STOCK PREDICTION MODEL

BACHELOR OF ENGINEERING

IN

CSE (H) IN BIG DATA ANALYTICS

Submitted By

Daksh Rawal(19BCS3849)

Aryan Gupta(19BCS3828)

Prajwal(19BCS3831)

Under The Supervision Of

Dr. Rahul Rastogi

CHANDIGARH UNIVERSITY, GHARUAN, MOHALI-140413,

PUNJAB

MAY-2022

PROBLEM FORMULATION

The financial market is a dynamic and composite system where people can buy and sell currencies, stocks, equities and derivatives over virtual platforms supported by brokers. The stock market allows investors to own shares of public companies through trading either by exchange or over the counter markets. This market has given investors the chance of gaining money and having a prosperous life through investing small initial amounts of money, low risk compared to the risk of opening new business or the need of high salary career. Stock markets are affected by many factors causing the uncertainty and high volatility in the market. Although humans can take orders and submit them to the market, automated trading systems (ATS) that are operated by the implementation of computer programs can perform better and with higher momentum in submitting orders than any human. However, to evaluate and control the performance of ATSs, the implementation of risk strategies and safety measures applied based on human judgements are required. Many factors are incorporated and considered when developing an ATS, for instance, trading strategy to be adopted, complex mathematical functions that reflect the state of a specific stock, machine learning algorithms that enable the prediction of the future stock value, and specific news related to the stock being analysed. So, the goal is to create a web application to predict stock prices of these shares .

Motivation For Work

Businesses primarily run over customer's satisfaction, customer reviews about their products. Shifts in sentiment on social media have been shown to correlate with shifts in stock markets. Identifying customer grievances thereby resolving them leads to customer satisfaction as well as trustworthiness of an organization. Hence there is a necessity of an un biased automated system to classify customer reviews regarding any problem. In today's environment where we're justifiably suffering from data overload (although this does not mean better or deeper insights), companies might have mountains of customer feedback collected; but for mere humans, it's still impossible to analyse it manually without any sort of error or bias. Oftentimes, companies with the best intentions find themselves in an insights vacuum. You know you need insights to inform your decision making and you know that you're lacking them, but don't know how best to get them. Sentiment analysis provides some answers into what the most important issues are, from the perspective of customers, at least. Because sentiment analysis can be automated, decisions can be made based on a significant amount of data rather than plain intuition.

METHODOLOGY

Random forest algorithm have a set of rules is used for characteristic extraction. Random forests or random selection forests are an ensemble gaining knowledge of technique for category, regression and different duties that perform with the aid of using building a mess of selection bushes at training time and outputting the magnificence this is the mode of the instructions for category or imply prediction for regression of the person bushes.

Algorithm 1: Feature Extraction Procedure
Input: Data set as CSV file.
Output: Selected the important attribute Listed.

1: Read the dataset.
2: Import RandomForestClassifier from sklearn.ensemble.
3: Assign the RandomForestClassifier to local variable model.
4: Train Rfc =(nestimators=100,randomstate=0,njobs=-1).
5: Create clf =sfm(clf,threshold=0.15)
6: assign sfm to clf
7: get the important attribute

Using the random forest algorithm the data has been split.They split the data for training and testing using the cross validation.The are fit in Random forest algorithm and split using cross validation. The split the train data in 70 percentage and test in 30 percentage.

Algorithm 2: Data Train, Test Split Procedure
1: Read the dataset.
2: Import RandomForestClassifier from sklearn.ensemble.
3: Create Xtest,Xtrain,Ytest,Ytrain.
4: Create featable.
5: Assign Date,open,close inside featlable.
6: Assign traintestsplit(dfx, dfy, testsize=0.2, rs=0).
7: X,Y are fit using Randomforestclassifier.
8: create clf variable and fit randomforest in that variable
9: for feature in featlable do
10: Print feature
11: get the train data and test data

This algorithm is used create the independent data set X and store the data in the variable dates. Create the dependent data set y and store the data in the variable prices. Both can be done by appending the data to each of the lists. The independent data set we want only the day from the date, so use the split function to get just the day and cast it to an integer while appending the data to the dates list.

Support Vector Regression is used predict the result, using SVR train the dataset to get the accuracy of the prediction and linear regression is also used to approach to modelling the relationship between a scalar response or dependent variable and one or more explanatory variables or independent variables.

Algorithm 3 Ticker Data Processing Procedure
1: Read the dataset
2: Create the list dates and prices
3: for date in dates do
4: dates append to date.split[o].
5: for prices in open do
6: prices append to open
7: Print the dates
8: Print the prices

Support Vector Regression SVR models with three different kernels to see which one performs the best. The function will ave three parameters, the dates, prices, and the day

that we want to do the prediction on to get the price. first I will create the three SVR models with three different kernels are linear, polynomial, radial basis function. Also add in the linear regression model.

Algorithm 4 Algorithm Evaluation Procedure
Input: The trained dataset
Output: The predicted open price for the day as the result

1: Read the dataset
2: import SVR from sklearn.SVM
3: import matplotlib.pyplot
4: Create the linear kernel
5: Create the polynomial kernel
6: Create the rbf kernel
7: Train the linear in dates,prices
8: Train the polynomial in dates,prices
9: Train the rbf in dates ,prices
10: Create the linear regression
11: Train the linear regression
12: plot the days in Xlabel
13: plot the price in Ylabel
14: plot dates and prices in poly and linear and rbf
15: Return rbf predicted result

TASK ASSIGNED

Daksh Rawal – Collected the datasets, Read research papers on Data Science, Business applications, Stock prediction, Share markets.

Prajwal – Read research Papers on Machine Learning, Different Machine learning techniques used in Stock Prediction.

Aryan Gupta – Work with the web application building, Learn frontend development, Learn Streamlit and other Libraries of python which are required for the development of the project.

REFERENCES

[1] Hu Z, Zhu J, Tse K. Stocks market prediction using support vector machine. In 2013 6th International Conference on Information Management, Innovation Management and Industrial Engineering 2013 Nov 23 (Vol. 2, pp. 115-118). IEEE.
[2] Usmani M, Adil SH, Raza K, Ali SS. Stock market prediction using machine learning techniques. In 2016 3rd international conference on computer and information sciences (ICCOINS) 2016 Aug 15 (pp. 322-327). IEEE.
[3] Mankar T, Hotchandani T, Madhwani M, Chidrawar A, Lifna CS. Stock market prediction based on social sentiments using machine learning. In 2018 International Conference on Smart City and Emerging Technology (ICSCET) 2018 Jan 5 (pp. 1-3). IEEE. Reshma R et al. / Stock Market

[4] Ouahilal M, El Mohajir M, Chahhou M, El Mohajir BE. Optimizing stock market price prediction using a hybrid approach based on HP filter and support vector regression. In 2016 4th IEEE International Colloquium on Information Science and Technology (CiSt) 2016 Oct 24 (pp. 290-294). IEEE.
[5] Huang H, Zhang W, Deng G, Chen J. Predicting stock trend using fourier transform and support vector regression. In 2014 IEEE 17th International Conference on Computational Science and Engineering 2014 Dec 19 (pp. 213-216). IEEE.