PESUIOProject Fake News Detection using Machine Learning

NAVNEET NAYAK
PES2202300060

MAHIKA PATNEY
PES2202300223

D. SRI SANJANA
PES2202300226

Table of contents

01 What is the Goal of this project?

The goal of this project is to build a machine learning model that is able to accurately classify news as real or fake when given a news article. To do this we need to follow a series of steps. Starting with Data PreProcessing, where we clean the data and make it ready to be fit into a machine learning model. The next step is to train our machine learning model. We tested our data on many different Machine Learning models to see which model produced the most accurate results.

02

Data PreProcessing

What is our Dataset?

We have two datasets, one containing real news articles and one containing fake news articles. Both datasets contain the title, text and date of the published articles and also the subject of the article

	title	text	subject	date
0	Donald Trump Sends Out Embarrassing New Year'...	Donald Trump just couldn t wish all Americans ...	News	December 31, 2017
1	Drunk Bragging Trump Staffer Started Russian ...	House Intelligence Committee Chairman Devin Nu...	News	December 31, 2017
2	Sheriff David Clarke Becomes An Internet Joke...	On Friday, it was revealed that former Milwauk...	News	December 30, 2017
3	Trump Is So Obsessed He Even Has Obama's Name...	On Christmas day, Donald Trump announced that ...	News	December 29, 2017
4	Pope Francis Just Called Out Donald Trump Dur...	Pope Francis used his annual Christmas Day mes...	News	December 25, 2017
...	...	...	...	...
23476	McPain: John McCain Furious That Iran Treated ...	21st Century Wire says As 21WIRE reported earl...	Middle-east	January 16, 2016
23477	JUSTICE? Yahoo Settles E-mail Privacy Class-ac...	21st Century Wire says It s a familiar theme. ...	Middle-east	January 16, 2016
23478	Sunnistan: US and Allied 'Safe Zone' Plan to T...	Patrick Henningsen 21st Century WireRemember ...	Middle-east	January 15, 2016
23479	How to Blow $700 Million: Al Jazeera America F...	21st Century Wire says Al Jazeera America will...	Middle-east	January 14, 2016
23480	10 U.S. Navy Sailors Held by Iranian Military ...	21st Century Wire says As 21WIRE predicted in ...	Middle-east	January 12, 2016

23481 rows × 4 columns

FAKE NEWS

	title	text	subject	date
0	As U.S. budget fight looms, Republicans flip t...	WASHINGTON (Reuters) - The head of a conservat...	politicsNews	December 31, 2017
1	U.S. military to accept transgender recruits o...	WASHINGTON (Reuters) - Transgender people will...	politicsNews	December 29, 2017
2	Senior U.S. Republican senator: 'Let Mr. Muell...	WASHINGTON (Reuters) - The special counsel inv...	politicsNews	December 31, 2017
3	FBI Russia probe helped by Australian diplomat...	WASHINGTON (Reuters) - Trump campaign adviser ...	politicsNews	December 30, 2017
4	Trump wants Postal Service to charge 'much mor...	SEATTLE/WASHINGTON (Reuters) - President Donal...	politicsNews	December 29, 2017
...	...	...	...	...
21412	'Fully committed' NATO backs new U.S. approach...	BRUSSELS (Reuters) - NATO allies on Tuesday we...	worldnews	August 22, 2017
21413	LexisNexis withdrew two products from Chinese ...	LONDON (Reuters) - LexisNexis, a provider of l...	worldnews	August 22, 2017
21414	Minsk cultural hub becomes haven from authorities	MINSK (Reuters) - In the shadow of disused Sov...	worldnews	August 22, 2017
21415	Vatican upbeat on possibility of Pope Francis ...	MOSCOW (Reuters) - Vatican Secretary of State ...	worldnews	August 22, 2017
21416	Indonesia to buy $1.14 billion worth of Russia...	JAKARTA (Reuters) - Indonesia will buy 11 Sukh...	worldnews	August 22, 2017

21417 rows × 4 columns

REAL NEWS

Importing, Labelling and concatenating the datasets

Data PreProcessing

```
# Importing libraries

import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
```

```
# Importing Dataset

truenews = pd.read_csv('true.csv')
fakenews = pd.read_csv('fake.csv')
```

```
# Labelling real and fake news

truenews['label'] = "true"
fakenews['label'] = "fake"
```

```
# Concatanating the datasets

news = pd.concat([truenews, fakenews])

# Shuffling the rows

from sklearn.utils import shuffle
news = shuffle(news)

news = news.reset_index(drop=True)
```

```
news.head()
```

	title	text	subject	date	label
0	Highlights of Reuters interview with House Spe...	WASHINGTON (Reuters) - Here are highlights of ...	politicsNews	October 25, 2017	true
1	Czech president's spokesman likens EU to Third...	PRAGUE (Reuters) - The Czech president s spoke...	worldnews	September 28, 2017	true
2	#PresidentObamaNotBarry Protests Blatant Raci...	There is a hashtag trending on Twitter in prot...	News	July 10, 2016	fake
3	Police fire tear gas to halt opposition protes...	NAIROBI (Reuters) - Kenyan police used tear ga...	worldnews	October 16, 2017	true
4	The Trump presidency on March 23 at 7:03 P.M. EDT	(Reuters) - Highlights of the day for U.S. Pre...	politicsNews	March 23, 2017	true

```
news.tail()
```

	title	text	subject	date	label
44893	FEDS: Dozens of Muslim Girls had Genitals Muti...	Changing the name of something doesn t change ...	left-news	Jun 7, 2017	fake
44894	BOOM! FIRST ANTIFA Coward ARRESTED For Not Rem...	The video below is an excellent summary of wha...	left-news	Apr 27, 2017	fake
44895	BREAKING: WIKILEAKS RELEASES LIST Of Reporters...	Here s the behind-the-scenes scoop:The Clinton...	politics	Oct 10, 2016	fake
44896	Elijah Cummings Just Asked 5 Questions The Wh...	Rep. Elijah E. Cummings is demanding answers a...	News	April 1, 2017	fake
44897	Lindsey Graham Is Literally Begging Republica...	Republicans are beginning to jump ship, and ar...	News	June 8, 2016	fake

Creating new field, cleaning data, making text lowercase and removing punctuation

```
# Combining text and title for entire news article

news["article"] = news["title"] + news["text"]
```

```
def punctuation_removal(text):
    all_list = [char for char in text if char not in string.punctuation]
    clean_str = ''.join(all_list)
    return clean_str
```

```
news['article'] = news['article'].apply(lambda x: x.lower())
```

```
import nltk
from nltk.corpus import stopwords
stop = stopwords.words('english')

news['article'] = news['article'].apply(lambda x: ' '.join([word for word in x.split() if word
```

```
import string

news['clean'] = news['article'].apply(punctuation_removal)
news = news.drop(['article'], axis=1)
```

Dropping unnecessary columns and rows with null values

```
# Removing the uneccesarry columns

news.drop(["title"],axis=1,inplace=True)
news.drop(["text"],axis=1,inplace=True)
news.drop(["date"],axis=1,inplace=True)
```

```
news.dropna()
```

	subject	article
0	politicsNews	Trump close to decision on addressing Chinese ...
1	politicsNews	Typical U.S. family earning $100,000 toget 1...
2	worldnews	U.S.-led surveillance aircraft leave area near...
3	News	Comey Sends DAMNING Warning About Future Elec...
4	worldnews	As Catalan vote looms, jailed leader offers ol...
...	...	...
44893	worldnews	Britain's May sees off challenges to Brexit pl...
44894	News	Ann Coulter Turns On Trump Over His Immigrati...
44895	politicsNews	Factbox: What is in Republican tax bill? Here ...
44896	politicsNews	Democrat gun control sit-in sparks social medi...
44897	News	The NRA Just Mocked Kim Kardashian After She ...

44898 rows × 2 columns

As this is a very large dataset we can drop the rows with null values without having to worry about reducing the training data for our model

Stemming and Splitting data into testing and training datasets

```
from nltk.stem.porter import PorterStemmer

port_stem = PorterStemmer()

def stemming(text):
    stemmed_content = [port_stem.stem(word) for word in text if not word in stopwords.words('english')]
    stemmed_content = ' '.join(stemmed_content)
    return stemmed_content
```

```
# Splitting the data into testing and training data

from sklearn.model_selection import train_test_split

X_train, X_test, y_train, y_test = train_test_split(news['clean'], news.label, test_size=0.2, random_state=42)
```

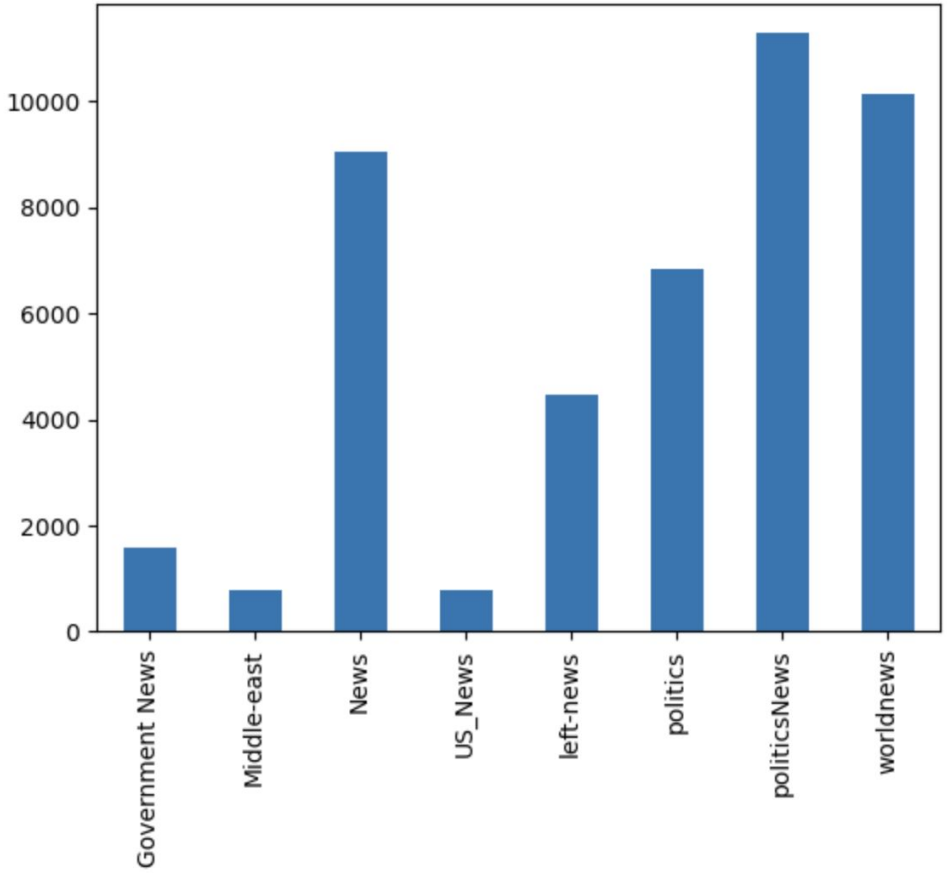
And that's all for data PreProcessing.....

03

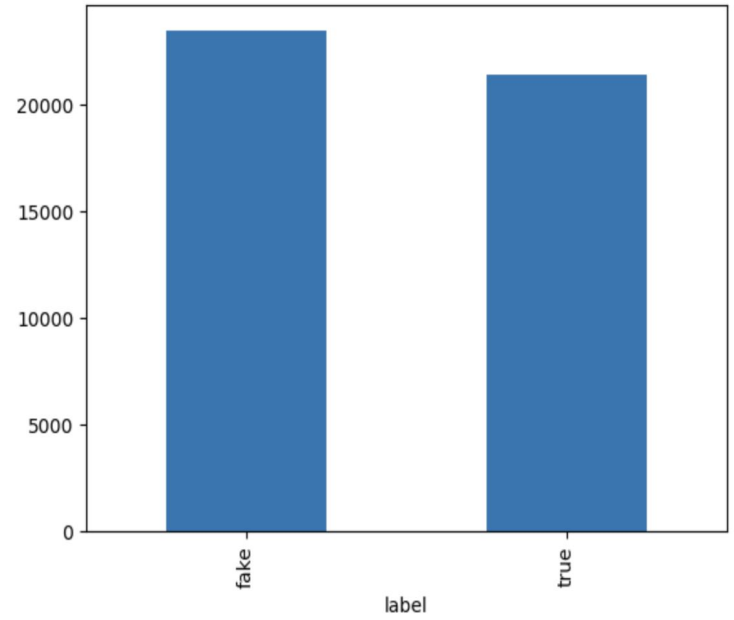
Data Visualisation

Using matplotlib and WordCloud to visualise the data

```
news.groupby(['subject'])['clean'].count().plot(kind="bar")
plt.show()
```

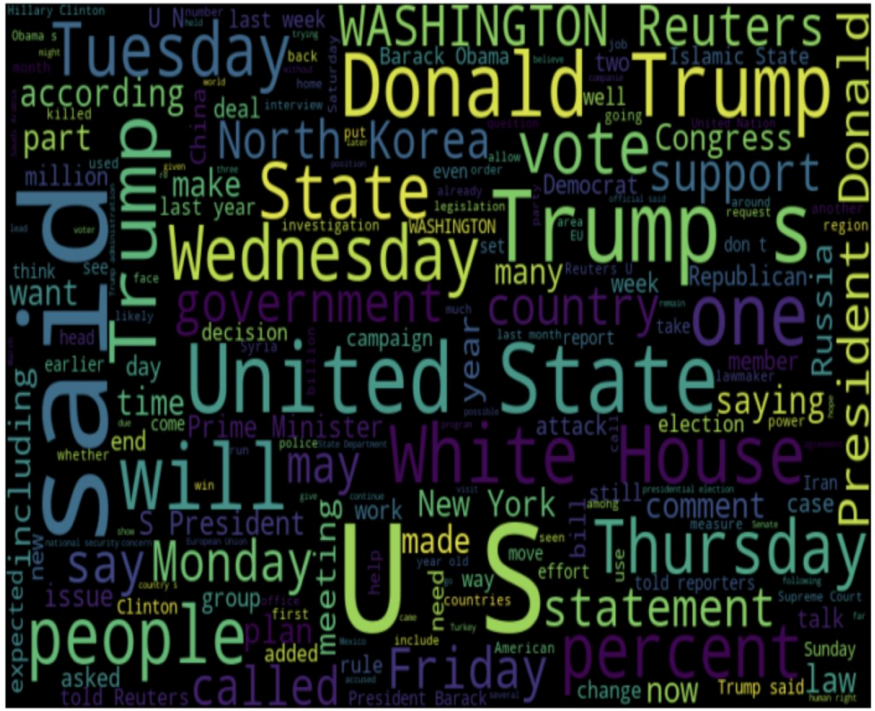


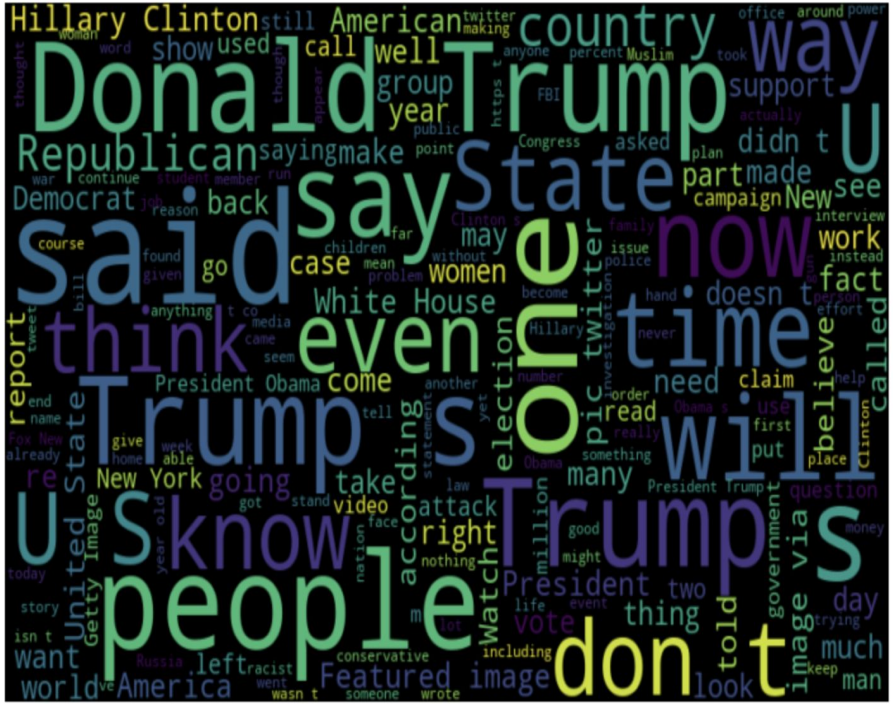

```
news.groupby(['label'])['clean'].count().plot(kind="bar")
plt.show()
```



```
all_words = ' '.join([text for text in truenews.text])

wordcloud = WordCloud(width= 800, height= 500,
                      max_font_size = 110).generate(all_words)

plt.figure(figsize=(9,6))
plt.imshow(wordcloud, interpolation='bilinear')
plt.axis("off")
plt.show()
```

```
all_words = ' '.join([text for text in fakenews.text])

wordcloud = WordCloud(width= 800, height= 500,
                      max_font_size = 110,).generate(all_words)

plt.figure(figsize=(9,6))
plt.imshow(wordcloud, interpolation='bilinear')
plt.axis("off")
plt.show()
```





04

Modelling

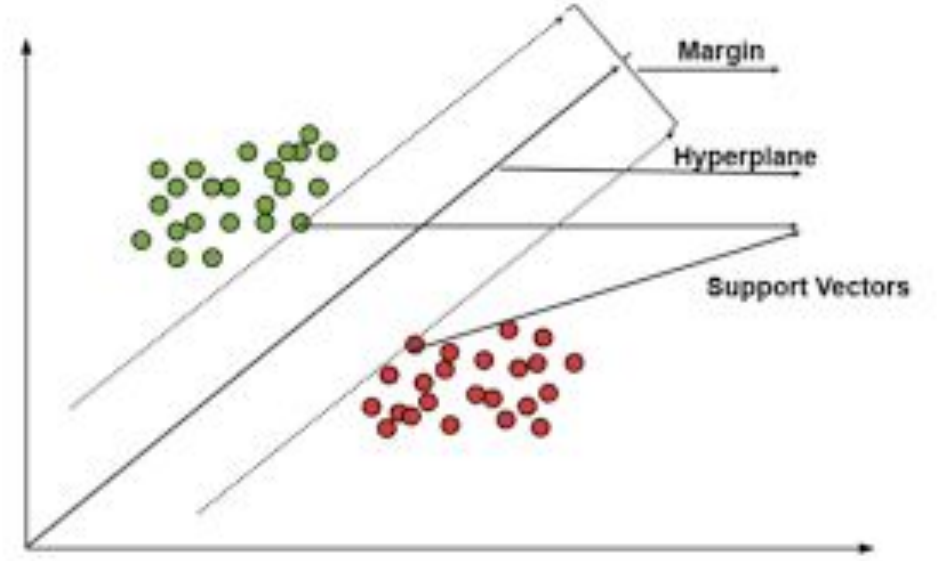
We tested many different models like Logistic Regression, KNN, Decision Trees etc.. to see which is most accurate

Models

1. SVC
2. Random Forest
3. Naive Bayes

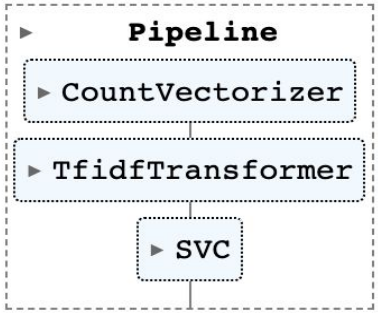
How does SVC Work?

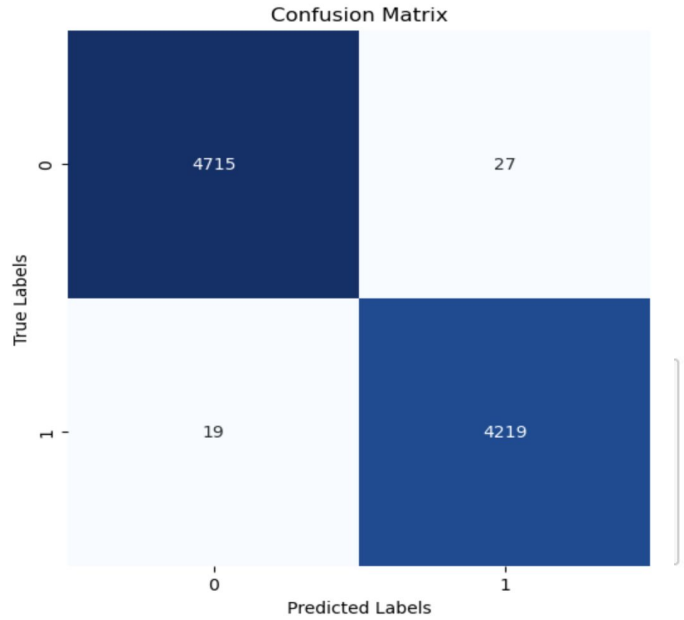
The goal of SVC algorithm is to create the best line or decision boundary that can segregate n-dimensional space into classes so that we can easily put the new data point into the correct category in the future.



Support Vector Classifier

```
pipeline_svc = Pipeline([
    ('vect', CountVectorizer()),  # strings to token integer counts
    ('tfidf', TfidfTransformer()),  # integer counts to weighted TF-IDF scores
    ('model', SVC(kernel="linear", C=1))])
pipeline_svc.fit(X_train,y_train)
```



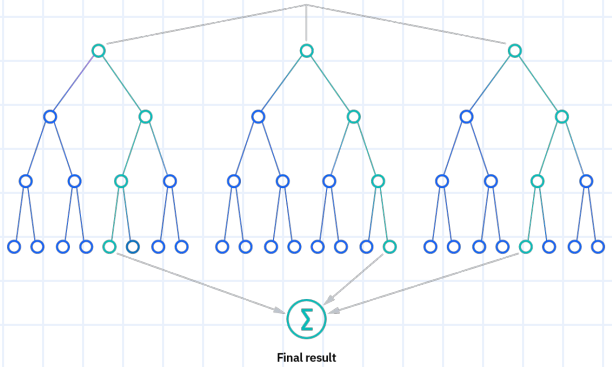



```
predictions_svc = pipeline_svc.predict(X_test)

score_svc = accuracy_score(y_test, predictions_svc)
conf_matrix_svc = confusion_matrix(y_test, predictions_svc)
print(conf_matrix_svc, score_svc)
```

```
[[4715   27]
 [  19 4219]] 0.9948775055679288
```

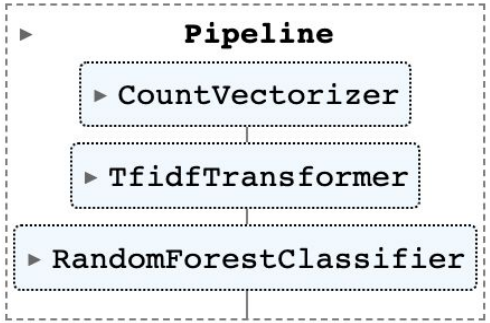
How does Random Forest Classifier work?

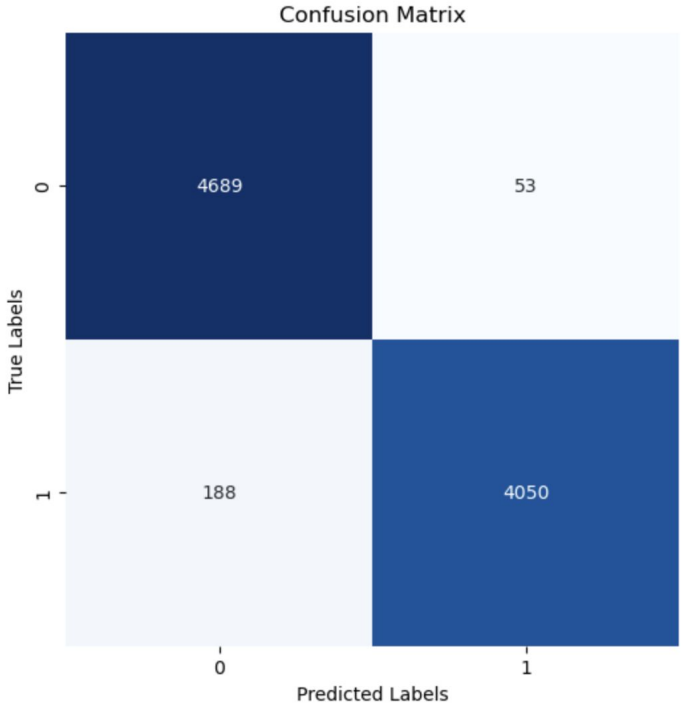
Random forest classifier works on the principle of permutation and combination. It is called “forest” because it is made up of many individual decision trees and it is “random” because it introduces randomness into the training process. We find all possible trees based on different priority orders and our aim is to find the best fit. While creating each decision tree, the Random Forest algorithm randomly selects a subset of the data (a random sample) and a subset of the features (the attributes or characteristics of the data). This randomness makes each tree slightly different from the others.



Random Forest Classifier

```
pipeline_rf = Pipeline([
    ('vect', CountVectorizer()),  # strings to token integer counts
    ('tfidf', TfidfTransformer()),  # integer counts to weighted TF-IDF scores
    ('model', RandomForestClassifier(n_estimators = 10, criterion = 'entropy', random_state = 0
pipeline_rf.fit(X_train,y_train)
```



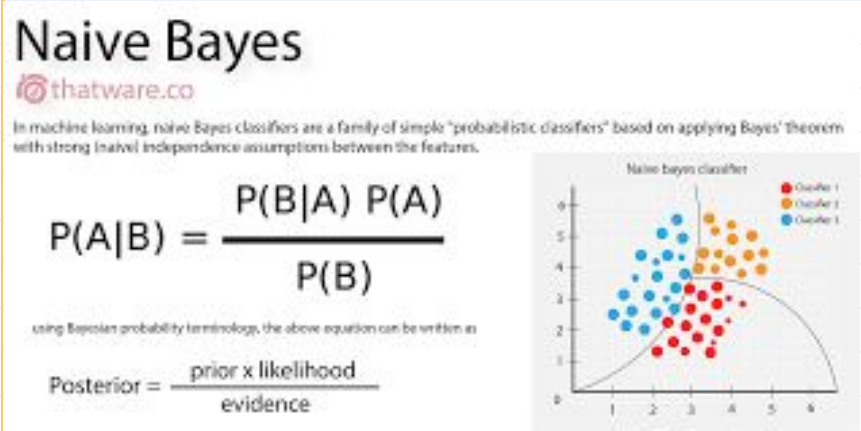



```
predictions_rf = pipeline_rf.predict(X_test)

score_rf = accuracy_score(y_test, predictions_rf)
conf_matrix_rf = confusion_matrix(y_test, predictions_rf)
print(conf_matrix_rf, score_rf)
```

```
[[4689   53]
 [ 188 4050]] 0.9731625835189309
```

How does Naive Bayes Model work?

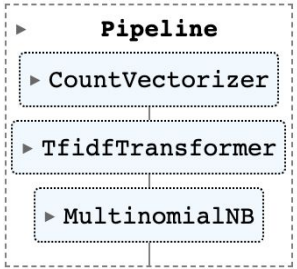
The Naive Bayes model is a probabilistic machine learning algorithm used for classification and, to some extent, for regression tasks. It is based on Bayes' theorem, which is a fundamental concept in probability theory. The "naive" part of the name comes from the assumption that the features used in the model are independent of each other, which is a simplifying but not always accurate assumption in practice.



Ex: Barack Obama is the new President

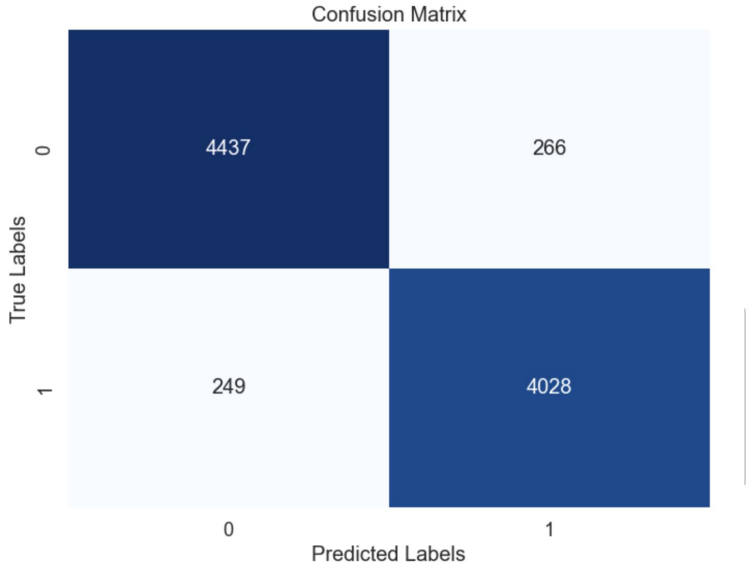
Naive Bayes

```
# Creating pipeline for model

pipeline_nb = Pipeline([
    ('bow', CountVectorizer()),  # strings to token integer counts
    ('tfidf', TfidfTransformer()),  # integer counts to weighted TF-IDF scores
    ('classifier', MultinomialNB()),  # train on TF-IDF vectors w/ Naive Bayes classifier
])
pipeline_nb.fit(X_train,y_train)
```




```
predictions_nb = pipeline_nb.predict(X_test)

score = accuracy_score(y_test, predictions_nb)
conf_matrix = confusion_matrix(y_test, predictions_nb)
print(conf_matrix, score)
```

```
[[4477  256]
 [ 239 4008]] 0.9448775055679287
```



Thank You