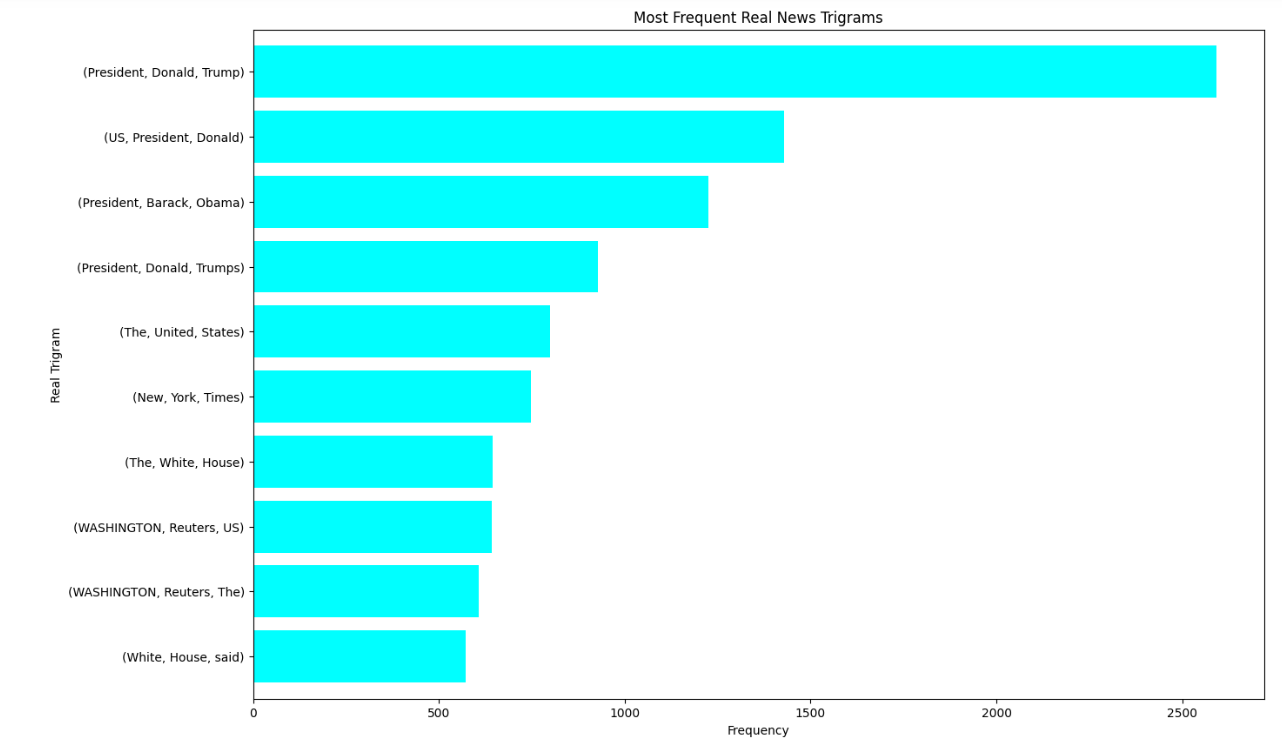
**Fake**

**Real**

**Trigrams :** A trigram is a sequence of three adjacent elements from a string , like “please turn your”,

or “turn your homework” , Please find results below for the Trigram for our data set

**Fake**

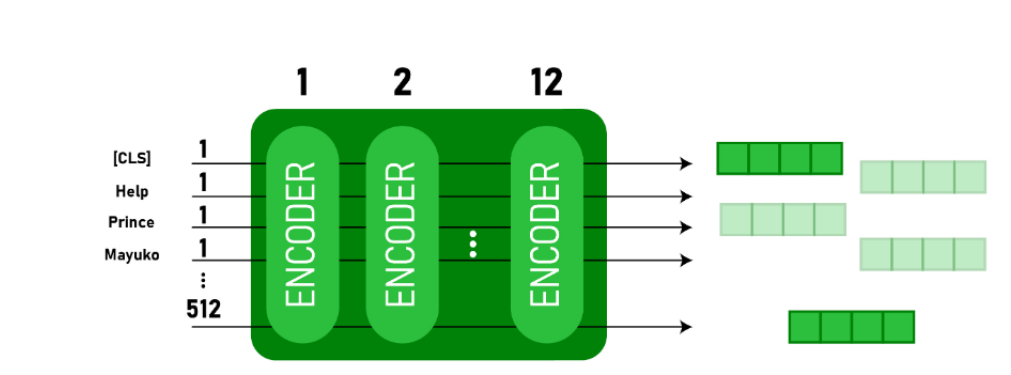
**Real**

Model Generation

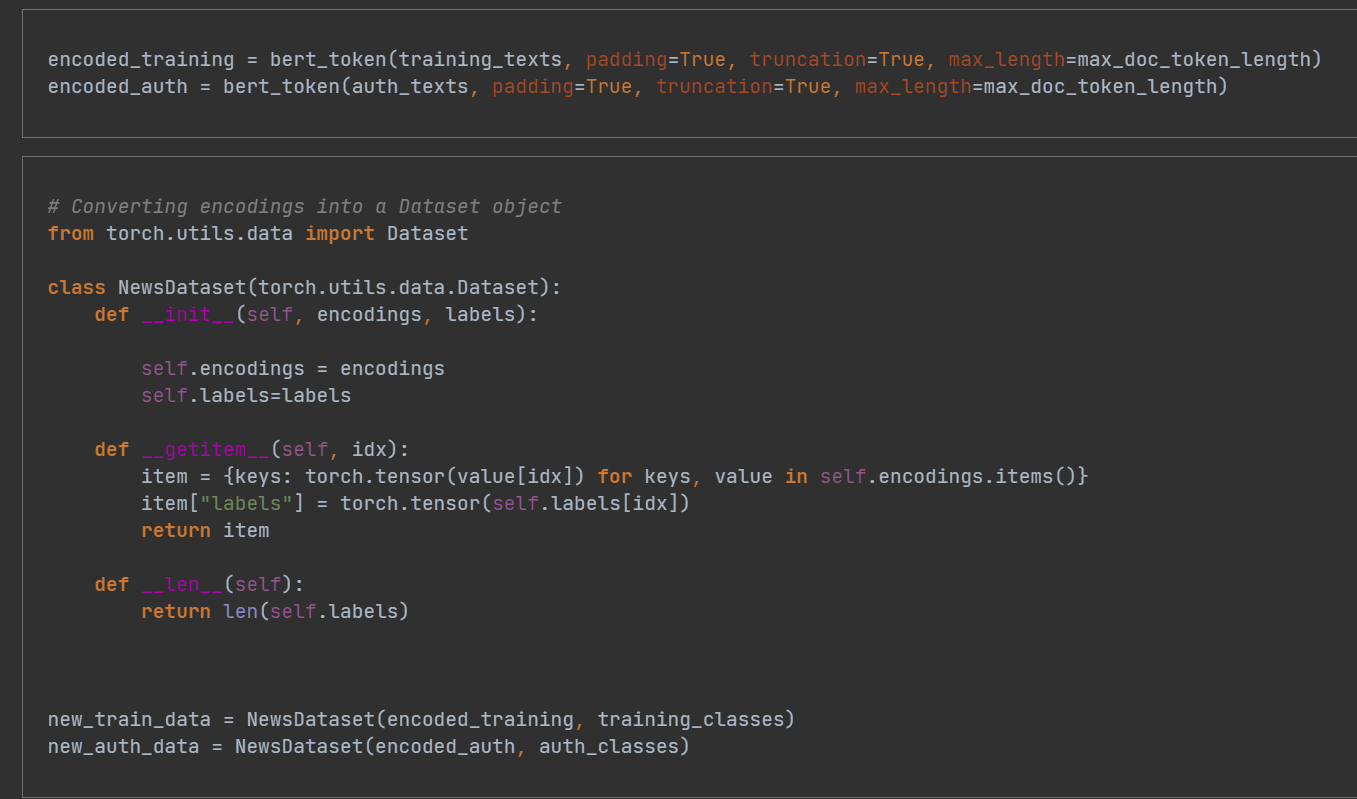
We have tried, Naïve-Bayes , BERT and Neural Network model , finally we selected BERT model. BERT model gave us best results over other 2 models.

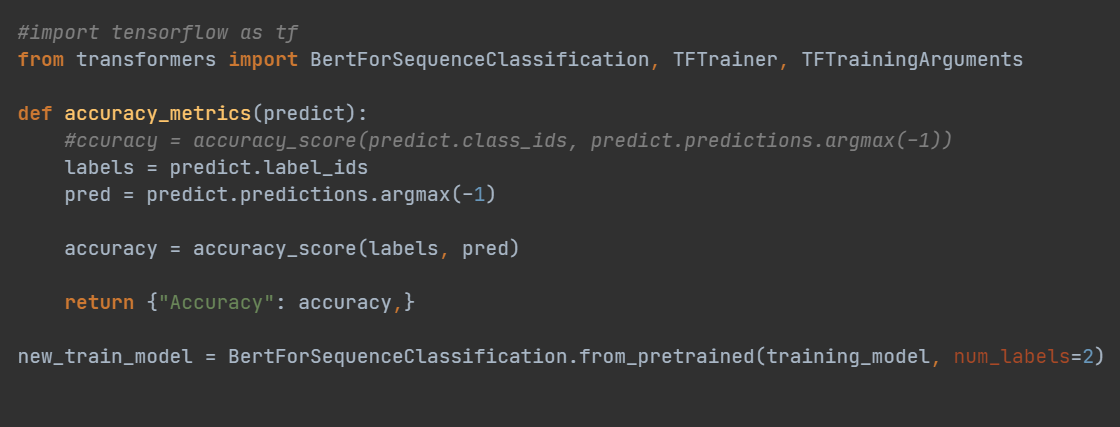
BERT, which stands for **Bidirectional Encoder Representations from Transformers1**, is based on Transformers.

A deep learning model in which **every output** element is connected to every **input element**, and the **weightings** between them are dynamically calculated based upon their **connection**.



Python code





Problems Encountered

* **Using Training Datasets with Insufficient Data was giving ambiguous results**

**Resolution:** Used different data set, merged multiple data set with sufficient counts & proper balance between fake and Real to Train the model with better accuracy.

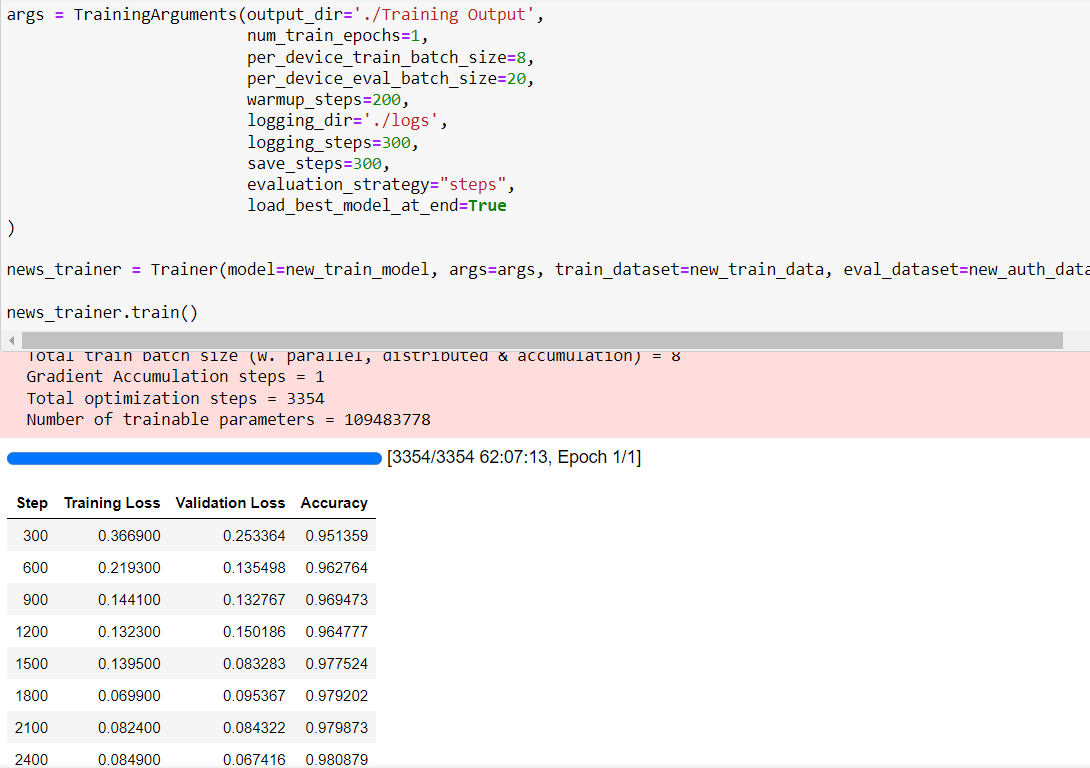
* **Takes lot of time to train the model and see the results and then again modify and see next iteration results.**
* **Also tried with Neural network – where accuracy was stuck to 50% only.**

Training Set and Test Set

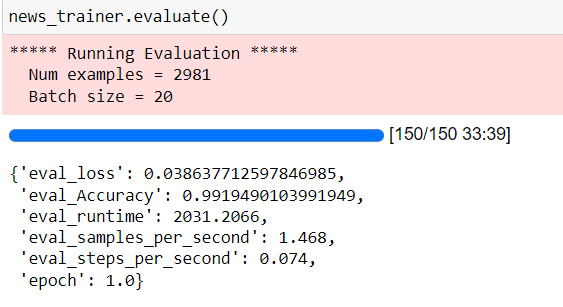
**We divided data in 10 % for testing & 90% training the model**

****

Results

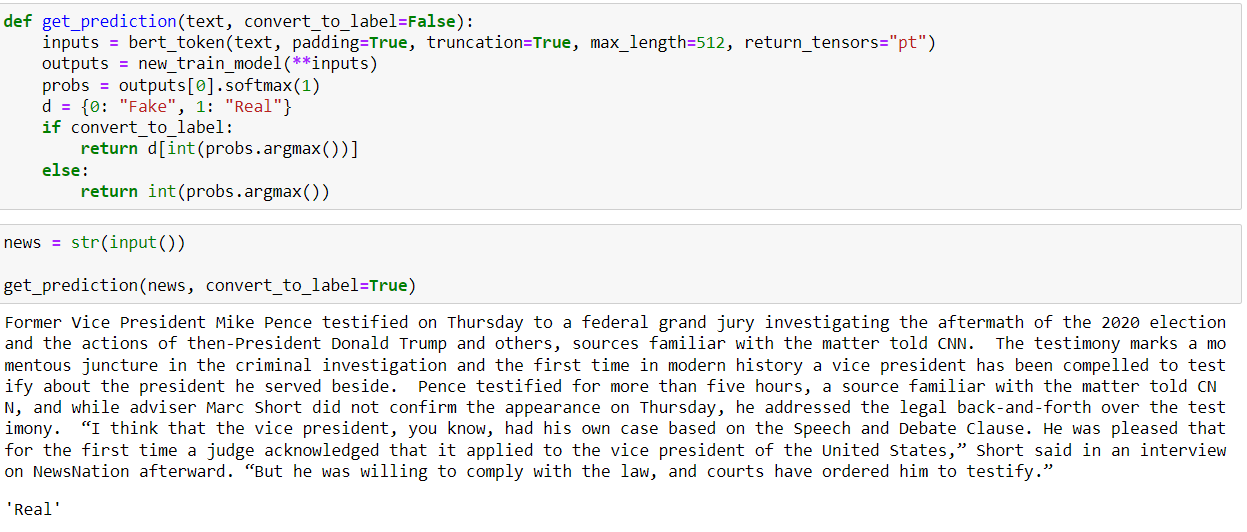


**98%** Accuracy



Testing & Conclusion

We tested by passing one news and it gave correct results !



**Conclusion:**

While social media has facilitated the timely delivery of various types of information around the world, a consequence is that news is emerging at an unprecedented rate, making it increasingly difficult to fact-check. A series of incidents over recent years have demonstrated the significant damage fake news can cause to society. Therefore, how to automate the process and accurately identify fake news before it is widespread has become an urgent challenge for research.

Based on our experiment and project we can conclude our model is working fine, just that data set size and with lower computer speed, code execution takes lot of time, where we can improve the code/algorithms. Also there are various other techniques and features we can count to optimize the performance which is sometimes the key to stop propagating false news.

**References**

[1] <https://towardsai.net/p/l/fake-news-detection-using-bert-model-python>

[2] <https://en.wikipedia.org/wiki/Fake_news>

[3] <https://cits.ucsb.edu/fake-news/what-is-fake-news>