

Abhishek Sharma

Data Science and Business Analytics Intern

Task #7 : Stock Market Prediction

Domain : Data Science and Business Analytics

Getting Data

```
In [ ]: 1 import pandas_datareader as pdr

In [ ]: 1 import os
2
3 df=pdr.get_data_yahoo('ibm',start='2017-01-1',end='2020-12-15')

In [ ]: 1 df.head()

Out[3]:
```

	High	Low	Open	Close	Volume	Adj Close
Date						
2017-01-03	167.869995	166.009995	167.000000	167.190002	2934300.0	139.613220
2017-01-04	169.869995	167.360001	167.770004	169.259995	3381400.0	141.341766
2017-01-05	169.389999	167.259995	169.250000	168.699997	2682300.0	140.874130
2017-01-06	169.919998	167.520004	168.690002	169.529999	2945500.0	141.567230
2017-01-09	169.800003	167.619995	169.470001	167.649994	3189900.0	139.997330

```
In [ ]: 1 # DownLoding data
2 df.to_csv('ibm.csv')
```

Packages

```
In [ ]: 1 import pandas as pd
2 import numpy as np
3 import matplotlib.pyplot as plt
```

Data

```
In [ ]: 1 data = pd.read_csv('ibm.csv')

In [ ]: 1 data.head()

Out[7]:
```

	Date	High	Low	Open	Close	Volume	Adj Close
0	2017-01-03	167.869995	166.009995	167.000000	167.190002	2934300.0	139.613220
1	2017-01-04	169.869995	167.360001	167.770004	169.259995	3381400.0	141.341766
2	2017-01-05	169.389999	167.259995	169.250000	168.699997	2682300.0	140.874130
3	2017-01-06	169.919998	167.520004	168.690002	169.529999	2945500.0	141.567230
4	2017-01-09	169.800003	167.619995	169.470001	167.649994	3189900.0	139.997330

```
In [ ]: 1 data1=data.reset_index()['Close']

In [ ]: 1 data1

Out[9]:
```

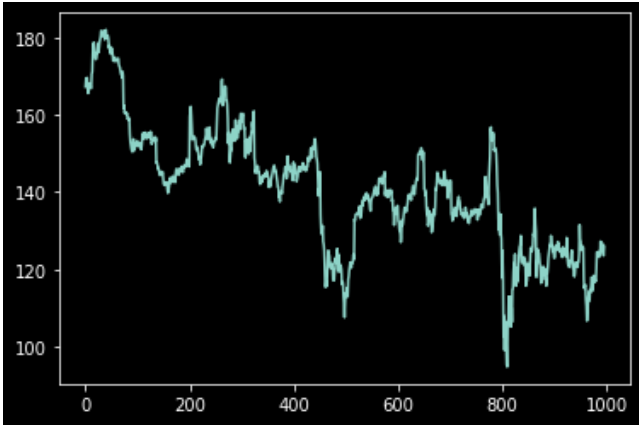
0	167.190002
1	169.259995
2	168.699997
3	169.529999
4	167.649994
...	
991	126.790001
992	124.959999
993	124.269997
994	123.529999
995	125.930000

Name: Close, Length: 996, dtype: float64

Visualization

```
In [ ]: 1 from matplotlib import style
2 plt.style.use(['dark_background'])

In [ ]: 1 plt.plot(data1);
```



```
In [ ]: 1 data1

Out[12]:
```

0	167.190002
1	169.259995
2	168.699997
3	169.529999
4	167.649994
...	
991	126.790001
992	124.959999
993	124.269997
994	123.529999
995	125.930000

Name: Close, Length: 996, dtype: float64

Preprocessing

```
In [ ]: 1 from sklearn.preprocessing import MinMaxScaler
2 scaler=MinMaxScaler(feature_range=(0,1))
3 data1=scaler.fit_transform(np.array(data1).reshape(-1,1))
```

```
In [ ]: 1 print(data1)

[[0.83069518]
 [0.85443906]
 [0.8480156 ]
 [0.85753615]
 [0.83597152]
 [0.81153943]
 [0.83711864]
 [0.83941271]
 [0.83241569]
 [0.83872451]
 [0.82622168]
 [0.82633632]
 [0.86923613]
 [0.87474193]
 [0.93060331]
 [0.95801785]
 [0.96226206]
 [0.94666215]
 [0.92945637]
 ...]]

In [ ]: 1 train_size=int(len(data1)*0.65)
2 test_size=len(data1)-train_size
3 print('Train Size:', train_size)
4 print('Test Size:',test_size)

Train Size: 647
Test Size: 349

In [ ]: 1 s=len(data1)
2 s

Out[16]: 996
```

Train Test Split

```
In [ ]: 1 train_data,test_data=data1[0:train_size:],data1[train_size:len(data1),:]

In [ ]: 1 def create_dataset(dataset, time_step=1):
2     datax,datay=[],[]
3     for i in range(len(dataset)-time_step-1):
4         a=dataset[i:(i+time_step),0]
5         datax.append(a)
6         datay.append(dataset[i+time_step,0])
7     return np.array(datax),np.array(datay)

In [ ]: 1 time_step=100
2 x_train,y_train=create_dataset(train_data,time_step)
3 x_test,y_test=create_dataset(test_data,time_step)

In [ ]: 1 print(x_train)

[[0.83069518 0.85443906 0.8480156 ... 0.65680204 0.66230784 0.67022253]
 [0.85443906 0.8480156 0.85753615 ... 0.66230784 0.67022253 0.66207856]
 [0.8480156 0.85753615 0.83597152 ... 0.67022253 0.66207856 0.65336085]
 ...
 [0.50080289 0.49449424 0.48417067 ... 0.63053463 0.63787575 0.633861 ]
 [0.49449424 0.48417067 0.46558848 ... 0.63787575 0.633861 0.63799039]
 [0.48417067 0.46558848 0.46249139 ... 0.633861 0.63799039 0.64911681]]

In [ ]: 1 x_train=x_train.reshape(x_train.shape[0],x_train.shape[1],1)
2 x_test=x_test.reshape(x_test.shape[0],x_test.shape[1],1)
```

Tensor Flow

```
In [ ]: 1 from tensorflow.keras.models import Sequential
2 from tensorflow.keras.layers import Dense
3 from tensorflow.keras.layers import LSTM

In [ ]: 1 model=Sequential()
2 model.add(LSTM(50,return_sequences=True,input_shape=(100,1)))
3 model.add(LSTM(50,return_sequences=True))
4 model.add(LSTM(50))
5 model.add(Dense(1))
6 model.compile(loss='mean_squared_error',optimizer='adam')

In [ ]: 1 model.summary()

Model: "sequential"
Layer (type) Output Shape Param #
=====
lstm (LSTM) (None, 100, 50) 10400
-----
lstm_1 (LSTM) (None, 100, 50) 20200
-----
lstm_2 (LSTM) (None, 50) 20200
-----
dense (Dense) (None, 1) 51
=====
Total params: 50,851
Trainable params: 50,851
Non-trainable params: 0
-----
```

Model Training

```
In [ ]: 1 model.fit(x_train,y_train,validation_data=(x_test,y_test),epochs=100,batch_size=64,verbose=1)

Epoch 1/100
9/9 [=====] - 7s 354ms/step - loss: 0.2359 - val_loss: 0.0640
Epoch 2/100
9/9 [=====] - 2s 193ms/step - loss: 0.0247 - val_loss: 0.0098
Epoch 3/100
9/9 [=====] - 2s 196ms/step - loss: 0.0119 - val_loss: 0.0296
Epoch 4/100
9/9 [=====] - 2s 190ms/step - loss: 0.0078 - val_loss: 0.0127
Epoch 5/100
9/9 [=====] - 2s 190ms/step - loss: 0.0063 - val_loss: 0.0202
Epoch 6/100
9/9 [=====] - 2s 191ms/step - loss: 0.0051 - val_loss: 0.0134
Epoch 7/100
9/9 [=====] - 2s 191ms/step - loss: 0.0054 - val_loss: 0.0170
Epoch 8/100
9/9 [=====] - 2s 182ms/step - loss: 0.0048 - val_loss: 0.0121
Epoch 9/100
9/9 [=====] - 2s 184ms/step - loss: 0.0042 - val_loss: 0.0144
Epoch 10/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 11/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 12/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 13/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 14/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 15/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 16/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 17/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 18/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 19/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 20/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 21/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 22/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 23/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 24/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 25/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 26/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 27/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 28/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 29/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 30/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 31/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 32/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 33/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 34/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 35/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 36/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 37/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 38/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 39/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 40/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 41/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 42/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 43/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 44/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 45/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 46/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 47/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 48/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 49/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 50/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 51/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 52/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 53/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 54/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 55/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 56/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 57/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 58/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 59/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 60/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 61/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 62/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 63/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 64/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 65/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 66/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 67/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 68/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 69/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 70/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 71/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 72/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 73/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 74/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 75/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 76/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 77/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 78/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 79/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 80/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 81/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 82/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 83/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 84/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 85/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 86/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 87/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 88/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 89/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 90/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 91/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 92/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 93/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 94/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 95/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 96/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 97/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 98/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 99/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114
Epoch 100/100
9/9 [=====] - 2s 183ms/step - loss: 0.0047 - val_loss: 0.0114

In [ ]: 1 train_predict=model.predict(x_train)
2 test_predict=model.predict(x_test)

In [ ]: 1 train_predict=scaler.inverse_transform(train_predict)
2 test_predict=scaler.inverse_transform(test_predict)
```

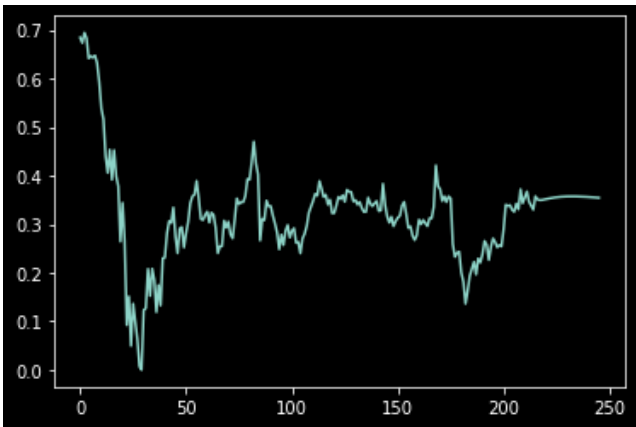
Evaluate Metrics

```
In [ ]: 1 import math
2 from sklearn.metrics import mean_squared_error
3 math.sqrt(mean_squared_error(y_train,train_predict))

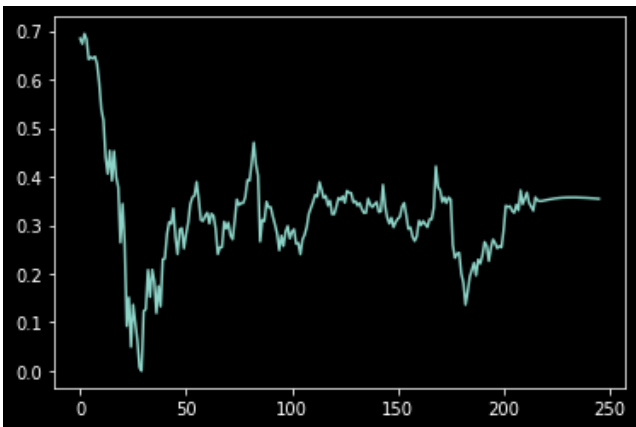
Out[28]: 141.86904849312512
```



```
In [ ]: 1 df3=data1.tolist()
2 df3.extend(1st_output)
3 plt.plot(df3[780:]);
```



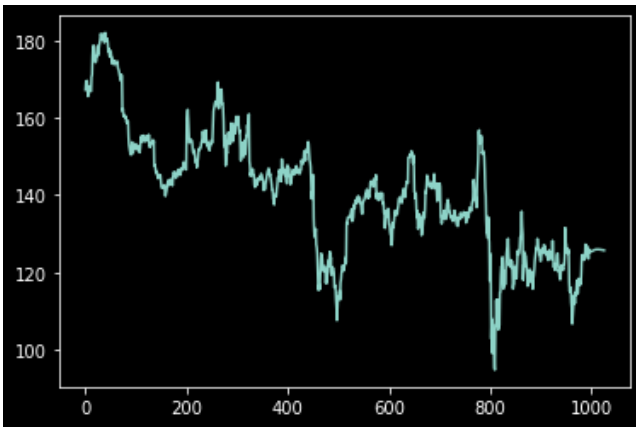
```
In [ ]: 1 df3=data1.tolist()
2 df3.extend(1st_output)
3 plt.plot(df3[780:]);
```



```
In [ ]: 1 df3=scaler.inverse_transform(df3).tolist()
```

```
In [ ]: 1 plt.plot(df3)
```

Out[41]: [<matplotlib.lines.Line2D at 0x7f1aa69e9978>]



Steps are involved in Stock Sentiment Analysis

Packages

```
In [ ]: 1 import pandas as pd
```

```
In [ ]: 1 from google.colab import drive
2 drive.mount('/content/drive/')

```

Drive already mounted at /content/drive/; to attempt to forcibly remount, call drive.mount("/content/drive/", force_remount=True).

Importing Data

```
In [ ]: 1 df=pd.read_csv('/content/drive/MyDrive/Data/news.csv', encoding = "ISO-8859-1")
```

```
In [ ]: 1 df.head()
```

	Date	Label	Top1	Top2	Top3	Top4	Top5	Top6	Top7	Top8	Top9	Top10	Top11	Top12	Top13	Top14	Top15	Top16	Top17	Top18	Top19	Top20	Top21	Top22
0	2000-01-03	0	A 'hindrance to operations': extracts from the...	Scorecard	Hughes' instant hit buoys Blues	Jack gets his skates on at ice-cold Alex	Chaos as Maracana builds up for United	Depleted Leicester prevail as Elliott spoils E...	Hungry Spurs sense rich pickings	Gunners so wide of an easy target	Derby raise a glass to Strupar's debut double	Southgate strikes, Leeds pay the penalty	Hammers hand Robson a youthful lesson	Saints party like it's 1999	Wear wolves have turned into lambs	Stump mike catches testy Gough's taunt	Langer escapes to hit 167	Flintoff injury piles on woe for England	Hunters threaten Jospin with new battle of the...	Kohl's successor drawn into scandal	The difference between men and women	Sara Denver, nurse turned solicitor	Diana's landmine crusade put Tories in a panic	Yeltsin's resignation caught oppositior flat-f..
1	2000-01-04	0	Scorecard	The best lake scene	Leader: German sleaze inquiry	Cheerio, boyo	The main recommendations	Has Cubie killed fees?	Has Cubie killed fees?	Has Cubie killed fees?	Hopkins 'furious' at Foster's lack of Hannibal...	Has Cubie killed fees?	A tale of two tails	I say what I like and I like what I say	Elbows, Eyes and Nipples	Task force to assess risk of asteroid collision	How I found myself at last	On the critical list	The timing of their lives	Dear doctor	Irish court halts IRA man's extradition to Nor...	Burundi peace initiative fades after rebels re...	PE points the way forward to the ECB	Campaigners keep up pressure or Nazi wa crime..
2	2000-01-05	0	Coventry caught on counter by Flo	United's rivals on the road to Rio	Thatcher issues defence before trial by video	Police help Smith lay down the law at Everton	Tale of Trautmann bears two more retellings	England on the rack	Pakistan retaliate with call for video of Walsh	Cullinan continues his Cape monopoly	McGrath puts India out of their misery	Blair Witch bandwagon rolls on	Pele turns up heat on Ferguson	Party divided over Kohl slush fund scandal	Manchester United (England)	Women in record South Pole walk	Vasco da Gama (Brazil)	South Melbourne (Australia)	Necaxa (Mexico)	Real Madrid (Spain)	Raja Casablanca (Morocco)	Corinthians (Brazil)	Tony's pet project	Al Nass (Saud Arabia
3	2000-01-06	1	Pilgrim knows how to progress	Thatcher facing ban	McIlroy calls for Irish fighting spirit	Leicester bin stadium blueprint	United braced for Mexican wave	Auntie back in fashion, even if the dress look...	Shoab appeal goes to the top	Hussain hurt by 'shambles' but lays blame on e...	England's decade of disasters	Revenge is sweet for jubilant Cronje	Our choice, not theirs	Profile of former US Nazi Party officer Willia...	New evidence shows record of war crimes suspec...	The rise of the supernerds	Written on the body	Putin admits Yeltsin quit to give him a head s...	BBC worst hit as digital TV begins to bite	How much can you pay for...	Christmas glitches	Upending a table, Chopping a line and Scoring ...	Scientific evidence 'unreliable', defence claims	Fusco wins judicia review ir extradition case
4	2000-01-07	1	Hitches and Horlocks	Beckham off but United survive	Breast cancer screening	Alan Parker	Guardian readers: are you all whingers?	Hollywood Beyond	Ashes and diamonds	Whingers - a formidable minority	Alan Parker - part two	Thuggery, Toxins and Ties	Met faces fresh attack on race crime	Everton fans top racist 'league of shame'	Our breasts, ourselves	Russia's new boss has an extremely strange his...	Always and forever	Most everywhere: UDIs	Most wanted: Chloe lunettes	Return of the cane 'completely off the agenda'	From Sleepy Hollow to Greenland	Blunkett outlines vision for over 11s	Embattled Dobson attacks 'play now, pay later' ...	Doom and the Dome

```
In [ ]: 1 train = df[df['Date'] < '20150101']
2 test = df[df['Date'] > '20141231']
```

Preprocessing steps are involved in this task


```
In [ ]: 1 data=train.iloc[:,2:27]
2 data.replace("[^a-zA-Z]", " ", regex=True, inplace=True)
3
4 # Renaming column names for ease of access
5 list1= [i for i in range(25)]
6 new_Index=[str(i) for i in list1]
7 data.columns= new_Index
8 data.head(5)

Out[47]:
```

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	
0	A hindrance to operations extracts from the...	Scorecard	Hughes instant hit buoys Blues	Jack gets his skates on at ice cold Alex	Chaos as Maracana builds up for United	Depleted Leicester prevail as Elliott spoils E...	Hungry Spurs sense rich pickings	Gunners so wide of an easy target	Derby raise a glass to Strupar s debut double	Southgate strikes Leeds pay the penalty	Hammers hand Robson a youthful lesson	Saints party like it s	Wear wolves have turned into lambs	Stump mike catches testy Gough s taunt	Langer escapes to hit	Flintoff injury piles on woe for England	Hunters threaten Jospin with new battle of the...	Kohl s successor drawn into scandal	The difference between men and women	Sara Denver nurse turned solicitor	Diana s landmine crusade put Tories in a panic	Yeltsin s resignation caught opposition flat f...	Russian roulette	Sold
1	Scorecard	The best lake scene	Leader German sleaze inquiry	Cheerio boyo	The main recommendations	Has Cubie killed fees	Has Cubie killed fees	Has Cubie killed fees	Hopkins furious at Foster s lack of Hannibal...	Has Cubie killed fees	A tale of two tails	I say what I like and I like what I say	Elbows Eyes and Nipples	Task force to assess risk of asteroid collision	How I found myself at last	On the critical list	The timing of their lives	Dear doctor	Irish court halts IRA man s extradition to Nor...	Burundi peace initiative fades after rebels re...	PE points the way forward to the ECB	Campaigners keep up pressure on Nazi war crime...	Jane Ratcliffe	m thir) woul kn with the
2	Coventry caught on counter by Flo	United s rivals on the road to Rio	Thatcher issues defence before trial by video	Police help Smith lay down the law at Everton	Tale of Trautmann bears two more retellings	England on the rack	Pakistan retaliate with call for video of Walsh	Cullinan continues his Cape monopoly	McGrath puts India out of their misery	Blair Witch bandwagon rolls on	Pele turns up heat on Ferguson	Party divided over Kohl slush fund scandal	Manchester United England	Women in record South Pole walk	Vasco da Gama Brazil	South Melbourne Australia	Necaxa Mexico	Real Madrid Spain	Raja Casablanca Morocco	Corinthians Brazil	Tony s pet project	Al Nassr Saudi Arabia	Ideal Holmes show	Pinoc leav hosp a te
3	Pilgrim knows how to progress	Thatcher facing ban	Mclroy calls for Irish fighting spirit	Leicester bin stadium blueprint	United braced for Mexican wave	Auntie back in fashion even if the dress look...	Shoab appeal goes to the top	Hussain hurt by shambles but lays blame on e...	England s decade of disasters	Revenge is sweet for jubilant Cronje	Our choice not theirs	Profile of former US Nazi Party officer Willia...	New evidence shows record of war crimes suspec...	The rise of the superners	Written on the body	Putin admits Yeltsin quit to give him a head s...	BBC worst hit as digital TV begins to bite	How much can you pay for	Christmas glitches	Upending a table Chopping a line and Scoring ...	Scientific evidence unreliable defence claims	Fusco wins judicial review in extradition case	Rebels thwart Russian advance	B ord shz u fail N
4	Hitches and Horlocks	Beckham off but United survive	Breast cancer screening	Alan Parker	Guardian readers are you all whingers	Hollywood Beyond	Ashes and diamonds	Whingers a formidable minority	Alan Parker part two	Thuggery Toxins and Ties	Met faces fresh attack on race crime	Everton fans top racist league of shame	Our breasts ourselves	Russia s new boss has an extremely strange his...	Always and forever	Most everywhere UDIs	Most wanted Chloe lunettes	Return of the cane completely off the agenda	From Sleepy Hollow to Greenland	Blunkett outlines vision for over s	Embattled Dobson attacks play now pay later ...	Doom and the Dome	What is the north south divide	Ait releases from

Converting the headlines into lower case

```
In [ ]: 1 for index in new_Index:
2     data[index]=data[index].str.lower()
3 data.head(1)

Out[48]:
```

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
0	a hindrance to operations extracts from the...	scorecard	hughes instant hit buoys blues	jack gets his skates on at ice cold alex	chaos as maracana builds up for united	depleted leicester prevail as elliot spoils e...	hungry spurs sense rich pickings	gunners so wide of an easy target	derby raise a glass to strupar s debut double	southgate strikes leeds pay the penalty	hammers hand robson a youthful lesson	saints party like it s	wear wolves have turned into lambs	stump mike catches testy gough s taunt	langer escapes to hit	flintoff injury piles on woe for england	hunters threaten jospin with new battle of the...	kohl s successor drawn into scandal	the difference between men and women	sara denver nurse turned solicitor	diana s landmine crusade put tories in a panic	yeltsin s resignation caught opposition flat f...	russian roulette	sold out	recovering a title

```
In [ ]: 1 headlines = []
2 for row in range(0,len(data.index)):
3     headlines.append(' '.join(str(x) for x in data.iloc[row,0:25]))

In [ ]: 1 headlines[0]

Out[51]: 'a hindrance to operations extracts from the leaked reports scorecard hughes instant hit buoys blues jack gets his skates on at ice cold alex chaos as maracana builds up for united depleted leicester prevail as elliot spoils everton s party hungry spurs sense rich pickings gunners so wide of an easy target derby raise a glass to strupar s debut double southgate strikes leeds pay the penalty hammers hand robson a youthfu l lesson saints party like it s wear wolves have turned into lambs stump mike catches testy gough s taunt langer escapes to hit flintoff injury piles on woe for england hunters threaten jospin with new bat tle of the somme kohl s successor drawn into scandal the difference between men and women sara denver nurse turned solicitor diana s landmine crusade put tories in a panic yeltsin s resignation caught opposition f lat footed russian roulette sold out recovering a title'
```

Apply CountVectorizer and Randomforest classifier

```
In [ ]: 1 from sklearn.feature_extraction.text import CountVectorizer
2 from sklearn.ensemble import RandomForestClassifier

Apply Bag of words

In [ ]: 1 countvector=CountVectorizer(ngram_range=(2,2))
2 traindataset=countvector.fit_transform(headlines)
```

Apply RandomForest Classifier

```
In [ ]: 1 randomclassifier=RandomForestClassifier(n_estimators=200,criterion='entropy')
2 randomclassifier.fit(traindataset,train['Label'])

Out[54]: RandomForestClassifier(bootstrap=True, ccp_alpha=0.0, class_weight=None,
criterion='entropy', max_depth=None, max_features='auto',
max_leaf_nodes=None, max_samples=None,
min_impurity_decrease=0.0, min_impurity_split=None,
min_samples_leaf=1, min_samples_split=2,
min_weight_fraction_leaf=0.0, n_estimators=200,
n_jobs=None, oob_score=False, random_state=None,
verbose=0, warm_start=False)
```

Prediction

```
In [ ]: 1 test_transform= []
2 for row in range(0,len(test.index)):
3     test_transform.append(' '.join(str(x) for x in test.iloc[row,2:27]))
4 test_dataset = countvector.transform(test_transform)
5 predictions = randomclassifier.predict(test_dataset)
```

Evaluate the Model

```
In [ ]: 1 from sklearn.metrics import classification_report,confusion_matrix,accuracy_score
```

Confusion Matrix

```
In [ ]: 1 matrix=confusion_matrix(test['Label'],predictions)
2 print(matrix)

[[137 49]
[ 7 185]]
```

Accuracy Score

```
In [ ]: 1 score=accuracy_score(test['Label'],predictions)
        2 print(score*100)
```

85.18518518518519

Classification report

```
In [ ]: 1 report=classification_report(test['Label'],predictions)
        2 print(report)
```

	precision	recall	f1-score	support
0	0.95	0.74	0.83	186
1	0.79	0.96	0.87	192
accuracy			0.85	378
macro avg	0.87	0.85	0.85	378
weighted avg	0.87	0.85	0.85	378