DATA SCIENCE IMMERSIVE

Fake News Detection Using Machine Learning

A presentation by Rohazeanti

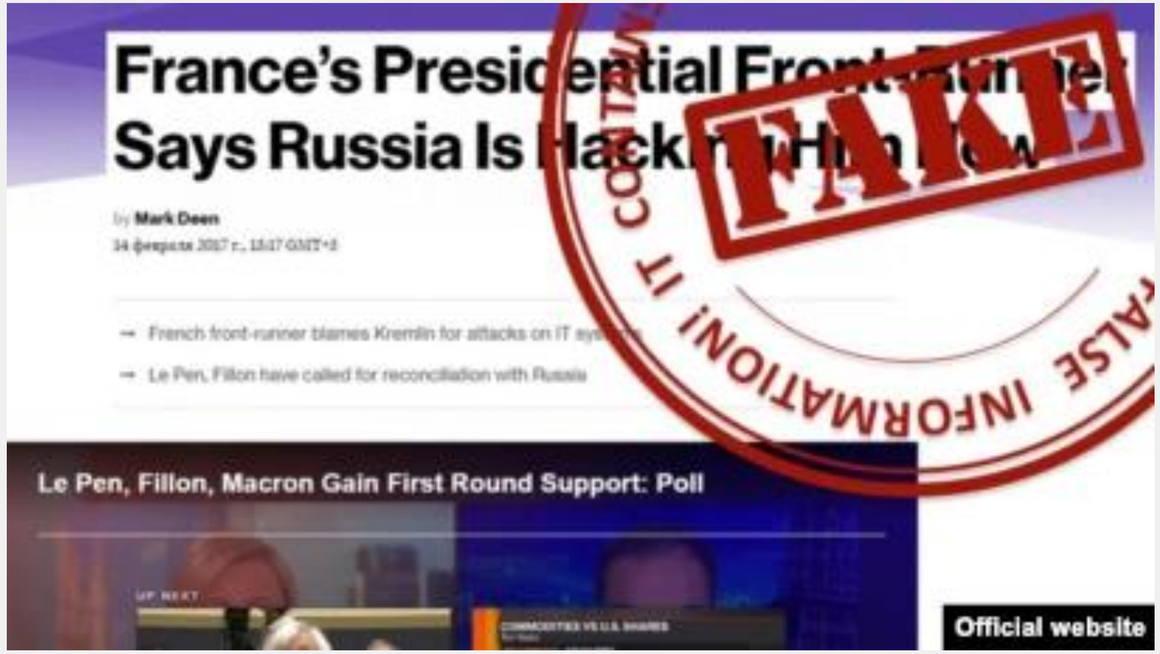
AGENDA

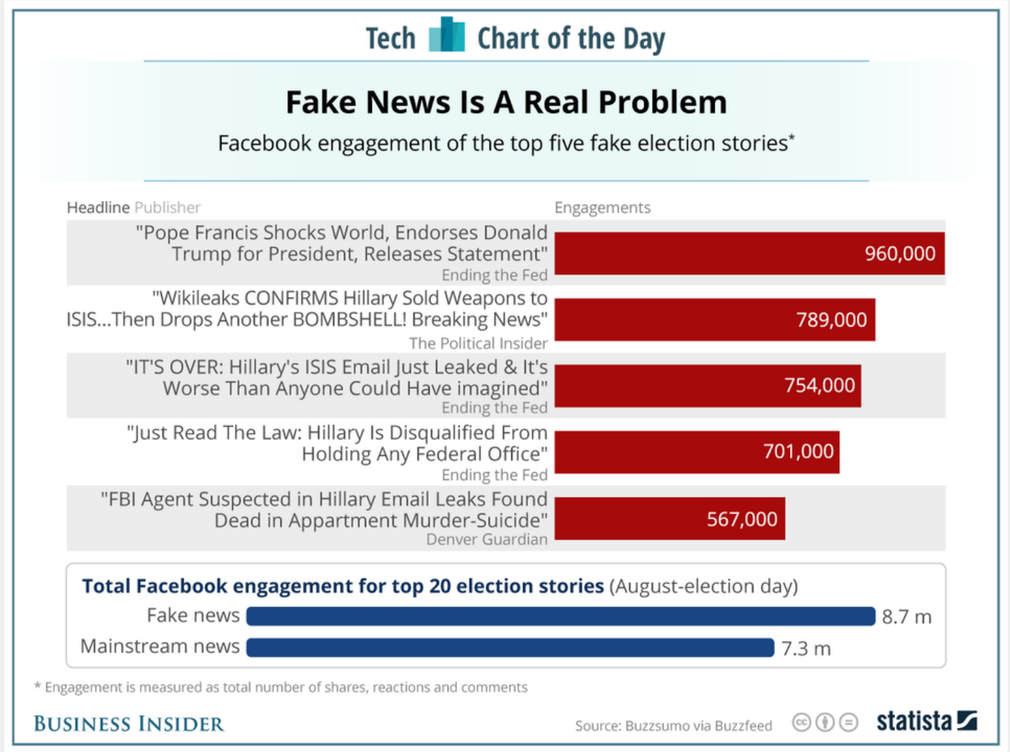
Topics Outline

- Problem Definiton
- Effects of Fake News
- Problem Statement
- ML Process
- Pre-process
- EDA
- Vectorization
- Model Result
- Limitations
- Challenges
- Improvement and Future Works
- Conclusion

Problem Definition

Fake news is information, stories or hoaxes made to misinform or mislead audiences, These narratives are made to influence people's views, push political agenda, create embarrassment, and get gains out of online publishers





Problem Definition

The proliferation of news on social media and the Internet is deceiving people to an extent which needs to be stopped.

01

Influence opinions

Alter organic processes of public opinion formation
Shape behaviours from voting to taking a stance on public issues

02

Affects governance

Threat to democracy and to efficient governance

03

Promote toxic narratives

Spreads doubts, and confusion, increases social polarisation, affecting democratic decision-making

Problem Statement

As a data scientist engaged by the Government, I am tasked to create a fake news detection technique, using Machine Learning model, for members of public to determine the authenticity of a given news.

Objective: Deploy a highly accurate model that classifies a given news article as either fake or true, allowing consumers to check for news reliability through their browsers efficiently.

Target Users

Fake News Detection System is built for all

Members of Public

These are the people who want to gain access to reliable information.

Journalists

These are the people who have the responsibility to inform people of the truthth.

PROCESS

Machine Learning Process

1 - Download Data

ISOT Fake News dataset

- Compilation of fake and truthful articles
- 44,898 rows
- 4 columns (title, text, subject, date)

2 - Pre-process

Prepare data for model building

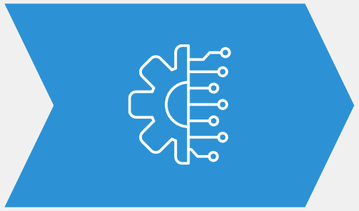
- Natural Language Toolkit(NLTK) library

3 -Exploratory

Understand our dataset

- matplotlib library
- visualisations
- wordcloud

4 - Feature Enginerring



Feature extraction

- Reduce overfitting risk
- Text Vectorization
  - CountVectorizer
  - TF-IDF

-

5 - Model

Traditional Machine Learning Models:

- Logistic Regression
- Random Forest
- DecisionTree Classifier
- MultinomialNB
- KNeighbors Classifier
- AdaBoost Classifier
- Gradient Boosting Classifier

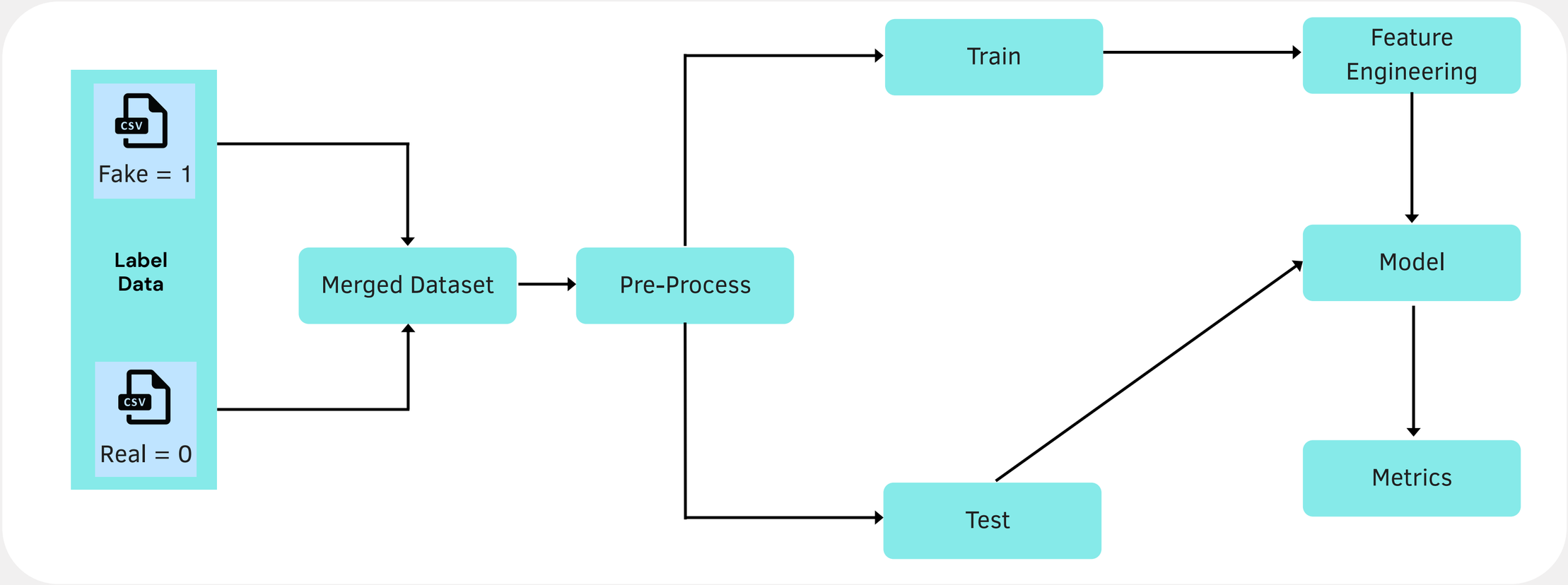
5 - Evaluation

Compare metrics:

- Confusion matrix
- F1, Recall, Precision

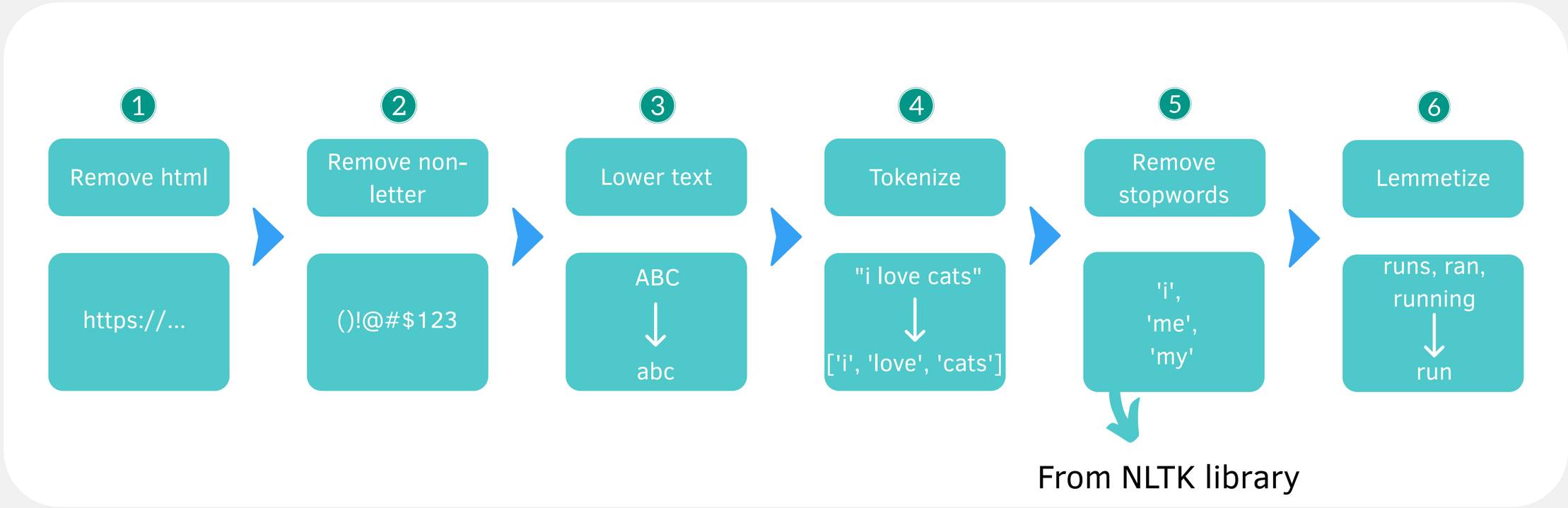
Process Flow

Step-by-step process



Pre-Processing Text Data

- Get rid of unhelpful parts of data, or *noise*
- Reduce size/dimensionality of text corpus

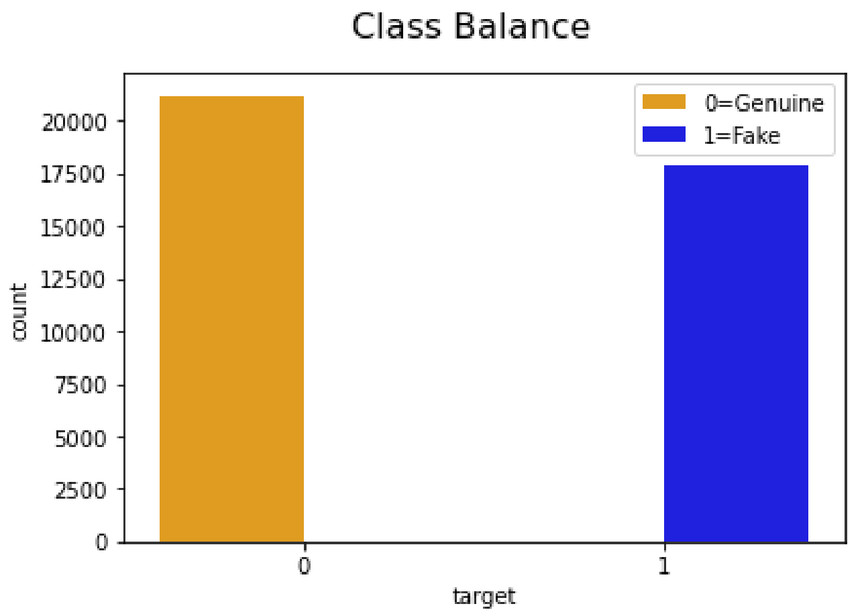


EDA: Class Balance

Is the dataset balanced?

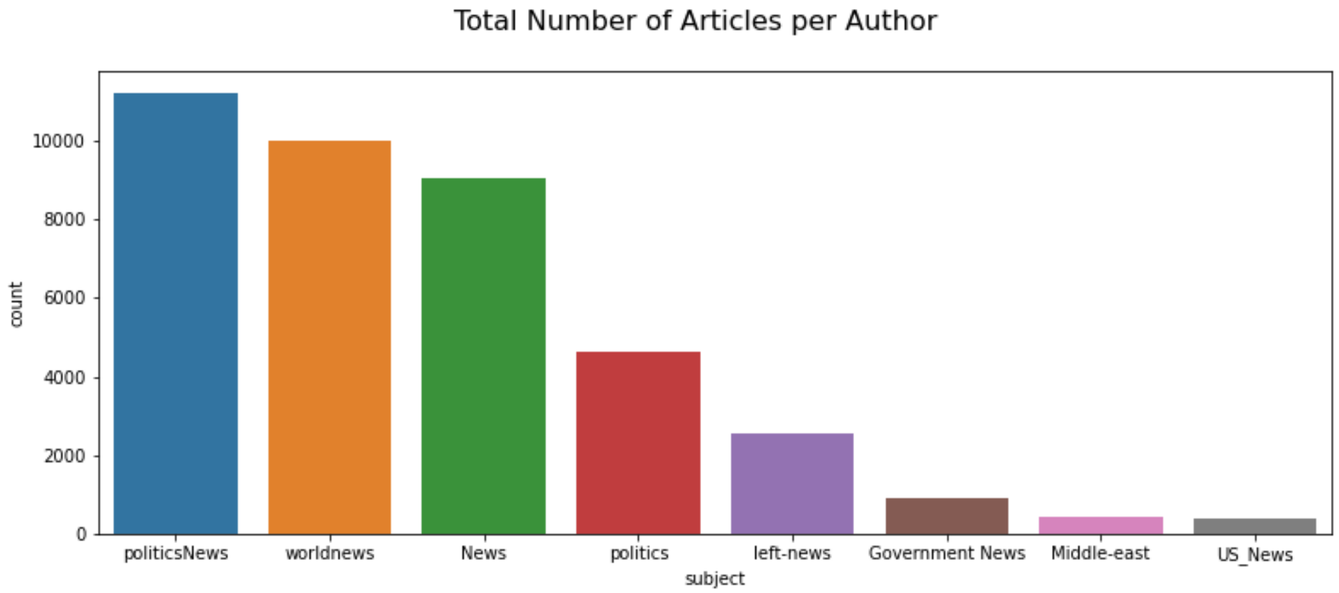
```
0    0.542139
1    0.457861
```

Slightly imbalanced but **acceptable**



EDA: Num of Articles

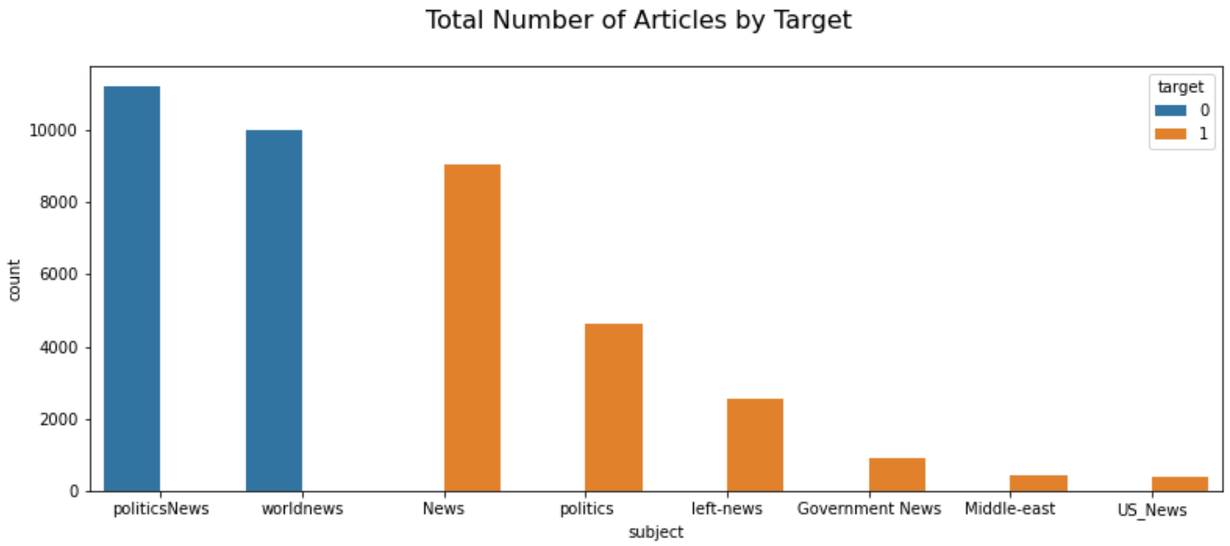
Who are the authors?



- 8 authors
- Majority of news comes from politicsNews, worldnews, News

EDA: Number of Articles

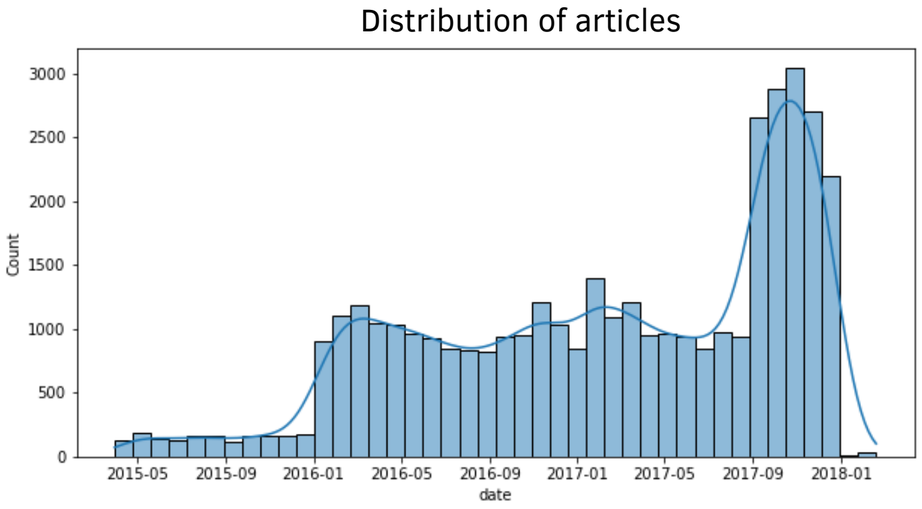
Who are the contributors of fake and real news?

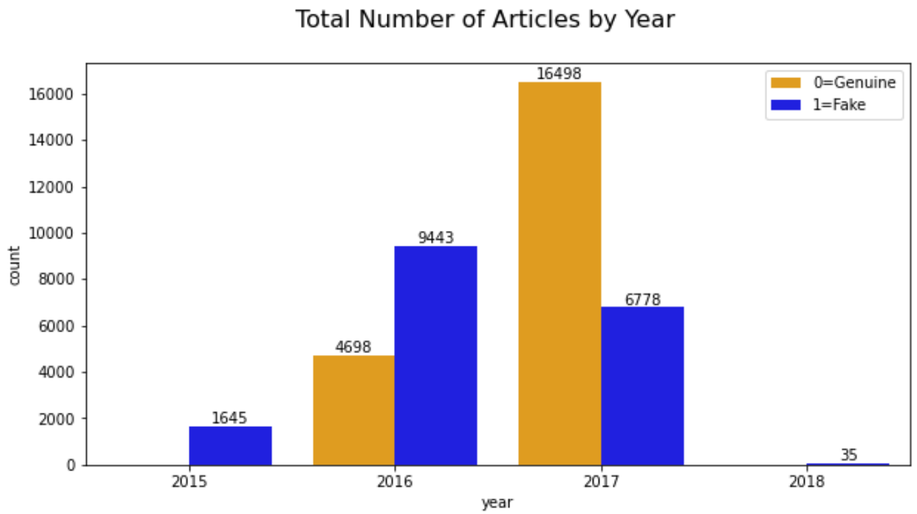


- Two authors publish genuine articles
- Six authors publish fake articles

EDA: Period of data

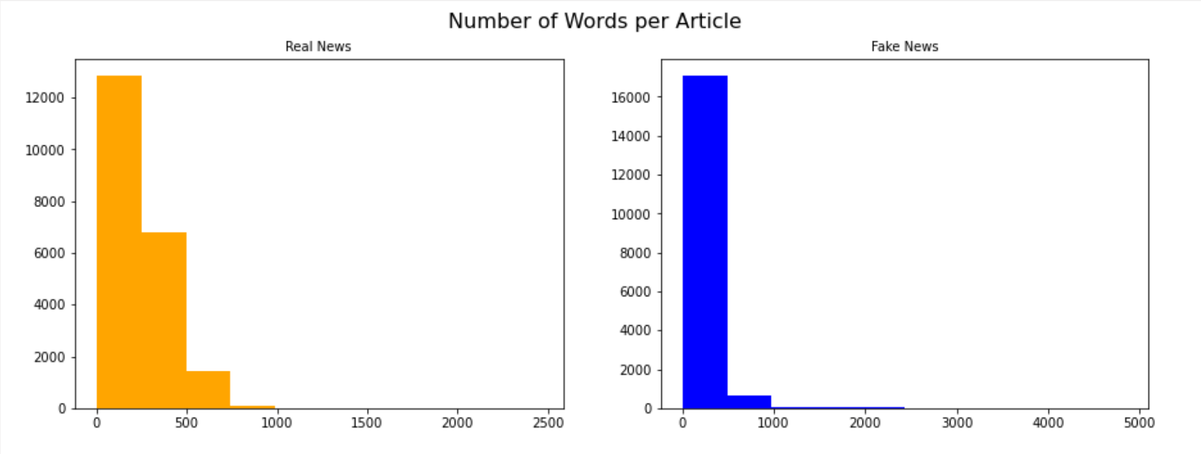
- 4 years, 2015 - 2018
- Highly imbalanced between September 2017 and Dec 2017
- US presidential election in 2017
- More efforts in publishing real news during election in 2017 (3 times more than prev. year)





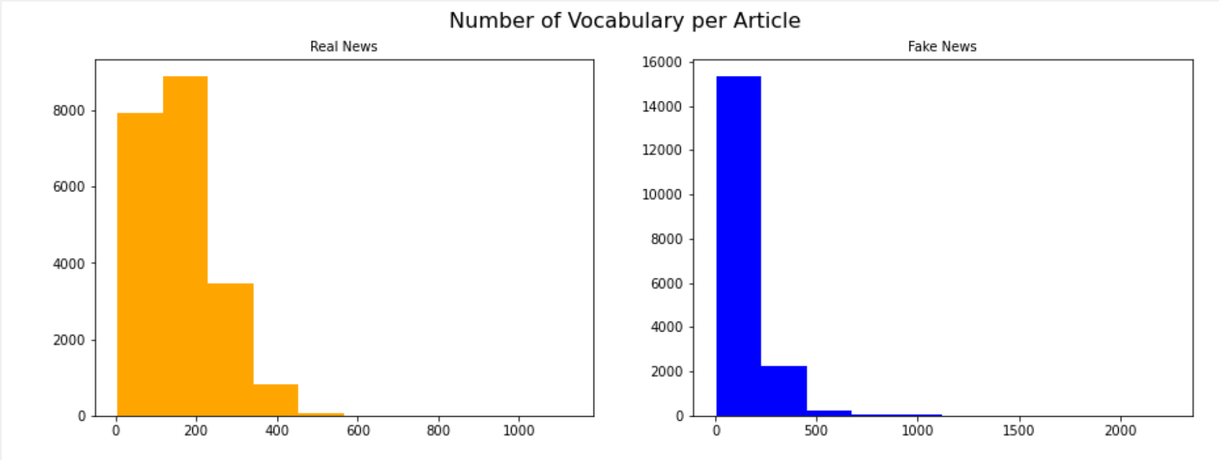
EDA: Text Characterics

Who has more number of words and vocabulary?



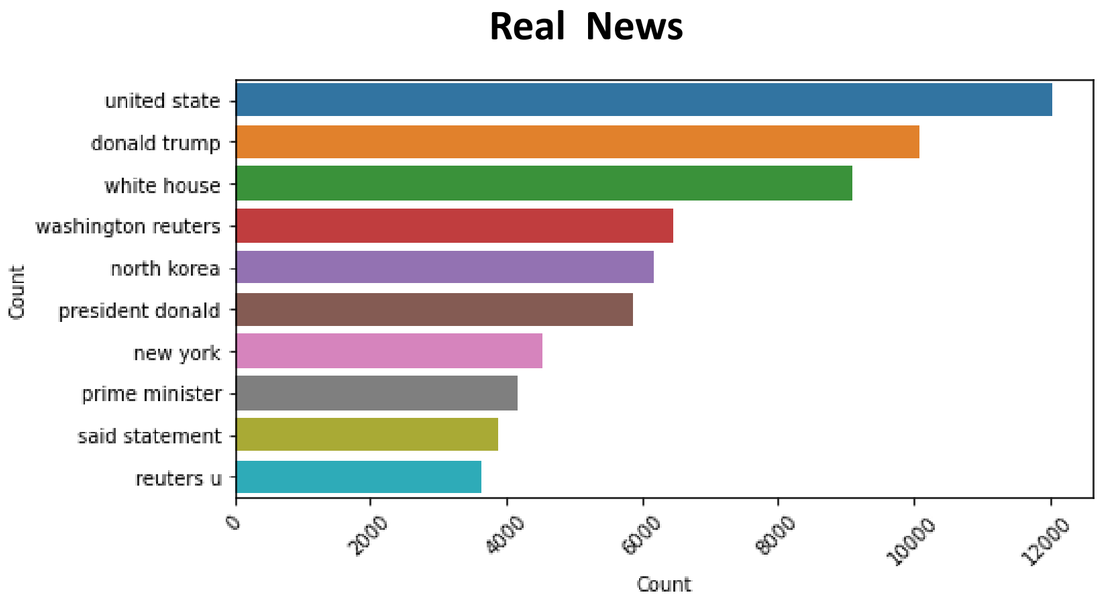

```
----Real News----
Mean:  1737.249952821287  Mode:  0    384
dtype: int64  Median:  1619.0
----Fake News----
Mean:  1727.6313613764594  Mode:  0    64
dtype: int64  Median:  1528.0
```

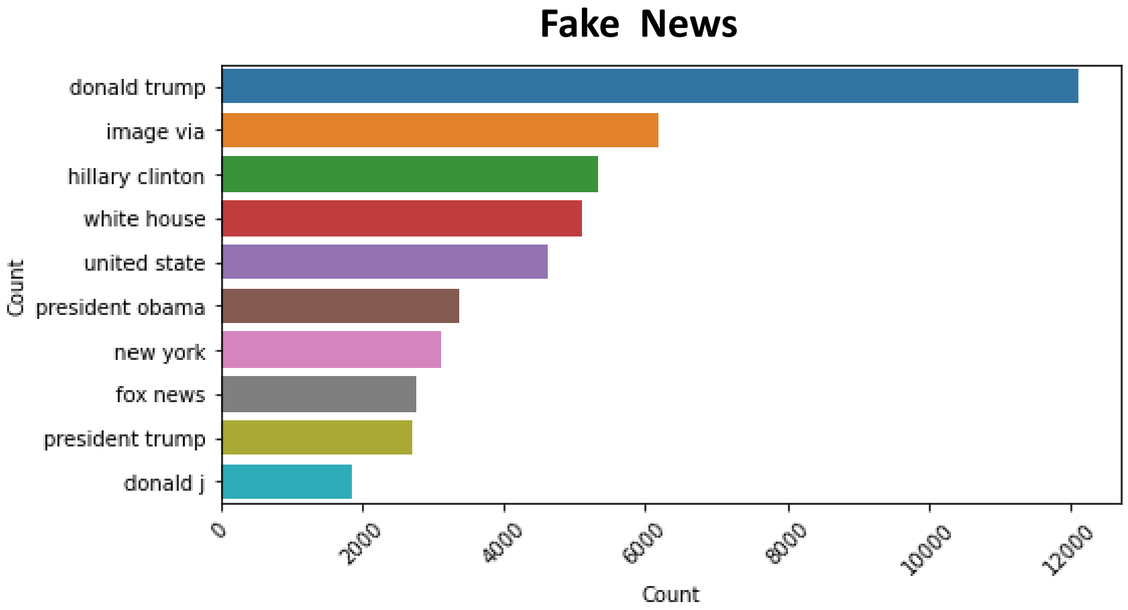
The average number of words and vocabulary in each dataset does not differ that much




```
----Real News----
Mean:  155.94942441970184  Mode:  0    43
dtype: int64  Median:  149.0
----Fake News----
Mean:  161.56589017373332  Mode:  0    149
dtype: int64  Median:  151.0
```

EDA: Tokenized N-gram



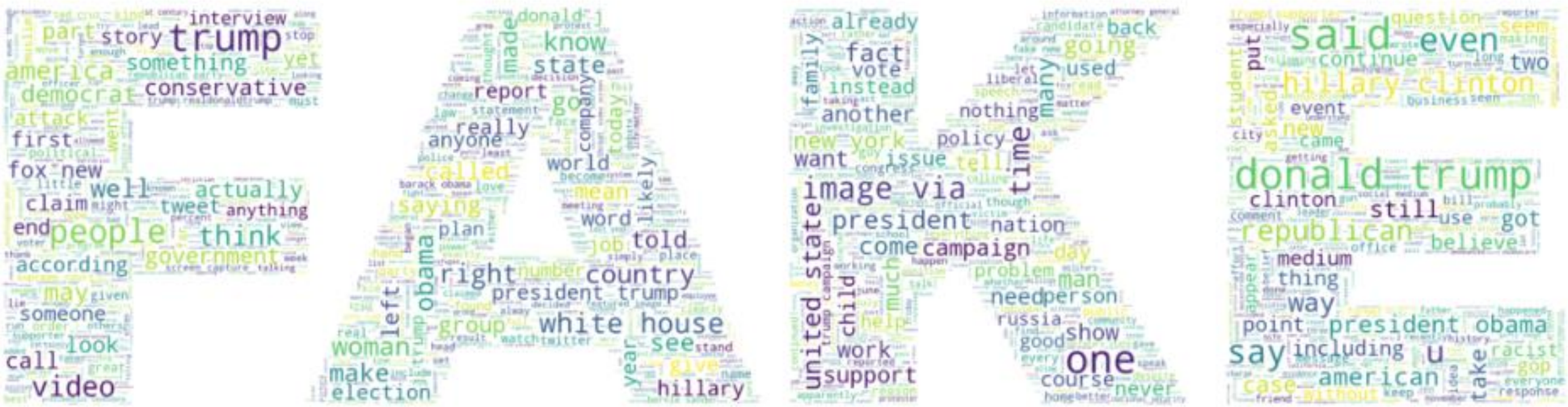


Wordcloud

Collection of words from **real news** dataset

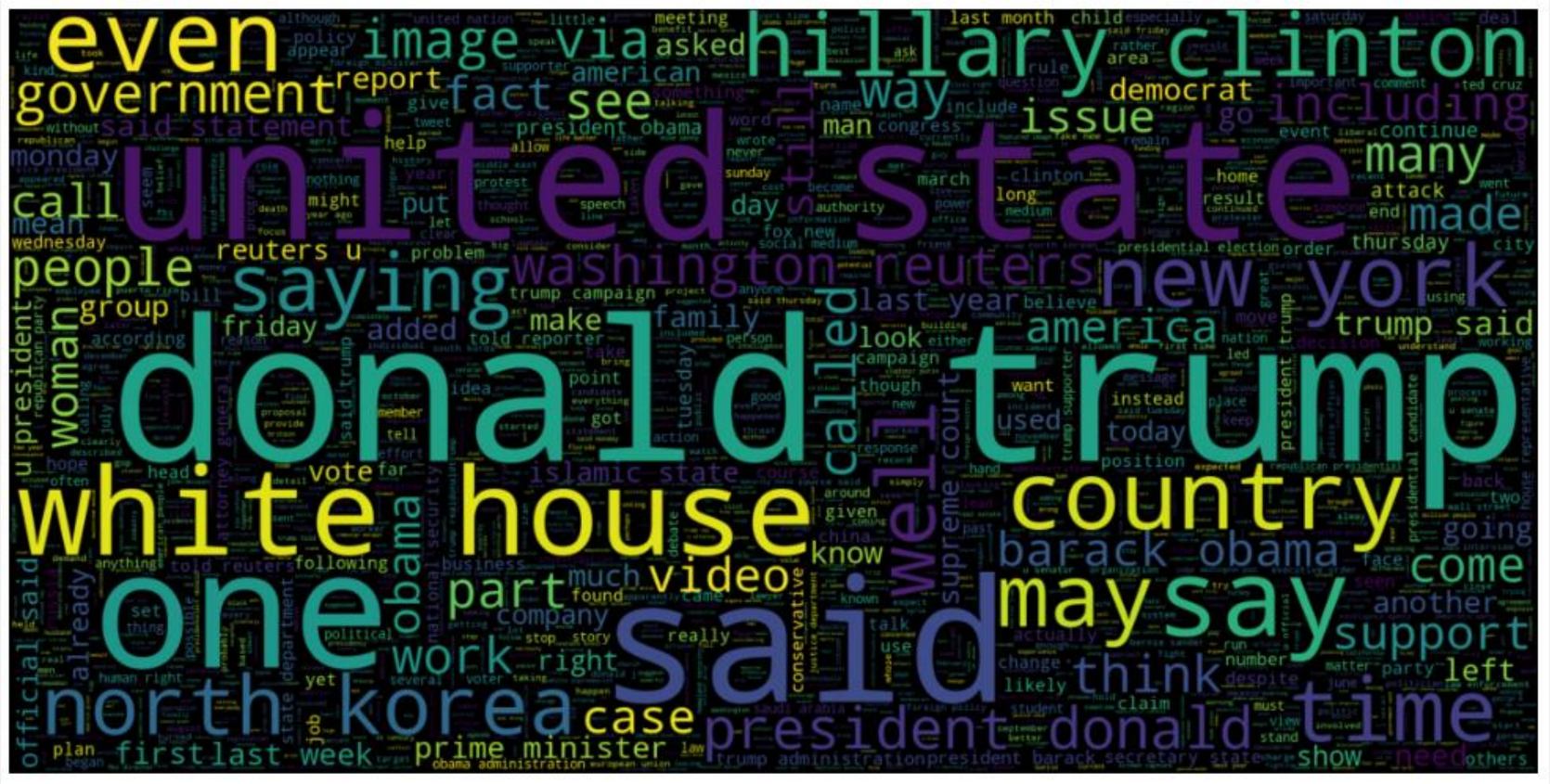
Wordcloud

Collection of words from **fake news** dataset



Wordcloud

Collection of words from **combined dataset**



Feature Engineering

- TF-IDF(term frequency-inverse document frequency) Vectorizer
- Scikit Learn
- Find frequent words in X_train
- Remove irrelevant and high-occurring words

1. Term Frequency

$$\mathrm{TF}(t,d) = \frac{Number\ of\ times\ t\ occurs\ in\ a\ document\ 'd'}{Total\ word\ count\ of\ document\ 'd'}$$

Raw count of instances a word appears in a document

2. Inverse Document Frequency

$$\mathrm{IDF}(t) = \log_e\left(\frac{Total\ number\ of\ documents}{Number\ of\ documents\ with\ term\ t\ in\ it}\right)$$

How common or rare a word is in the entire document set. The closer it is to 0, the more common a word is

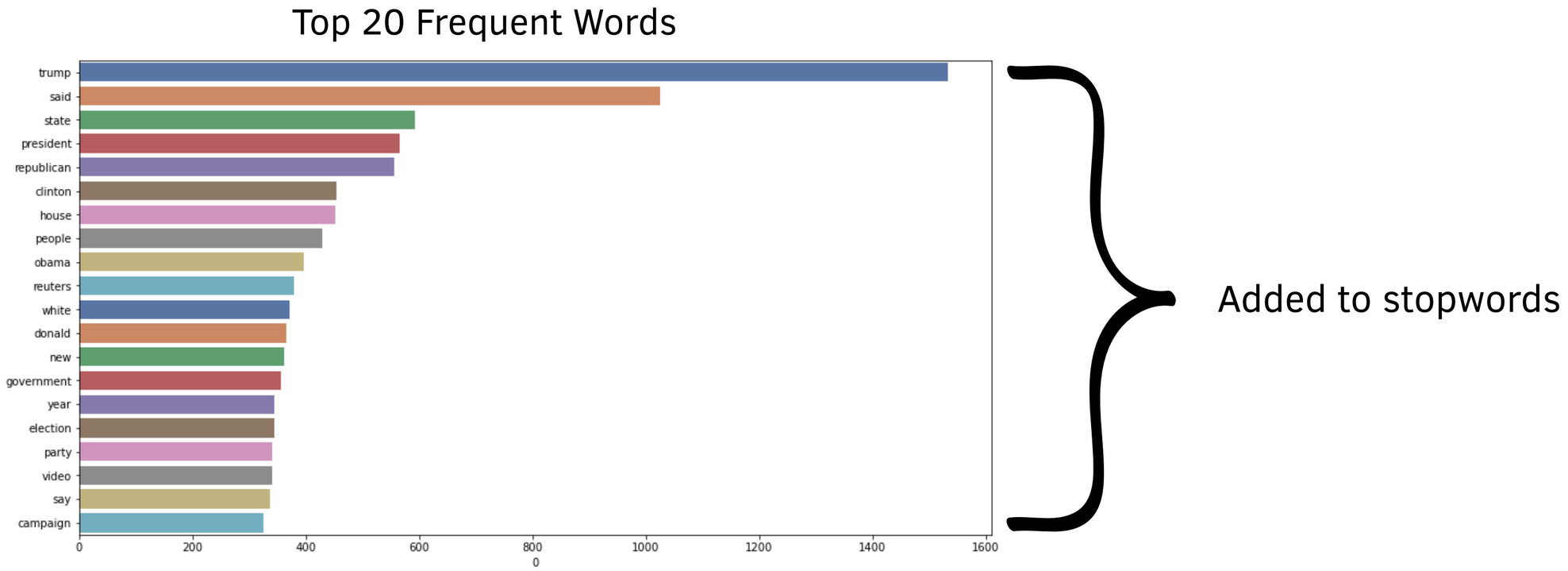
3. Multiply

$$\mathrm{TF\text{-}IDF}(t,d) = \mathrm{TF}(t,d) * \mathrm{IDF}(t)$$

The higher the score, the more relevant that word is in that particular document

Feature Engineering

- TF-IDF Vectorizer (Scikit Learn)
- Find frequent words in X_train
- Remove irrelevant and high-occurring words



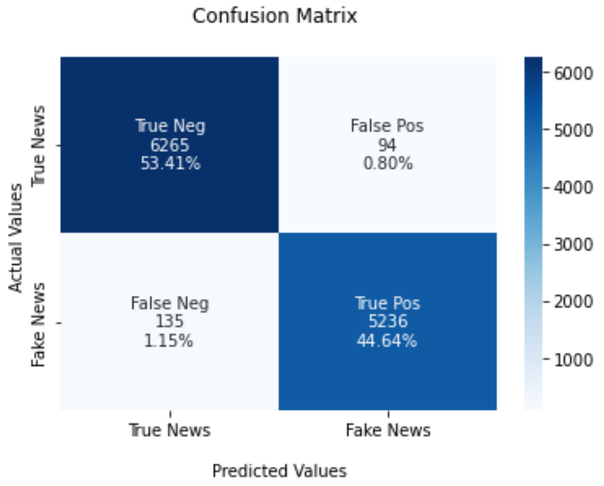
Metrics

- Precision, Recall, F1, ROC AUC
- Confusion Matrix

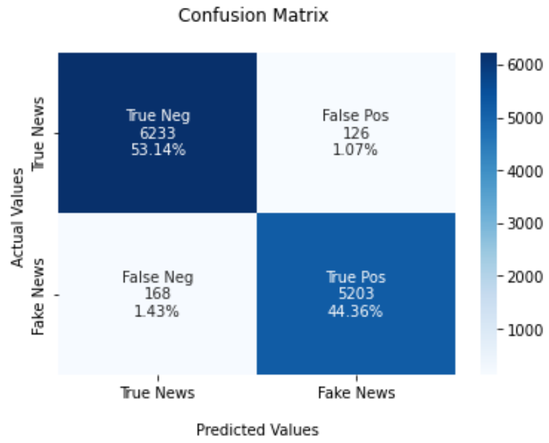
LogisticRegression with TFIDF
- Efficient based on execution time

	Train score	Test score	Generalisation	Accuracy	Precision	Recall	Specificity	F1	ROC AUC	Execution Time
LogisticRegression_TfidfVectorizer	0.982	0.975	0.713	0.975	0.976	0.969	0.98	0.972	0.9962	00:38:14
RandomForestClassifier_TfidfVectorizer	1.0	0.965	3.5	0.965	0.974	0.949	0.978	0.961	0.9943	00:45:47
DecisionTreeClassifier_TfidfVectorizer	0.973	0.913	6.166	0.913	0.903	0.907	0.917	0.905	0.9184	00:46:01
MultinomialNB_TfidfVectorizer	0.924	0.923	0.108	0.923	0.912	0.922	0.925	0.917	0.9765	00:10:56
KNeighborsClassifier_TfidfVectorizer	1.0	0.846	15.4	0.846	0.785	0.913	0.788	0.844	0.9275	01:00:05
AdaBoostClassifier_TfidfVectorizer	0.99	0.98	1.01	0.98	0.982	0.975	0.985	0.978	0.9973	02:15:50
GradientBoostingClassifier_TfidfVectorizer	0.985	0.964	2.132	0.964	0.964	0.958	0.969	0.961	0.9917	01:11:21

AdaBoostClassifier



LogisticRegression



True Positives: Number of fake news that are correctly identified as fake news

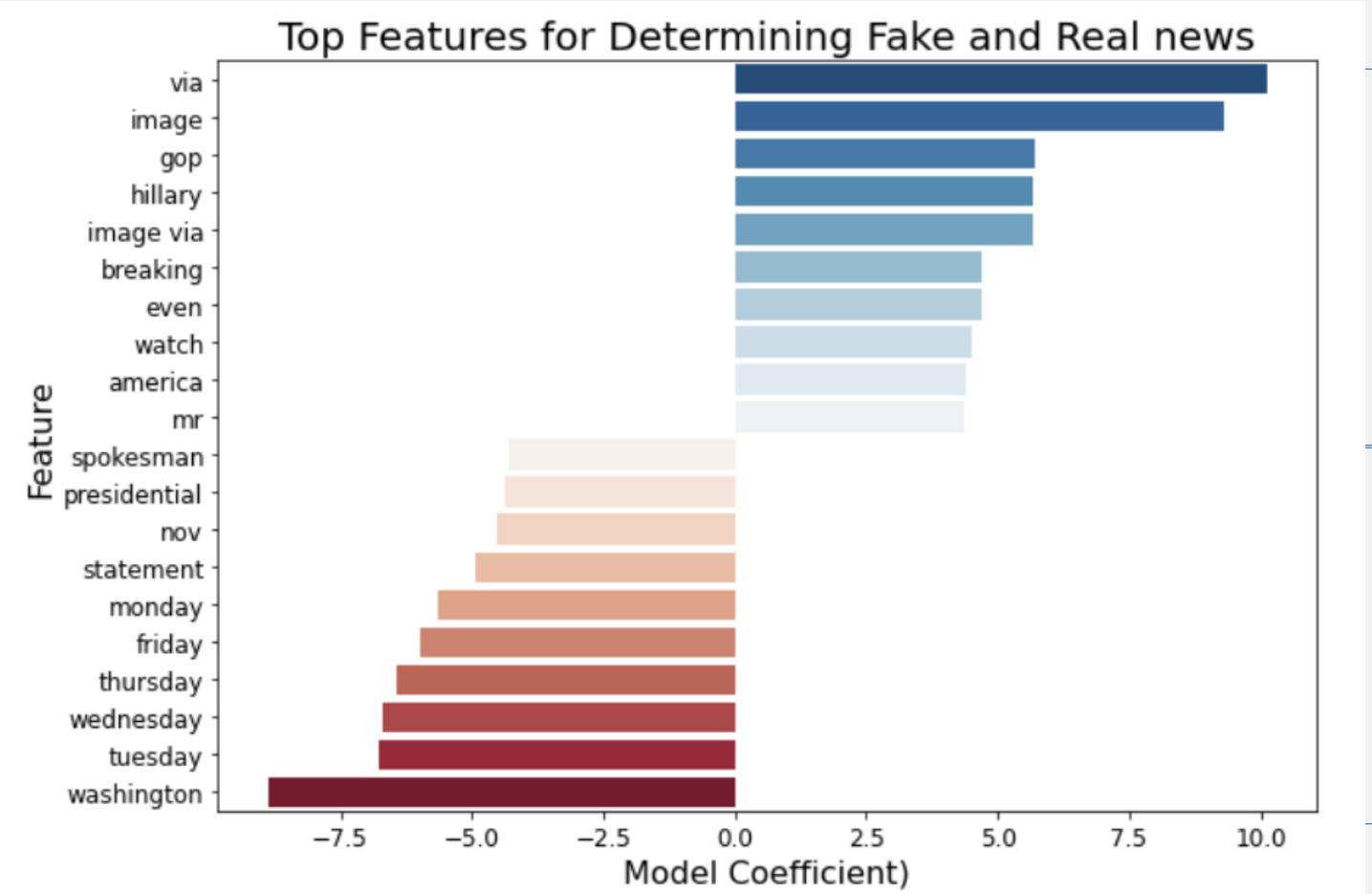
False Positive: Number of real news that are identified as fake news

True Negative: Number of real news that are identified as real news

False Negative: Number of fake news that are identified as real news

Top Features

- Top 10 features that determine fake and real news



FAKE

REAL

Deployment of ML Model

Fake News Detection System

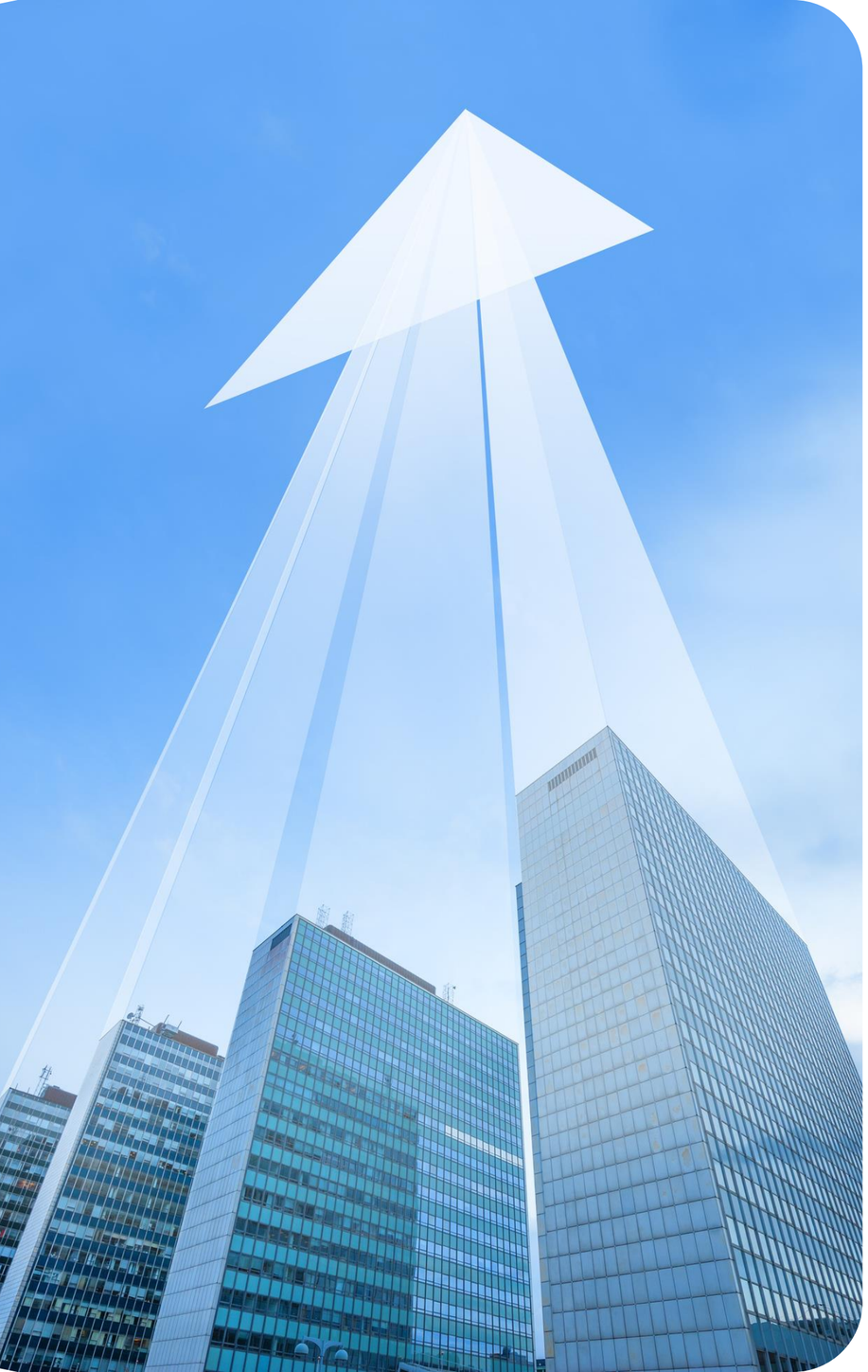
Enter Any News:

Web based deployment using **Streamlit**

Process:

- Load saved feature engineering and best model
- User input news article
- Make a predictoin

DEMO



Limitations

Period of data

- Data captured only 2015 - 2018

Real-time requirement

- Dataset is not current

Limited topics

- Political & World news topics
- Unable to detect Covid-related fake news

False Positive/False Negative

- What was fake may be real now

Fact-checkers biasness

- Gender, race, prejudices

Improvements

HYPERPARAMETER

Tweak hyperparameter

DEEP LEARNING

Apply deep learning and compare performance

MORE DATA

Train the model using more and current data

Future Works

SENTIMENTS

Analyse sentiments of fake news

VIDEO/IMAGES

Detect fake news from video and images

TIME-SERIES

Fake news propagation pattern

Challenges

Time Consuming

Laborious and time-consuming process

Limited Resource

GPUs, memory, data storage

Conclusion

- Hardcopy newspapers that were earlier preferred are now being substituted with social media and Internet.
- To detect fake news manually by cross-checking multiple sources can be daunting, time consuming and may cause further confusion.
- With a fake news detection system, it speeds up the process of determining whether a piece of news is fake or real. However, it does not stop there.
- Government are now taking measures ensuring its citizens consume legitimate news. (POFMA, Education)
- Even though there are other models that performed better, I stick with Logistic Regression simply due to its **efficiency**.
- Collecting the data once isn't going to cut it given how quickly information spreads in today's connected world and the number of articles being churned out.
- Compare performance with Deep Learning

Stay Connected

HTTPS://WWW.LINKEDIN.COM/IN/ROHAZEANTI/

ROHAZEANTI@GMAIL.COM

HTTPS://GITHUB.COM/ROHAZEANTI