PREDICTION ANALYSIS TECHNIQUES USING MACHINE LEARNING ALGORITHM FOR MINING OF WEB DATA

[1]Parul Saini, [2]Anshu Tiwari, [3]Manish Saxena

[1]Student, [2] Asst.Prof., [3] Asst.Prof.

[1]Computer Science Engineering , BIST, Bhopal. M.P.

Abstract: The world web data trading is one among the most important activities. Web data prediction is an act of trying to work out the longer term value of a web other financial instrument traded on a financial exchange. This explains the prediction of a web using Machine Learning. The technical and fundamental or the statistic analysis is employed by the foremost of the stockbrokers while making the web predictions. During this we propose a Machine Learning (ML) approach which will be trained from the available data and gain intelligence then uses the acquired knowledge for an accurate prediction. In these we've taken three algorithm like KNN (Supervised Algorithm) and Neural Network (Unsupervised Algorithm) and from the summary of model performance parameters, we will see that Unsupervised Algorithm performs better as compared to baseline supervised algorithms.

Index Terms **- Web Data, Machine Learning, Predictions, Classification, forecasting, data mining, web data Forecasting, KNN, Neural Network**.

I. INTRODUCTION

Investment firms, hedge funds and even individuals have been using financial models to better understand market behavior and make profitable investments and trades. A wealth of information is available in the form of historical web data and company performance data, suitable for machine learning algorithms to process. Can we actually predict data with machine learning? Investors make educated guesses by analyzing data. They'll read the news, study the company history, industry trends and other lots of data points that go into making a prediction. The prevailing theories are that web data are totally random and unpredictable but that raises the question why top firms like Morgan Stanley and Citigroup hire quantitative analysts to build predictive models. We have this idea of a trading floor being filled with adrenaline infuse men with loose ties running around yelling something into a phone but these days they're more likely to see rows of machine learning experts quietly sitting in front of computer screens. In fact about 75% of all orders on Wall Street are now placed by software, we're now living in the age of the algorithm.

Web data prediction is basically defined as trying to determine the web value and offer a robust idea for the people to know and predict the market and the web data 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 web trends.

The problem with estimating the webdata will remain a problem if a better web prediction algorithm is not proposed. Predicting how the webdata will perform is quite difficult. The movement is the usually determined by the sentiments of thousands of investors. Web data 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 hyper parameters. All these factors make webdata prediction very difficult. Once the right data is collected, it then can be used to train a machine and to generate a predictive result.

MACHINE LEARNING

When a computer needs to perform a certain task, a programmer's solution is to write a computer program that performs the task. A computer program is a piece of code that instructs the computer which actions to take in order to perform the task. The field of machine learning is concerned with the higher-level question of how to construct computer programs that automatically learn with experience. A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks in T, as measured by P, improves with experience E. Thus, machine learning algorithms automatically extract knowledge from machine readable information. In machine learning, computer algorithms (learners) attempt to automatically distill knowledge from example data. This knowledge can be used to make predictions about novel data in the future and to provide insight into the nature of the target concepts. Applied to the research at hand, this means that a computer

would learn to classify alerts into incidents and non-incidents (task T). A possible performance measure (P) for this task would be the accuracy with which the machine learning program classifies the instances correctly. The training experiences (E) could be labeled instances.

MACHINE LEARNING BENEFITS

In particular, machine learning plays an essential role in the following three areas of software engineering:

1. Data mining problems where large databases may contain valuable implicit regularities that can be discovered automatically.
2. Difficult-to-program applications, which are too difficult for traditional manual programming.
3. Software applications that customize to the individual user's preferences, such as personalized advertising.

There are several reasons why machine-learning plays a role in these three domains. First of all, for the classification of security incidents, a vast amount of data has to be analyzed containing historical data. It is difficult for human beings to find a pattern in such an enormous amount of data. Machine-learning, however, seems well-suited to overcome this problem and can therefore be used to discover those patterns. With respect to the difficult-to-program applications, an analyst's knowledge is often implicit, and the environments are dynamic.

II. LITERATURE REVIEW

According to Srinath Ravikumar, et. al., 2020, The webdata is an interesting industry to study. There are various variations present in it. Many experts have been studying and researching on the various trends that the webdata goes through. One of the major studies has been the attempt to predict the webdata of various companies based on historical data. Prediction of webdata will greatly help people to understand where and how to invest so that the risk of losing money is minimized. This application can also be used by companies during their Initial Public Offering (IPO) to know what value to target for and how many shares they should release. So far there have been significant developments in this field. Many researchers are looking at machine learning and deep learning as possible ways to predict webdata. The proposed system works in two methods – Regression and Classification. In regression, the system predicts the closing data of web of a company, and in classification, the system predicts whether the closing data of web will increase or decrease the next day.

Kunal Pahwa, et. al, 2019, Web data or Share market is one of the most complicated and sophisticated way to do business. Small ownerships, brokerage corporations, banking sector, all depend on this very body to make revenue and divide risks; a very complicated model. However, this paper proposes to use machine learning algorithm to predict the future webdata for exchange by using open source libraries and preexisting algorithms to help make this unpredictable format of business a little more predictable. We shall see how this simple implementation will bring acceptable results. The outcome is completely based on numbers and assumes a lot of axioms that may or may not follow in the real world so as the time of prediction.

Indronil Bhattacharjee, et. al.,2019, Webdata is one of the most important sectors of a country's economy. Prediction of webdatas is not easy since it is not stationary in nature. The objective of this paper is to find the best possible method to predict the closing datas of webdata through a comparative study between different traditional statistical approaches and machine learning techniques. Predictions using statistical methods like Simple Moving Average, Weighted Moving Average, Exponential Smoothing, Naive approach, and machine learning methods like Linear Regression, Lasso, Ridge, K-Nearest Neighbors, Support Vector Machine, Random Forest, Single Layer Perceptron, Multi-layer Perceptron, Long Short Term Memory are performed. Moreover, a comparative study between statistical approaches and machine learning approaches has been done in terms of prediction performances and accuracy. After studying all the methods individually, the machine learning approach, especially the neural network models are found to be the most accurate for webdata prediction.

Pawee Werawithayaset, et. al.,2018, this research was prepared to predict the closing data of the webin the WebExchange of Thailand (SET). We are using the Multi-Layer Perceptron model, Support Vector Machine model, and Partial Least Square Classifier to predict the closing data of the stock. In the present, people have more knowledge and understanding of investing in the webdata then the Thai webdata has grown significantly. From the statistical data, we can find the movement of webdatas in that webdata moves in a cycle. Form this point; we have the idea that if we can predict the webdata nearby real data. We can be investing at the right time and help investors to reduce investment risks. The experimental result shows that Partial Least Square is the best algorithm of the three algorithms to predict the web closing data.

III. PROBLEM DEFINITION

Over the last two decades, humans have grown a lot of dependence on data and information in society and with this advent growth, technologies have evolved for their storage, analysis and processing on a huge scale. The fields of Data Mining and Machine Learning have not only exploited them for knowledge and discovery but also to explore certain hidden patterns and concepts which led to the prediction of future events, not easy to obtain. And one of the difficult things to predict that caught our attention is webor commonly called as shares. Webdata prediction is one of the most important topics to be investigated in

academic and financial researches. Various Data mining techniques are frequently involved in the studies. To solve this problem. But technique using machine learning/deep learning will give more accurate, precise and simple way to solve such issues related to weband market datas. There are ample amounts of data about webdata but the most difficult and intriguing thing is to predict the data of these webdata based on old data.

.

IV. PROPOSED WORK

In the finance world webtrading is one of the most important activities. Webdata prediction is an act of trying to determine the future value of a webother financial instrument traded on a financial exchange. This explains the prediction of a webusing Machine Learning. The technical and fundamental or the time series analysis is used by the most of the stockbrokers while making the web predictions. In this paper we propose a Machine Learning (ML) approach that will be trained from the available webdata data and gain intelligence and then uses the acquired knowledge for an accurate prediction.

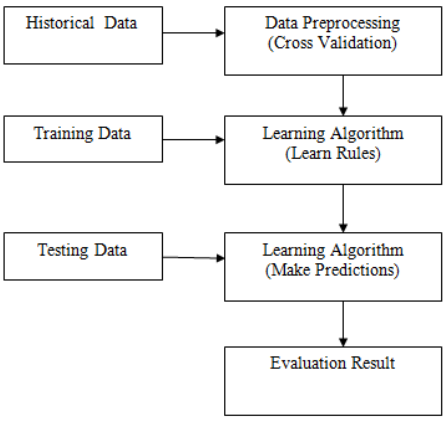


Figure 1. Flow Diagram

1. Dataset:- we first collect the historical webdata set from publicly available datasets.
2. Data preprocessing: It is the most important phase in prediction models as the data consists of ambiguities, errors, redundancy which needs to be cleaned beforehand. The data gathered from multiple sources first is aggregated and then cleaned as the complete data collected is not suitable for modeling purposes. The records with unique values do not have any significance as they do not contribute much in predictive modeling. Fields with too many null values also need to be discarded.
3. Learn ML Model: After preprocessing we can learn algorithm on training data and make prediction on test dataset.
4. Obtain Result: Based on Prediction we can calculate the performance of the algorithm.

Initialize weights W and biases b of the neural network N with random values
do
  for each training example (x_i, y_i)
    p_i = neural-network-prediction (N, x_i)
    calculate gradients of loss function (p_i, y_i) with respect to w^2 at layer L_3
    get Δw^2 for all weights from hidden layer L_2 to output layer L_3
    calculate gradient with respect to w^1 by chain rule at layer L_2
    get Δw^1 for all weights from input layer L_1 to hidden layer L_2
    update (w^1, w^2)
until all training examples are classified correctly or other stopping criteria are met
return the trained neural network

V. EXPERIMENTAL & RESULT ANALYSIS

This algorithm is a process that seeks to predict future values based on the past and present data. This historical data points are extracted and prepared trying to predict future values for a selected variable of the dataset. During market history there have been a continuous interest trying to analyse its tendencies, behavior and random reactions. This continuous concern to understand what happens before it really happens motivates us to continue with this study. The goal of this project is to predict the future webdata of Google using various predictive forecasting models and then analyzing the various models. The dataset for Google webdata is obtained from Yahoo Finance using Quantmod package in R.

We obtain the data of from of Google Webdata for our analysis using the quantmod package. The final datasets can be found below in an interactive table which is shown in figure 2.

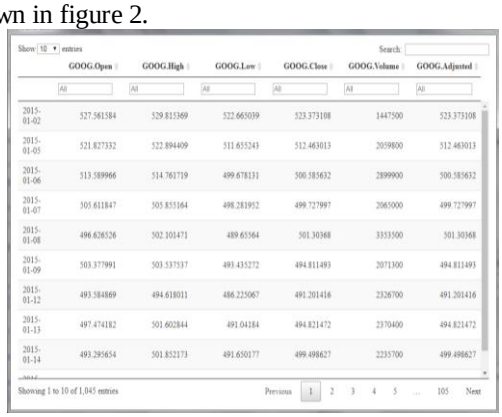


Figure 2. Dataset Preview

Summary of variables

Variable	Class	Description
GOOG.Open	num	Opening data of the webon the day
GOOG.High	num	Highest data of the webon the day
GOOG.Low	num	Lowest data of the webon the day
GOOG.Close	num	Closing data of the webon the day
GOOG.Volume	num	Total Volume Traded
GOOG.Adjusted	num	Adjusted data of the webincluding any risks or strategies

We can build the new model and we can trained the model and perform various test as mention above and the performance result we have got are shown in figure 3.

```
> modelfit <- auto.arima(tsData, lambda = "auto")
> summary(modelfit)
Series: tsData
ARIMA(2,1,0) with drift
Box Cox transformation: lambda= -0.263658

Coefficients:
         ar1      ar2  drift
      0.0456  -0.0416  1e-04
s.e.  0.0309   0.0310  1e-04

sigma^2 estimated as 6.652e-06:  log likelihood=4743.4
AIC=-9478.79   AICc=-9478.75   BIC=-9458.99

Training set error measures:
                      ME     RMSE      MAE        MPE     MAPE     MASE        ACF1
Training set -0.07075168 13.08155 8.81548 -0.0116473 1.028236 1.000694 -0.02713286
>
```

Figure 3. Performance Evaluation of model

KNN Regression Time Series Forecasting Model

KNN model can be used for both classification and regression problems. The most popular application is to use it for classification problems. Now with the tsfknn package KNN can be implemented on any regression task. The idea of this study is illustrating the different forecasting tools, comparing them and analysing the behavior of predictions. Following our KNN study, we proposed it can be used for both classification and regression problems. For predicting values of new data points, the model uses 'feature similarity', assigning a new point to a values based on how close it resembles the points on the training set. The first task is to determine the value of k in our KNN Model. The general rule of thumb for selecting the value of k is taking the square root of the number of data points in the sample. Hence, for the data set we take k = 32, and the forecasting prediction are shown in figure 4.

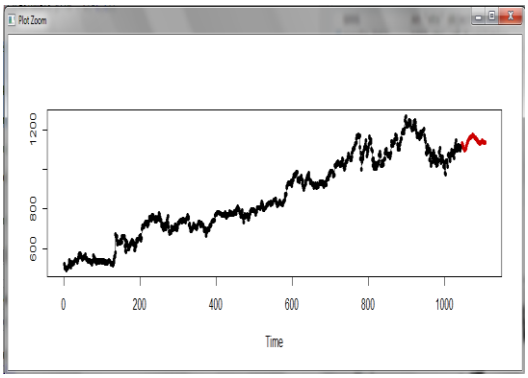


Figure 4 KNN Regression Time Series Forecasting

Then evaluate the KNN model for our forecasting time series which is shown in figure 5.

```
 [6,] -34.42499 -34.46307       NA       NA       NA       NA
 [7,] -30.62807       NA       NA       NA       NA       NA
 [8,]       NA       NA       NA       NA       NA       NA
 [9,]       NA       NA       NA       NA       NA       NA
[10,]       NA       NA       NA       NA       NA       NA
[11,]       NA       NA       NA       NA       NA       NA
[12,]       NA       NA       NA       NA       NA       NA
[13,]       NA       NA       NA       NA       NA       NA
[14,]       NA       NA       NA       NA       NA       NA
[15,]       NA       NA       NA       NA       NA       NA
[16,]       NA       NA       NA       NA       NA       NA
 [ reached getOption("max.print") -- omitted 45 rows ]

$global_accu
     RMSE       MAE      MAPE
44.046959 33.780280  3.170659
```

Figure 5. Evaluate of the KNN model

Feed Forward Neural Network Modelling

The next model which we would try and implement is a forecasting model with neural networks. In this model, we are using single hidden layer form where there is only one layer of input nodes that send weighted inputs to a subsequent layer of receiving nodes. The nnetar function in the forecast package fits a single hidden layer neural network model to a timeseries. The function model approach is to use lagged values of the time series as input data, reaching to a non-linear autoregressive model.

The first step is to determine the number of hidden layers for our neural network. Although, there is no specific method for calculating the number of hidden layers, the most common approach followed for time series forecasting is by calculating is using the formula:

N(hidden)=Ns/(a∗ (Ni+No))

where Ns: Number of train samples Ni: Number of input neurons No: Number of output neurons a : 1.5^-10

and the forecasting result are shown in figure 6.

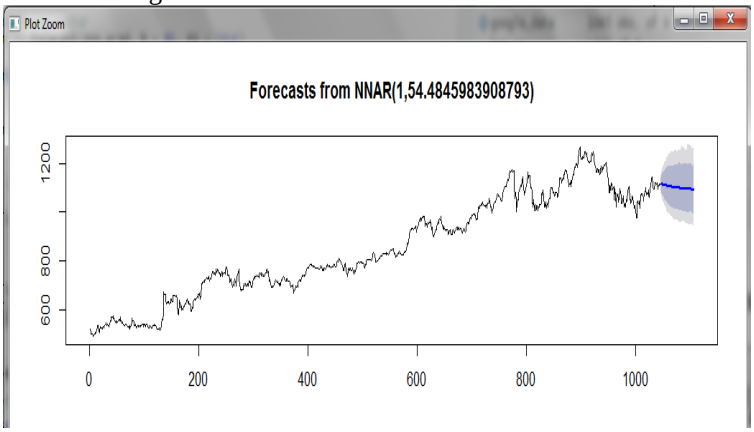


Figure 6 Forecasting using Feed Forward Neural Network

We then analyze the performance of the neural network model using the following parameters:

```
> plot(dnn_forecast, title = "NN")
Warning messages:
1: In plot.window(xlim, ylim, log, ...) :
  "title" is not a graphical parameter
2: In title(main = main, xlab = xlab, ylab = ylab, ...) :
  "title" is not a graphical parameter
3: In axis(1, ...) : "title" is not a graphical parameter
4: In axis(2, ...) : "title" is not a graphical parameter
5: In box(...) : "title" is not a graphical parameter
> accuracy(dnn_forecast)
                     ME     RMSE      MAE          MPE     MAPE      MASE        ACF1
Training set 0.1044731 13.02199 8.770758 -0.009518388 1.023297 0.9956171 0.02238915
There were 15 warnings (use warnings() to see them)
>
```

Figure 7 Performance Evaluation using Feed Forward Neural Network

Comparison of all models

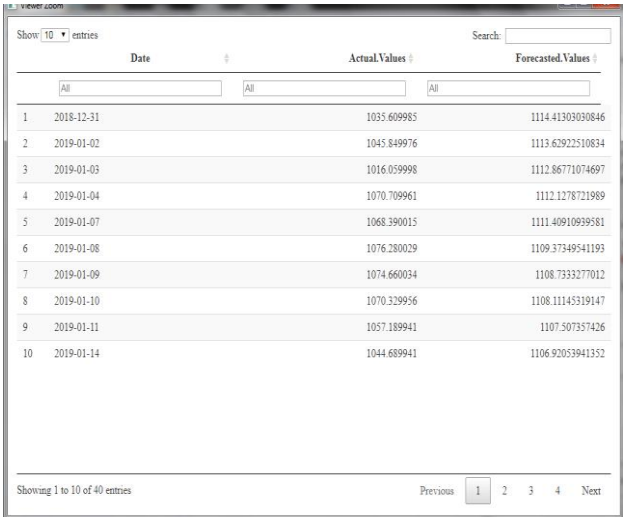
Analyze all the three models with parameters such as RMSE (Root Mean Square Error), MAE (Mean Absolute Error) and MAPE (Mean Absolute Percentage Error) and the comparison summary are shown in figure 8.

Summary of Models

Model	RMSE	MAE	MAPE
KNN	44.04	33.78	3.17
Neural Network	13.01	8.77	1.02

Table 1 Summary of Models

Thus, from the above summary of model performance parameters, we can see that Neural Network Model performs better than the KNN Model for the datasets. Hence, we will use the Neural Network Model to forecast the webdatas for the next two months which is shown in figure 8.


Show 10 entries Search:

	Date	Actual.Values	Forecasted.Values
1	2018-12-31	1035.609985	1114.41303030846
2	2019-01-02	1045.849976	1113.62922510834
3	2019-01-03	1016.059998	1112.86771074697
4	2019-01-04	1070.709961	1112.1278721989
5	2019-01-07	1068.390015	1111.40910939581
6	2019-01-08	1076.280029	1109.37349541193
7	2019-01-09	1074.660034	1108.7333277012
8	2019-01-10	1070.329956	1108.11145319147
9	2019-01-11	1057.189941	1107.507357426
10	2019-01-14	1044.689941	1106.92053941352

Showing 1 to 10 of 40 entries Previous 1 2 3 4 Next

Figure 8. Forecasting a value for next two months

VI. CONCLUSION

A forecasting algorithm is a process that seeks to predict future values based on the past and present data. This historical data points are extracted and prepared trying to predict future values for a selected variable of the dataset. During market history there have been a continuous interest trying to analyze its tendencies, behavior and random reactions. From the summary of model performance parameters, we can see that Neural Network Model performs better than the NN and the KNN Model for the datasets. Most importantly, the above experiment not only helped us in predicting the outcome but also gave us valuable insights about the nature of data, which can be used in future to train our classifiers in a much better way.

REFERENCES

[1] Ali, A. 2001.Macroeconomic variables as common pervasive risk factors and the empirical content of the Arbitrage Pricing Theory. Journal of Empirical finance, 5(3): 221–240.

[2] Srinath Ravikumar , Prasad Saraf " Prediction of WebData using Machine Learning (Regression,Classification) Algorithms " in 2020, IEEE .

[3] Kunal Pahwa , Neha Agarwal" Webdata Analysis using Supervised Machine Learning " in IEEE 2019.

[4] Indronil Bhattacharjee, Pryonti Bhattacharja "WebData Prediction: A Comparative Study between Traditional Statistical Approach and Machine Learning Approach", in IEEE 2019.

[5] Pawee Werawithayaset, Suratose Tritilanunt, "WebClosing Data Prediction Using Machine Learning", in IEEE, 2019.

[6] H. L. Siew and M. J. Nordin, "Regression techniques for the prediction of webdata trend," 2012 International Conference on Statistics in Science, Business and Engineering (ICSSBE), Langkawi, 2012, pp. 1-5.

[7] K. V. Sujatha and S. M. Sundaram, "Webindex prediction using regression and neural network models under non normal conditions," INTERACT-2010, Chennai, 2010, pp. 59-63.

[8] S. Liu, G. Liao and Y. Ding, "Webtransaction prediction modeling and analysis based on LSTM," 2018 13th IEEE Conference on Industrial Electronics and Applications (ICIEA), Wuhan, 2018, pp. 2787-2790.

[9] T. Gao, Y. Chai and Y. Liu, "Applying long short term momory neural networks for predicting webclosing data," 2017 8th IEEE International Conference on Software Engineering and Service Science (ICSESS), Beijing, 2017, pp. 575-578.

[10] K. A. Althelaya, E. M. El-Alfy and S. Mohammed, "Evaluation of bidirectional LSTM for short-and longterm webdata prediction," 2018 9th International Conference on Information and Communication Systems (ICICS), Irbid, 2018, pp. 151-156

[11] M. Usmani, S. H. Adil, K. Raza and S. S. A. Ali, "Webdata prediction using machine learning techniques," 2016 3rd International Conference on Computer and Information Sciences (ICCOINS), Kuala Lumpur, 2016, pp. 322-327.

[12] K. Raza, "Prediction of Webdata performance by using machine learning techniques," 2017 International Conference on Innovations in Electrical Engineering and Computational Technologies (ICIEECT), Karachi, 2017, pp. 1-1.

[13] H. Gunduz, Z. Cataltepe and Y. Yaslan, "Webdata direction prediction using deep neural networks," 2017 25th Signal Processing and Communications Applications Conference (SIU), Antalya, 2017, pp. 1-4.