A ETE Project Report

on

Real Time Stock Prediction Using ML

Submitted in partial fulfillment of the
requirement for the award of the degree of

Bachelor of Technology Computer
Science Engineering

GALGOTIAS
UNIVERSITY
(Established under Galgotias University Uttar Pradesh Act No. 14 of 2011)

Under The Supervision of
Dr . T Ganesh Kumar
Associate Professor

Submitted By
Harshit Saxena
(18SCSE1010268)
Kumar Shubham
(18SCSE1010452)

SCHOOL OF COMPUTING SCIENCE AND ENGINEERING
DEPARTMENT OF COMPUTER SCIENCE ENGINEERING
GALGOTIAS UNIVERSITY, GREATER NOIDA
INDIA
December,2021

SCHOOL OF COMPUTING SCIENCE AND ENGINEERING
GALGOTIAS UNIVERSITY, GREATER NOIDA

CANDIDATE'S DECLARATION

We hereby certify that the work which is being presented in the project entitled **"Real Time Stock Prediction Using ML"** in partial fulfillment of the requirements for the award of the the School of Computing Science and Engineering of Galgotias University, Greater Noida, is an original work carried out during the period of 2021 to December 2021, under the supervision of Dr. T Ganesh Kumar, Associate Professor, Department of Computer Science and Engineering/Computer Application and Information and Science, of School of Computing Science and Engineering , Galgotias University, Greater Noida

The matter presented in the project has not been submitted by us for the award of any other degree of this or any other places.

Harshit Saxena

Kumar Shubham

This is to certify that the above statement made by the candidates is correct to the best of my knowledge.

Dr. T Ganesh Kumar

Associate Professor

CERTIFICATE

The Final Project Viva-Voice examination of Harshit Saxena (18SCSE1010268) and Kumar Shubham (18SCSE1010452) has been held on __________________ and his/her work is recommended for the award of Bachelor Of Technology.

Signature of Examiner(s) **Signature of Supervisor(s)**

Signature of Project Coordinator **Signature of Dean**

Date: December, 2021

Place: Greater Noida

TABLE OF CONTENTS

Abstract

Accurate forecasting of stock market returns is a highly challenging task due to volatile and indirect financial markets. With the introduction of installation technology and the development of computer skills, pre-prepared dictionary methods have been shown to be more effective in predicting prices. Technical and basic or periodic analysis is used by many stock traders while making stock forecasts. System language is used to predict the stock market using Python machine learning. On this page we suggest a Learning Mission (ML) method trained .

In the details of available stocks and intelligence and then use the information obtained to obtain accurate forecasts. In this context the study uses a machine learning process called Support Vector Machine (SVM) predicting prices for large and small stocks with three different markets, prices on both days and waves up to minute.

The first thing we have taken into account is the dataset of the stock market prices from previous year. The dataset was pre-processed and tuned up for real analysis. Hence, our paper will also focus on data preprocessing of the raw dataset. Secondly, after pre processing the data, we will review the use of random forest, support vector machine on the dataset and the outcomes it generates. In addition, the proposed paper examines the use of the prediction system in real-world settings and issues associated with the accuracy of the overall values given. The paper also presents a machine-learning model to predict the longevity of stock in a competitive market. The successful prediction of the stock will be a great asset for the stock market institutions and will provide real-life solutions to the problems that stock investors face.

Introduction

Basically, most retailers have a lot of money from stock markets buy stocks and stocks at low prices price and later when you sell them at a higher price. Practice at stock market predictions are nothing new but still a problem kept discussed by various organizations. The stock market seems strong, unpredictable and out of line by nature. Predicting stock prices a challenging task as it depends on a variety of factors including but not limited to political, global and economic conditions, Company financial and performance reports etc. Therefore, to maximize profits and reduce losses, strategiesfor this predicting stock prices in advance by analyzing trends overthe past few years, can be very helpful.

The stock market is basically an aggregation of various buyers and sellers of stock. A stock (also known as shares more commonly) in general represents ownership claims on business by a particular individual or a group of people. The attempt to determine the future value of the stock market is known as a stock market prediction. The prediction is expected to be robust, accurate and efficient. The system is also expected to take into account all the variables that might affect the stock's value and performance. There are various methods and ways of implementing the prediction system like Fundamental Analysis, Technical Analysis, Machine Learning, Market Mimicry, and Time series aspect structuring. With the advancement of the digital era, the prediction has moved up into the technological realm. The most prominent and promising technique involves the use of Artificial Neural Networks, Recurrent Neural Networks, that is basically the implementation of machine learning.

Problem Definition

Stock market prediction is basically defined as trying to determine the stock value and offer a robust idea for the people to know and predict the market and the stock prices. It is generally presented using the quarterly financial ratio using the dataset. Thus, relying on a single dataset may not be sufficient for the prediction and can give a result which is inaccurate. Hence, we are contemplating towards the study of machine learning with various datasets integration to predict the market and the stock trends. The problem with estimating the stock price will remain a problem if a better stock market prediction algorithm is not proposed. Predicting how the stock market will perform is quite difficult. The movement in the stock market is usually determined by the sentiments of thousands of investors. Stock market prediction, calls for an ability to predict the effect of recent events on the investors. These events can be political events like a statement by a political leader, a piece of news on scam etc. It can also be an international event like sharp movements in currencies and commodity etc. All these events affect the corporate earnings, which in turn affects the sentiment of investors. It is beyond the scope of almost all investors to correctly and consistently predict these hyperparameters. All these factors make stock price prediction very difficult. Once the right data is collected, it then can be used to train a machine and to generate a predictive result.

Literature Survey

The predicted target for the stock market can be Future stock price or depreciation or market inclination. In the forecast there are two types such as dummy and real-time forecasts used in the stock market weather system. In the Dummy forecast they explain another set of rules and predicts the price of future shares as well average price calculated. In real-time prediction Forced internet was used and saw the current number of shares business development has led to the introduction of machine learning strategies for predictive systems in financial markets. In this project we use the Machine Learning process, i.e., Support Vector Machine (SVM) in to order stock market predictions and we use Python programming language.

The stock market prediction has become an increasingly important issue in the present time. One of the methods employed is technical analysis, but such methods do not always yield accurate results. So it is important to develop methods for a more accurate prediction. Generally, investments are made using predictions that are obtained from the stock price after considering all the factors that might affect it. The technique that was employed in this instance was a regression. Since financial stock marks generate enormous amounts of data at any given time a great volume of data needs to undergo analysis before a prediction can be made. Each of the techniques listed under regression hasits own advantages and limitations over its other counterparts. One of the noteworthy techniques that were mentioned was linear regression. The way linear regression models work is that they are often fitted using the stock prediction technique to analyze the data to be shown to user.

Methodology

In this project the prediction of stock market is done by the Support Vector Machine. Support Vector Machine (SVM) discriminates against them classifier officially defined by a divisive hyperplane.

In other words, the information provided by the training label (supervised learning), the algorithm extracts the appropriate hyperplane separating new models. On both sides space the hyperplane line divides the plane into two parts where each class lies on both sides.

Support Vector Machine (SVM) is considered the most relevant algorithms available at the time series predictions. The monitored algorithm can be used for both, regression and segmentation. SVM includes classifying data as a point in size space n . This magnitude is the identified symptoms some connection. The SVM algorithm pulls the limit above a set of data called hyper-plane, which separates data in two categories. Hyper-flight is the deciding factor later extended or extended back and forth between data points. A Support Vector Machine (SVM) is a discriminative classifier that formally defined by the separating hyperplane. In other words, the given labeled training data (supervised learning), the algorithm outputs the optimal hyperplane which categorizes new examples. In the two-dimensional space this hyperplane is a line dividing a plane into two parts where in each class lay in either side. Support Vector Machine (SVM) is considered to be as one of the most suitable algorithms available for the time series prediction. The supervised algorithm can be used in both, regression and classification. The SVM involves in plotting of data as point in the space of n dimensions.

Classification - Classification is an instance of supervised learning where a set is analyzed and categorized based on a common attribute. From the values or the data are given, classification draws some conclusion from the observed value. If more than one input is given then classification will try to predict one or more outcomes for the same. A few classifiers that are used here for the stock market prediction includes the random forest classifier, SVM classifier.

Random Forest Classifier

Random forest classifier is a type of ensemble classifier and also a supervised algorithm. It basically creates a set of decision trees, that yields some result. The basic approach of random class classifier is to take the decision aggregate of random subset decision tress and yield a final class or result based on the votes of the random subset of decision trees.

Parameters

The parameters included in the random forest classifier are n estimators which is total number of decision trees, and other hyper parameters like score to determine the generalization accuracy of the random forest, max features which includes the number of features for best-split. Min weight fraction leaf is the minimum weighted fraction of the sum total of weights of all the input samples required to be at a leaf node. Samples have equal weight when sample weight is not provided. SVM classifier SVM classifier is a type of discriminative classifier. The SVM uses supervised learning i.e. a labeled training data. The output are hyperplanes which categorizes the new dataset. They are supervised learning models that uses associated learning algorithm for classification and as well as regression.

Random Forest Algorithm

Random forest algorithm is being used for the stock market prediction. Since it has been termed as one of the easiest to use and flexible machine learning algorithm, it gives good accuracy in the prediction. This is usually used in the classification tasks. Because of the high volatility in the stock market, the task of predicting is quite challenging. In stock market prediction we are using random forest classifier which has the same hyperparameters as of a decision tree.The decision tool has a model similar to that of a tree. It takes the decision based on possible consequences, which includes variables like event outcome, resource cost, and utility. The random forest algorithm represents an algorithm where it randomly selects different observations and features to build several decision tree and then takes the aggregate of the several decision trees outcomes. The data is split into partitions based on the questions on a label or an attribute. The data set we used was from the previous year's stock markets collected from the public database available online, 80 % of data was used to train the machine and the rest 20 % to test the data. The basic approach of the supervised learning model is to learn the patterns and relationships in the data from the training set and then reproduce them for the test data.

Support Vector Machine Algorithm

The main task of the support machine algorithm is to identify an N-dimensional space that distinguishably categorizes the data points. Here, N stands for a number of features. Between two classes of data points, there can be multiple possible hyperplanes that can be chosen. The objective of this algorithm is to find a plane that has maximum margin. Maximizing margin refers to the distance between data points of both classes. The benefit associated with maximizing the margin is that it provides is that it provides some reinforcement so that future data points can be more easily classified. Decision boundaries that help classify data points are called hyperplanes. Based on the position of the data points relative to the hyperplane they are attributed to different classes. The dimension of the hyperplane relies on the number of attributes, if the number of attributes is two then the hyperplane is just a line, if the number of attributes is three then the hyperplane is two dimensional.

The tuning parameters of SVM classifier are kernel parameter, gamma parameter and regularization parameter.

- Kernels can be categorized as linear and polynomial kernels calculates the prediction line. In linear kernels prediction for a new input is calculated by the dot product between the input and the support vector.
- C parameter is known as the regularization parameter; it determines whether the accuracy of model is increases or decreases. The default value of c=10.Lower regularization value leads to misclassification.
- Gamma parameter measures the influence of a single training on the model. Low values signifies far from the plausible margin and high values signifies closeness from the plausible margin.

Method of Creation and Exploration

In this project we use four factors to predict the stock price direction - price fluctuations, price pressure, sector instability, and sectoral momentum.

Stock Market Forecasting Steps –

This step is important for download details from the site. We predict the market value of any stock. For the share amount until the closing date are download from the site.

In the next step the amount of data for any stock can converted to CSV file that it will easily load into the algorithm.

In the next step when the GUI is open and where we are click the SVM button will show the window from which it appears we select the share data file.

After selecting the stock data file from its folder we will show the Stock graph before mapping and stock after to make a map.

Next step algorithm calculated by log2c and log2g debugging. Therefore, it will predict a graph with a good amount of data.

In the final section algorithm show the predicted value the selected stock graph showing the actual value as well the predicted amount of stock.

Proposed System

In this proposed system, we focus on predicting the stock values using machine learning algorithms like Random Forest and Support Vector Machines. We proposed the system “Stock market price prediction” we have predicted the stock market price using the random forest algorithm. In this proposed system, we were able to train the machine from the various data points from the past to make a future prediction. We took data from the previous year stocks to train the model.

We majorly used two machine-learning libraries to solve the problem. The first one was numpy, which was used to clean and manipulate the data, and getting it into a form ready for analysis. The other was scikit, which was used for real analysis and prediction. The data set we used was from the previous years stock markets collected from the public database available online, 80 % of data was used to train the machine and the rest 20 % to test the data. The basic approach of the supervised learning model is to learn the patterns and relationships in the data from the training set and then reproduce them for the test data. We used the python pandas library for data processing which combined different datasets into a data frame. The tuned up data frame allowed us to prepare the data for feature extraction. Financial organizations and merchants have made different exclusive models to attempt and beat the market for themselves or their customers, yet once in a while has anybody accomplished reliably higher-than-normal degrees of profitability. Nevertheless, the challenge of stock forecasting is so engaging in light of the fact that the improvement of only a couple of rate focuses can build benefit by a large number of dollars for these organizations.

System Architecture

Kaggle is an online community for data analysis and predictive modeling. It also contains dataset of different fields, which is contributed by data miners. Various data scientist competes to create the best models for predicting and depicting the information. It allows the users to use their datasets so that they can build models and work with various data science engineers to solve various real-life data science challenges. The dataset used in the proposed project has been downloaded from Kaggle. However, this data set is present in what we call raw format. The data set is a collection of stock market information about a few companies. The first step is the conversion of this raw data into processed data. This is done using feature extraction, since in the raw data collected there are multiple attributes but only a few of those attributes are useful for the purpose of prediction. So the first step is feature extraction, where the key attributes are extracted from the whole list of attributes available in the raw dataset. Feature extraction starts from an initial state of measured data and builds derived values or features. These features are intended to be informative and non-redundant, facilitating the subsequent learning and generalization steps. Feature extraction is a dimensionality reduction process, where the initial set of raw variables is diminished to progressively reasonable features for ease of management, while still precisely and totally depicting the first informational collection. The first step is the conversion of this raw data into processed data. This is done using feature extraction, since in the raw data collected there are multiple attributes but only a few of those attributes are useful for the purpose of prediction. So the first step is feature extraction, where the key attributes are extracted from the whole list of attributes available in raw dataset.

Module Description

The various modules of the project would be divided into the segments as described.

Data Collection - Data collection is a very basic module and the initial step towards the project. It generally deals with the collection of the right dataset. The dataset that is to be used in the market prediction has to be used to be filtered based on various aspects. Data collection also complements to enhance the dataset by adding more data that are external. Our data mainly consists of the previous year stock prices. Initially, we will be analyzing the Kaggle dataset and according to the accuracy, we will be using the model with the data to analyze the predictions accurately.

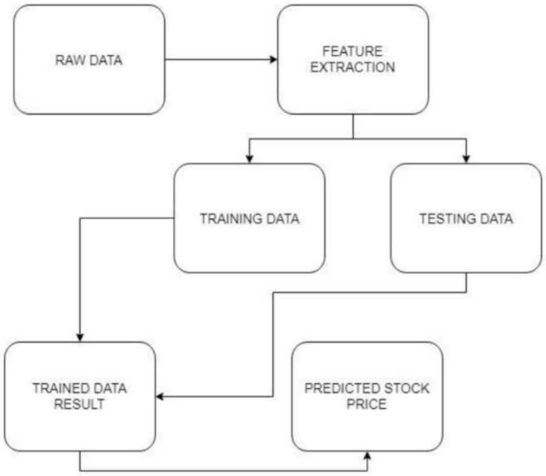
Pre Processing - Data pre-processing is a part of data mining, which involves transforming raw data into a more coherent format. Raw data is usually, inconsistent or incomplete and usually contains many errors. The data pre-processing involves checking out for missing values, looking for categorical values, splitting the data-set into training and test set and finally do a feature scaling to limit the range of variables so that they can be compared on common environs.

Training Machine - Training the machine is similar to feeding the data to the algorithm to touch up the test data. The training sets are used to tune and fit the models. The test sets are untouched, as a model should not be judged based on unseen data. The training of the model includes cross-validation where we get a well-grounded approximate performance of the model using the training data. Tuning models are meant to specifically tune the hyperparameters like the number of trees in a random forest. We perform the entire cross-validation loop on each set of hyperparameter values. Finally, we will calculate a cross-validated score, for individual sets of hyperparameters. Then, we select the best hyperparameters. The idea behind the training of the model is that we some initial values with the dataset and then optimize the parameters which we want to in the model. This is kept on repetition until we get the optimal values. Thus, we take the predictions from the trained model on the inputs from the test dataset. Hence, it is divided in the ratio of 80:20 where 80% is for the training set and the rest 20% for a testing set of the data.

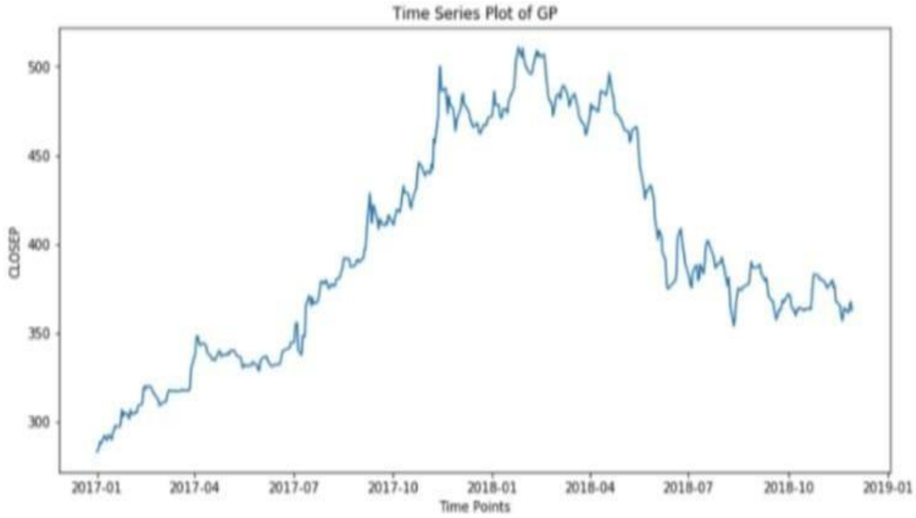
Data Scoring - The process of applying a predictive model to a set of data is referred to as scoring the data. The technique used to process the dataset is the Random Forest Algorithm. Random forest involves an ensemble method, which is usually used, for classification and as well as regression. Based on the learning models, we achieve interesting results. The last module thus describes how the result of the model can help to predict the probability of a stock to rise and sink based on certain parameters. It also shows the vulnerabilities of a particular stock or entity. The user authentication system control is implemented to make sure that only the authorized entities are accessing the results.

Results –

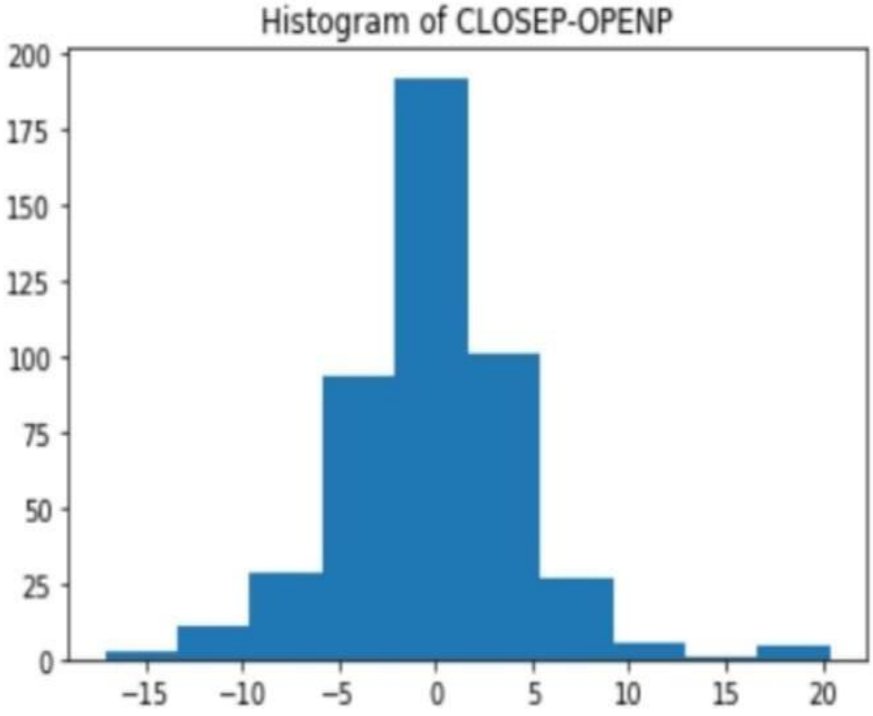
System Architecture



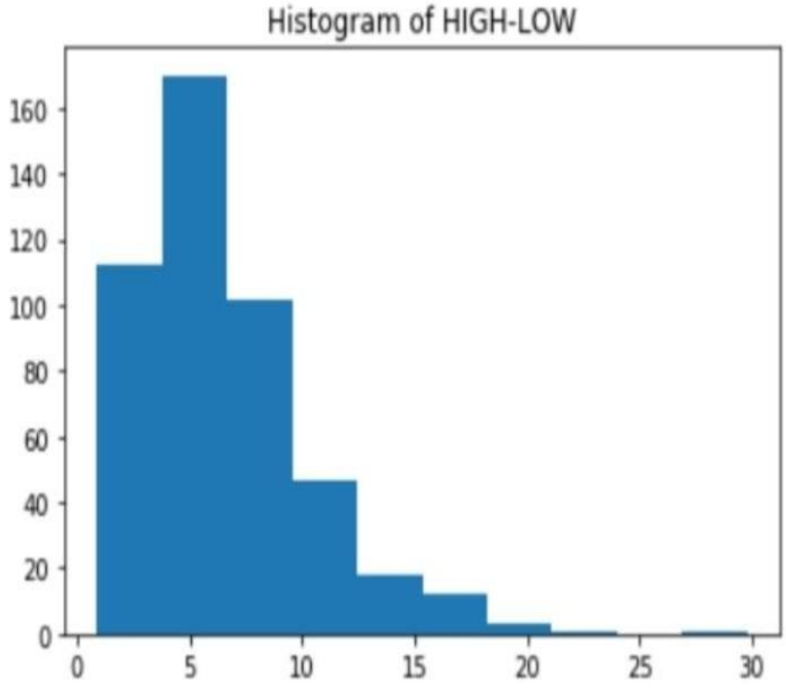
Time series plot of GP



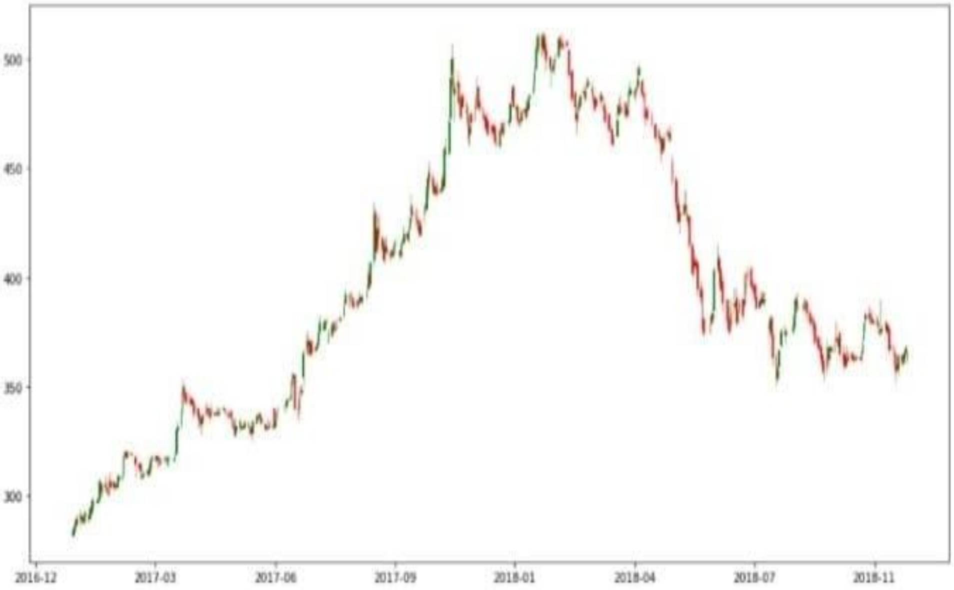
Histogram of CLOSEPOPENP



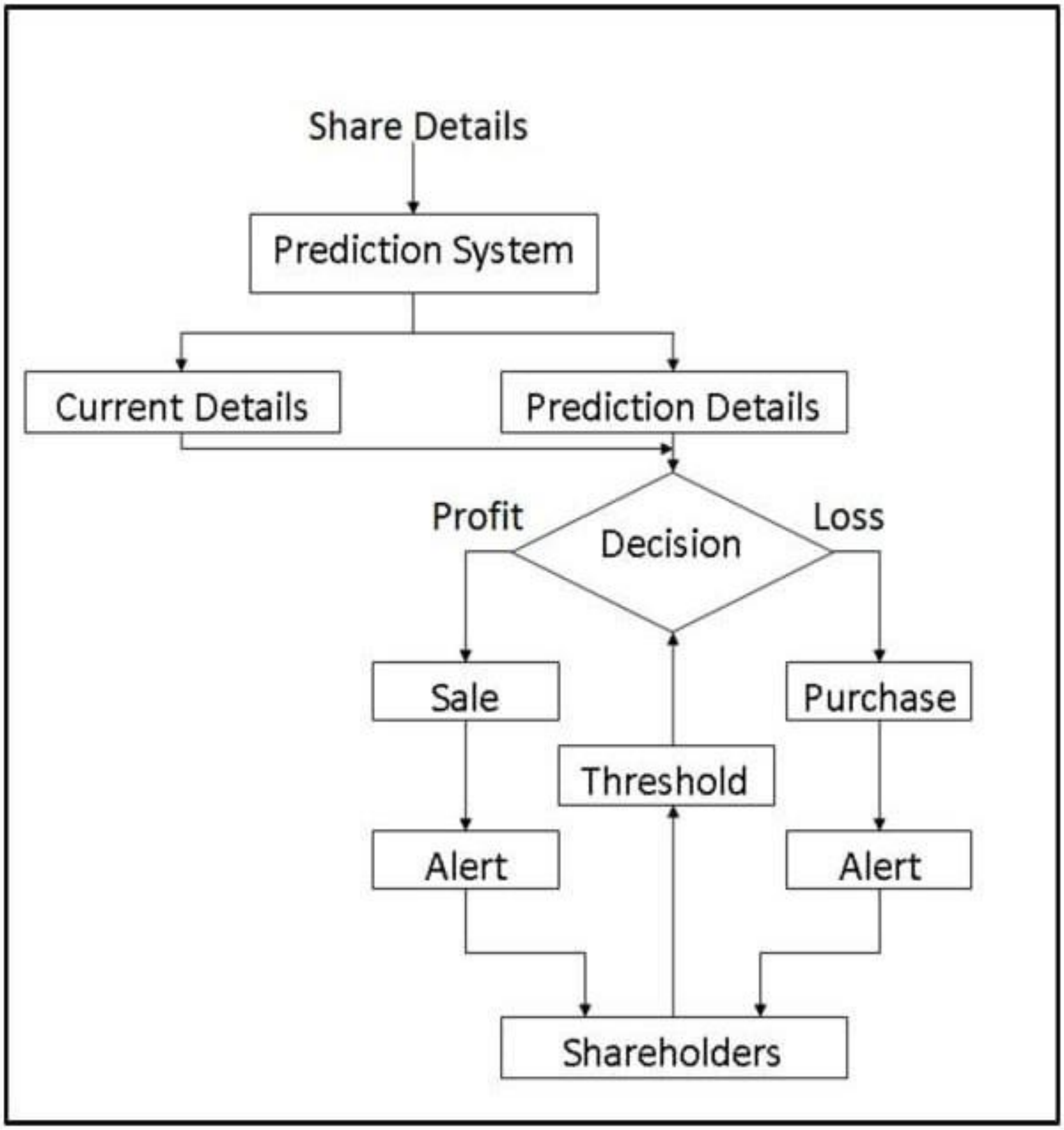
Histogram of HIGH-LOW

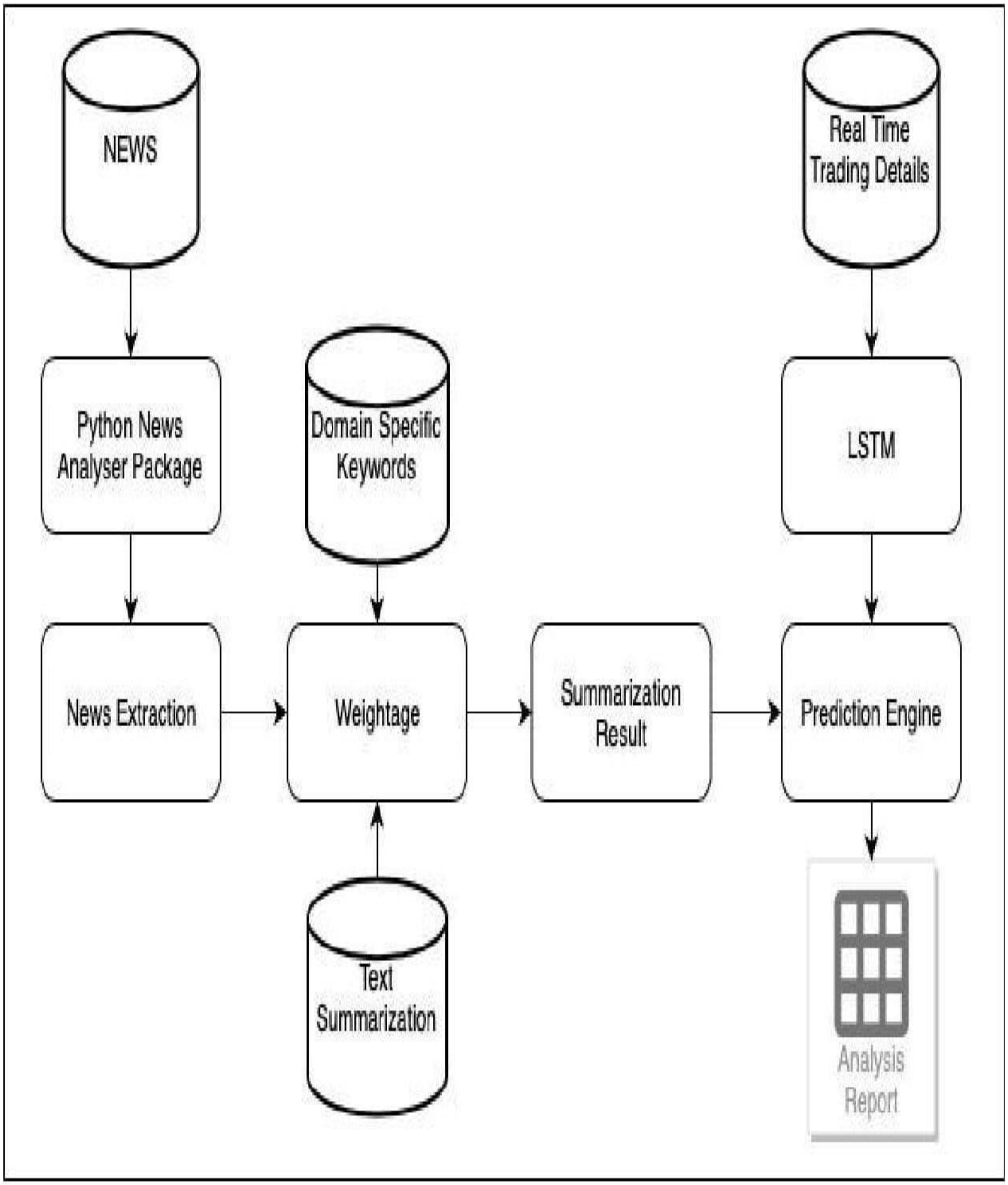


Candlestick plot



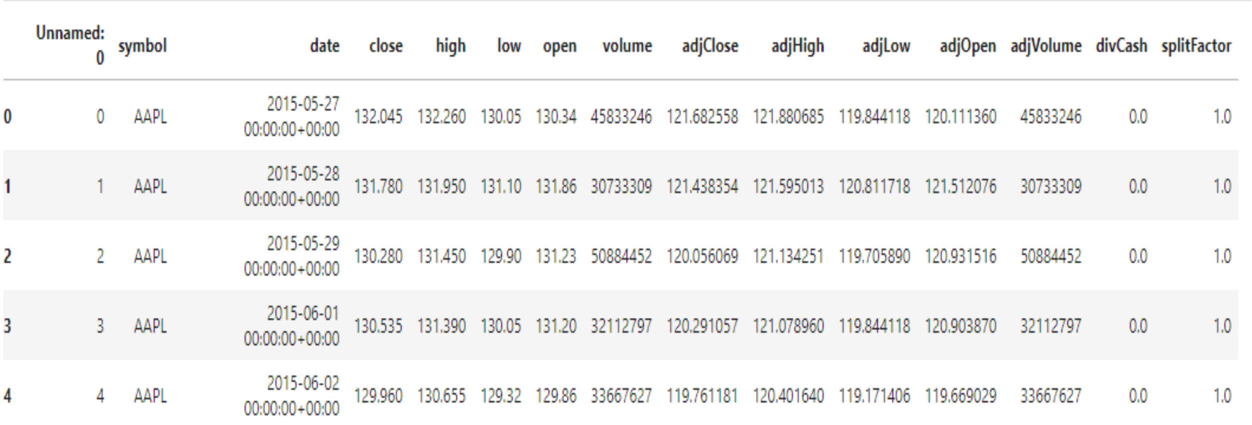
Project Design




NEWS
Real Time Trading Details
Python News Analyser Package
Domain Specific Keywords
LSTM
News Extraction
Weightage
Summarization Result
Prediction Engine
Text Summarization
Analysis Report

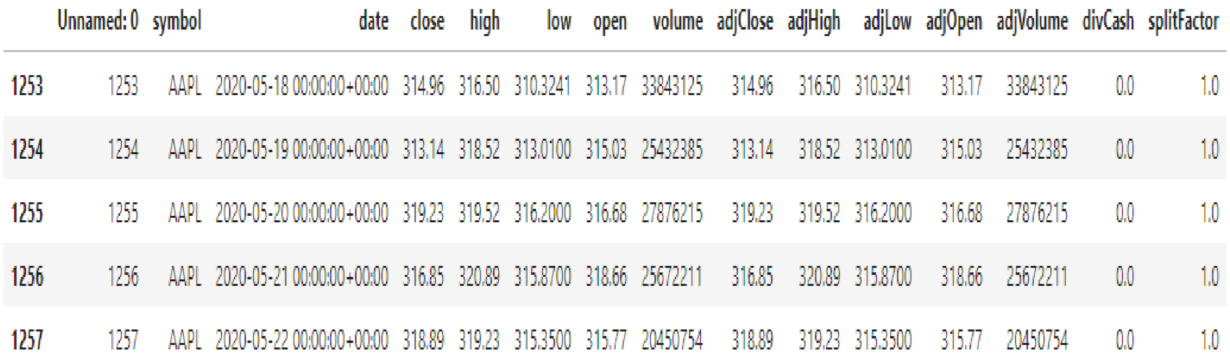
Code –

```
import pandas_datareader as pdr
key=""
df = pdr.get_data_tiingo('AAPL', api_key=key)
df.to_csv('AAPL.csv')
import pandas as pd
df=pd.read_csv('AAPL.csv')
df.head()
```



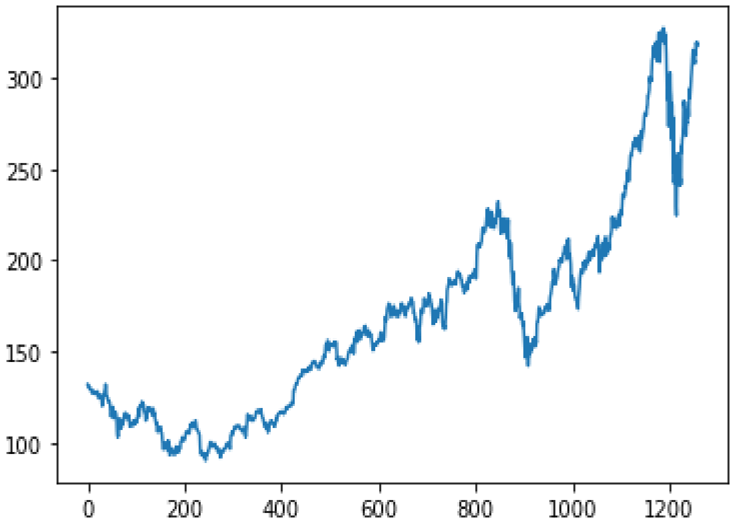
	Unnamed: 0	symbol	date	close	high	low	open	volume	adjClose	adjHigh	adjLow	adjOpen	adjVolume	divCash	splitFactor
0	0	AAPL	2015-05-27 00:00:00+00:00	132.045	132.260	130.05	130.34	45833246	121.682558	121.880685	119.844118	120.111360	45833246	0.0	1.0
1	1	AAPL	2015-05-28 00:00:00+00:00	131.780	131.950	131.10	131.86	30733309	121.438354	121.595013	120.811718	121.512076	30733309	0.0	1.0
2	2	AAPL	2015-05-29 00:00:00+00:00	130.280	131.450	129.90	131.23	50884452	120.056069	121.134251	119.705890	120.931516	50884452	0.0	1.0
3	3	AAPL	2015-06-01 00:00:00+00:00	130.535	131.390	130.05	131.20	32112797	120.291057	121.078960	119.844118	120.903870	32112797	0.0	1.0
4	4	AAPL	2015-06-02 00:00:00+00:00	129.960	130.655	129.32	129.86	33667627	119.761181	120.401640	119.171406	119.669029	33667627	0.0	1.0

```
df.tail()
```



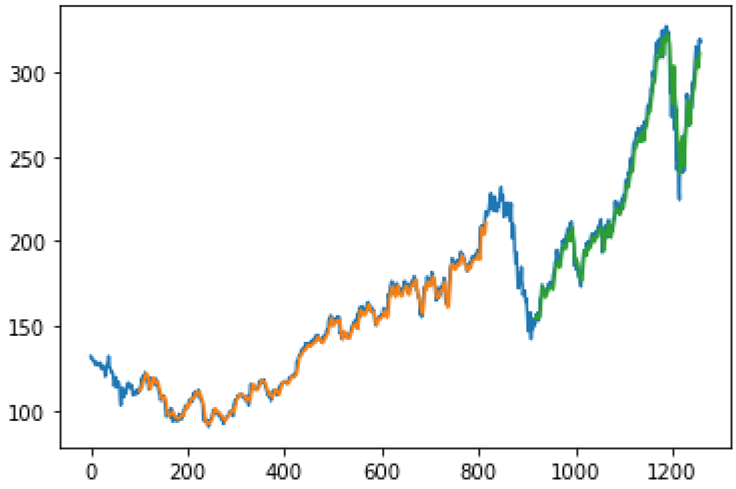
	Unnamed: 0	symbol	date	close	high	low	open	volume	adjClose	adjHigh	adjLow	adjOpen	adjVolume	divCash	splitFactor
1253	1253	AAPL	2020-05-18 00:00:00+00:00	314.96	316.50	310.3241	313.17	33843125	314.96	316.50	310.3241	313.17	33843125	0.0	1.0
1254	1254	AAPL	2020-05-19 00:00:00+00:00	313.14	318.52	313.0100	315.03	25432385	313.14	318.52	313.0100	315.03	25432385	0.0	1.0
1255	1255	AAPL	2020-05-20 00:00:00+00:00	319.23	319.52	316.2000	316.68	27876215	319.23	319.52	316.2000	316.68	27876215	0.0	1.0
1256	1256	AAPL	2020-05-21 00:00:00+00:00	316.85	320.89	315.8700	318.66	25672211	316.85	320.89	315.8700	318.66	25672211	0.0	1.0
1257	1257	AAPL	2020-05-22 00:00:00+00:00	318.89	319.23	315.3500	315.77	20450754	318.89	319.23	315.3500	315.77	20450754	0.0	1.0

```
import matplotlib.pyplot as plt
plt.plot(df1)
```

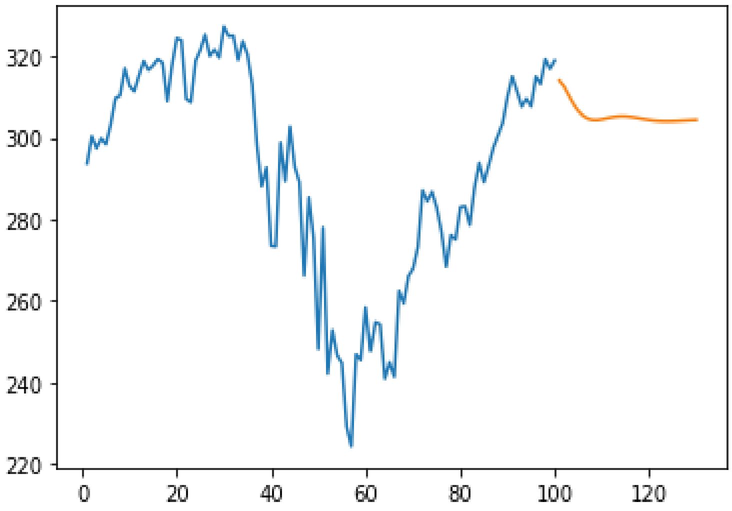


```
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense
from tensorflow.keras.layers import LSTM
model=Sequential()
model.add(LSTM(50,return_sequences=True,input_shape=(100,1)))
model.add(LSTM(50,return_sequences=True))
model.add(LSTM(50))
model.add(Dense(1))
model.compile(loss='mean_squared_error',optimizer='adam')
model.summary()
train_predict=model.predict(X_train)
test_predict=model.predict(X_test)
train_predict=scaler.inverse_transform(train_predict)
test_predict=scaler.inverse_transform(test_predict)
import math
from sklearn.metrics import mean_squared_error
math.sqrt(mean_squared_error(y_train,train_predict))
math.sqrt(mean_squared_error(ytest,test_predict))
```

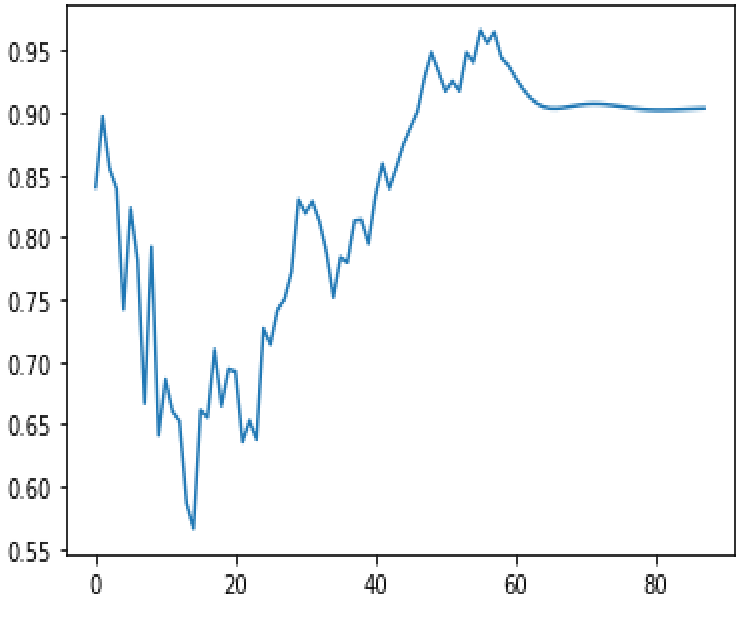
```
look_back=100
trainPredictPlot = numpy.empty_like(df1)
trainPredictPlot[:, :] = np.nan
trainPredictPlot[look_back:len(train_predict)+look_back, :] = train_predict
testPredictPlot = numpy.empty_like(df1)
testPredictPlot[:, :] = numpy.nan
testPredictPlot[len(train_predict)+(look_back*2)+1:len(df1)-1, :] = test_predict
plt.plot(scaler.inverse_transform(df1))
plt.plot(trainPredictPlot)
plt.plot(testPredictPlot)
plt.show()
```



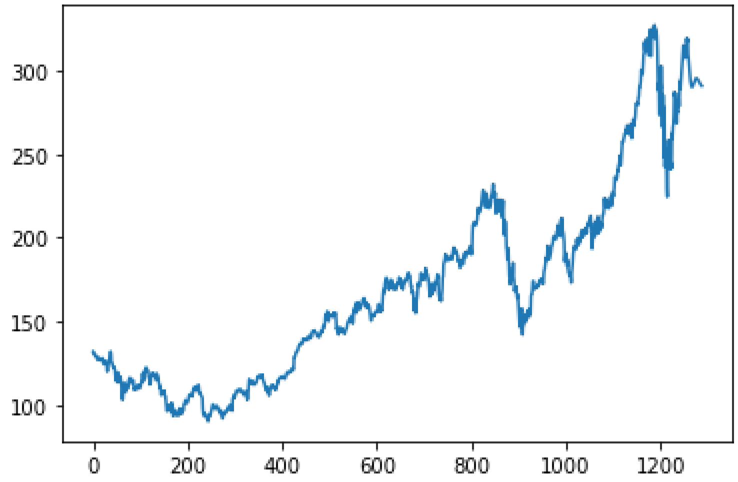
```
day_new=np.arange(1,101)
day_pred=np.arange(101,131)
import matplotlib.pyplot as plt
len(df1)
plt.plot(day_new,scaler.inverse_transform(df1[1158:]))
plt.plot(day_pred,scaler.inverse_transform(lst_output))
```



```
df3=df1.tolist()
df3.extend(lst_output)
plt.plot(df3[1200:])
```



```
df3=scaler.inverse_transform(df3).tolist()
plt.plot(df3)
```



Conclusion

In the project, we suggested the use of collected data from various world markets for electronic finance read algorithms to predict stock index movement. The SVM algorithm works on large amounts of data collected in various global financial markets.

Also, SVM does not present the problem of over equality. Variety machine-based models are proposed for prediction the daily trend of stocks in the Market. Many results suggest high efficiency.

Functional trading models are built on them our well-trained forecaster. The model produces higher profit compared to selected benchmarks. By measuring the accuracy of the different algorithms, we found that the most suitable algorithm for predicting the market price of a stock based on various data points from the historical data is the random forest algorithm. The algorithm will be a great asset for brokers and investors for investing money in the stock market since it is trained on a huge collection of historical data and has been chosen after being tested on a sample data. The project demonstrates the machine learning model to predict the stock value with more accuracy as compared to previously implemented machine learning models.

Reference

1-GFG Machine Learning

2-“Impact Of Financial Ratios And Technical Analysis On Stock Price Prediction Using Random Forests”, IEEE.

3-“Stock Market Prediction via Multi-Source Multiple Instance Learning.”

4-Zhen Hu, Jibe Zhu, and Ken Tse “Stocks Market Prediction Using Support Vector Machine”, 6th International Conference on Information Management, Innovation Management and Industrial Engineering, 2013.M.

5-Wei Huang, Yoshiteru Nakamori, Shou-Yang Wang, “Forecasting stock market movement direction with support vector machine”, Computers & Operations Research, Volume 32, Issue 10, October 2005, Pages 2513–2522.

6-N. Ancona, Classification Properties of Support Vector Machines for Regression, Technical Report, RIIESI/CNRNr. 02/99.

7-K. jae Kim, “Financial time series forecasting using support vector machines,” Neurocomputing, vol. 55, 2003