# INDIAN STOCK MARKET PREDICTOR SYSTEM

[1]VIVEK JOHN GEORGE, [2]DARSHAN M. S, [3]SNEHA PRICILLA, [4]ARUN S, [5]CH. VANIPRIYA

Department of Computer Science and Engineering, Sir M Visvesvarya Institute of Technology, Yelahanka, Bangalore, Karnataka, India
Email: v1v3k.john@gmail.com

**Abstract** – Stock market prediction models are one the most challenging fields in computer science. Our proposed system combines both the statistical numeric data and sentiments of the stock on the internet to predict future prices in the stock market. The existing models are predicting stock market prices either by using statistical data or by analyzing the sentiments on the internet. The proposed system combines both these methods to develop a hybrid machine learning Stock Market Predictor based on Neural Networks, with intent of improving the accuracy.

**Keywords**: Machine Learning, Neural networks, Sentiment analysis

## I. INTRODUCTION

People generally want to invest their money in stock markets and expect high returns in short period of time. All of these investors have one common goal, which is to maximize their profits .They need to know the right time to buy or sell their investment. Only with a deep understanding of the working principles of the stock markets can one make the right decisions.

As of June 2012 India has the world's third-largest Internet user-base with over 137 million. The major Indian stock markets introduced Internet trading (online-trading) in February 2002.

With the exponential growth of online trading in India, large amount of information is available on the net about stock related data. There are two kinds of data available, numerical data in the form of historical statistics and textual data in the form of news feeds provided by online media. Most of the earlier work on stock prediction was based on the numeric data like historical stock prices. They used to analyze the technical predictors to expect the rise or fall of stock. Now the online media is also playing an important role in the stock market.

This research is aimed at improving the efficiency of stock market prediction models by combining historical pricing models with sentimental analysis by developing a hybrid neural network to which historical prices and sentimental values are fed as inputs.

Historical pricing models involve analysis of historical prices of a particular stock to identify patterns in the past and are extended to predict the future prices. Market sentiment is monitored with a variety of technical and statistical methods such as the number of advancing versus declining stocks and new highs versus new lows comparison. A major share of overall movement of an individual stock has been attributed to the market sentiment. Web crawlers can be created to extract specific news feed data for individual stock for sentimental analysis.

In the last decade, investors are also known to measure market sentiment through the use of news analytics, which include sentiment analysis on textual stories about companies and sectors. For instance, in October 2008, there was an online attack on the ICICI bank, which made the stock value of that to come down from 634.45 to 493.30 within 7 days of span. It clearly shows that the rumors which were spreading through the online media make an impact on the stock price. This clearly shows that the news about a particular financial firm on various on line media also plays an important role in stock price prediction.

## II. LITERATURE REVIEW

Several authors have attempted to analyze the stock market. They used quantitative and qualitative information on the net for predicting the movement of the stock.

In [1], the authors proposed a system for quantifying text sentiment based on Neural Networks predictor. By using the methodology from empirical finance, they proved statistically significant relation between text sentiment of published news and future daily returns.

In [2], the author work used only volume of posted internet stock news to train neural network and predict changes in stock prices.

In [3], Liang and Chen employed natural language processing techniques and hand crafted dictionary to predict stock returns. They used feed forward neural network with five neurons in the input layer, 27 in the hidden layer, and one output neuron. Since only 500 news items was used for the analysis, no statistical significance of the results could be found.

In [4], the authors proposed a system learns the correlation between the sentiments and the stock

values. The learned model can then be used to make future predictions about stock values. They showed that their method is able to predict the sentiment with high precision and also showed that the stock performance and its recent web sentiments are also closely correlated.

[5] Developed a system called E-Analyst which collects two types of data, the financial time series and time stamp news stories .It generates trend from time series and align them with relevant news stories and build language models for trend type. In their work they treated the news articles as bag of words.

## III. PROPOSED SYSTEM ARCHITECTURE

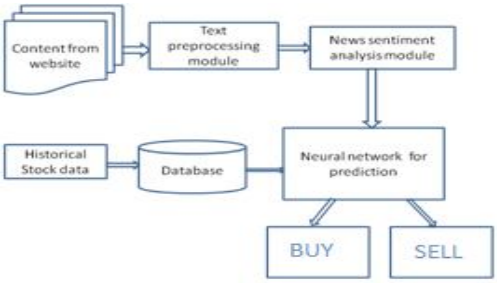


**Figure 1. System Model**

Steps involved in financial sentiment extraction:

*A. Extraction of Data*

There are two kinds of data to be extracted historical data about a firm and news articles regarding that company. Gathering data from internet is solely based on the (SOR) Subject of Reference (e.g. ICICI bank).Some web mining techniques (ex. crawler) are used to gather all web pages. Extracting historical stock values is easy, but extracting news articles is tricky, since different websites have different structures.

Various sources of financial related data are:

1. 20 to 30 mainstream digital newspapers or online news channels.
2. Authorized sources: financial newsletter
3. Expert commentary: money control, trader G
4. Social media: Facebook, Twitter
5. Alerts &feeds: Bloomberg, Google alerts.

*B. Text Cleaning*

It is mostly heuristic based and case specific. By this what we mean is to identify the unwanted portions in the extracted contents with respect to different kinds of web documents (e.g. News article, Blogs, Review, Micro Blogs etc) and then write simple cleanup codes based on that learning what , which will remove such unwanted portions with high accuracy.

*C. Extract the Sentiment*

We have a list of positive negative lexicon words. Sentiment Analysis of web documents can be defined as the consumer opinion expressed through online medium e.g. Blog or review. Now days a consumer can choose to post his/her sentiment about a particular brand/product/feature online which can be categorized broadly as Positive, Negative or Neutral. The web documents where such sentiment has been expressed can be referenced for various analytical/actionable causes by that brand representative. For example, considerable amount of negative sentiment expressed by consumers about customer service of ICICI bank can be actionable insight for ICICI bank, in which case ICICI might want to restructure its customer service to give better customer satisfaction and thus tend to reduce negative sentiment about it in the net. Sentiment analysis can be done on the clean extracted web documents in two manners - manual rating or automated rating of such web documents. While manual rating is a near perfect method to do it, but it is a slow process when the volume of web documents is too high. Whereas automated system will be much faster method, but is bound to lack accuracy since it is effectively machine learning and deriving human sentiments through user generated content. Also, language barrier is a major challenge for automated sentiment analysis. Nevertheless, extensive research work on Natural Language Processing has addressed such challenges well and reasonably high performance machine learning techniques have evolved which can do sentiment analysis of web documents.

*D. Neural Networks*

Use one Neural networks are members of a family of computational architectures inspired by biological brains. Such architectures are commonly called "connectionist systems", and are composed of interconnected and interacting components called nodes or neurons (these terms are generally considered synonyms in connectionist terminology, and are used interchangeably here). Neural networks are characterized by a lack of explicit representation of knowledge; there are no symbols or values that directly correspond to classes of interest. Rather, knowledge is implicitly represented in the patterns of interactions between network components (Lugar and Stubblefield, 1993). A graphical depiction of a typical feedforward neural network is given in Figure 2. The term “feedforward” indicates that the network has links that extend in only one direction. Except during training, there are no backward links in a feedforward network; all links proceed from input nodes toward output nodes.

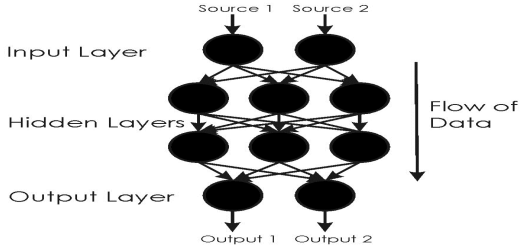


**Figure 2: A typical feedforward neural network.**

Individual nodes in a neural network emulate biological neurons by taking input data and performing simple operations on the data, selectively passing the results on to other neurons. The output of each node is called its "activation" (the terms "node values" and "activations" are used interchangeably here). Weight values are associated with each vector and node in the network, and these values constrain how input data (e.g., satellite image values) are related to output data (e.g., land-cover classes). Weight values associated with individual nodes are also known as biases. Weight values are determined by the iterative flow of training data through the network (i.e., weight values are established during a training phase in which the network learns how to identify particular classes by their typical input data characteristics). Once trained, the neural network can be applied toward the classification of new data. Classifications are performed by trained networks through 1) the activation of network input nodes by relevant data sources [these data sources must directly match those used in the training of the network], 2) the forward flow of this data through the network, and 3) the ultimate activation of the output nodes.

*Multi-Layer Networks and Backpropagation*

A multi-layer, feed forward, backpropagation neural network is composed of 1) an input layer of nodes, 2) one or more intermediate (hidden) layers of nodes, and 3) an output layer of nodes (Figure 1). The output layer can consist of one or more nodes, depending on the problem at hand. In most classification applications, there will either be a single output node (the value of which will identify a predicted class), or the same number of nodes in the output layer as there are classes (under this latter scheme, the predicted class for a given set of input data will correspond to that class associated with the output node with the highest activation).It is important to recognize that the term "multi-layer" is often used to refer to multiple layers of weights. This contrasts with the usual meaning of "layer", which refers to a row of nodes .For clarity, it is often best to describe a particular network by its number of layers, and the number of nodes in each layer (e.g., a "4-3-5" network has an input layer with 4 nodes, a hidden layer with 3 nodes, and an output layer with 5 nodes).

The use of a smooth, non-linear activation function is essential for use in a multi-layer network employing gradient-descent learning. An activation function commonly used in backpropagation networks is the sigma (or sigmoid) function:

$$a_{j_m} = \frac{1}{(1+e^{-S_{j_m}})} \qquad where\ S_{j_m} = \sum_{x=0}^{n} w_{ij_x} a_{i_x} \qquad \ldots\ldots[1]$$

where $a_j$ sub m is the activation of a particular "receiving" node m in layer j, Sj is the sum of the products of the activations of all relevant "emitting" nodes (i.e., the nodes in the preceding layer i) by their respective weights, and $w_{ij}$ is the set of all weights between layers i and j that are associated with vectors that feed into node m of layer j. This function maps all sums into [0, 1] (an alternate version of the function maps activations into [-1, 1]). If the sum of the products is 0, the sigma function returns 0.5. As the sum gets larger the sigma function returns values closer to 1, while the function returns values closer to 0 as the sum gets increasingly negative.

## IV. DATA COLLECTION

We merge two sources of data: a corpus of news articles, a dataset of historical data

*A. News Data*

The first source is extraction of news articles from moneycontrol, it contains important news for individual stocks. *Moneycontrol* is India's leading financial information source. Manage your finance with our online Investment Portfolio, Live Stock Price, Stock Trading news etc. The corpus is taken for INFOSYS for one year from Jan 2012 to Jan 2013. We wrote a web scraper in R language to get the news articles and calculate the sentiments.

*B. Historical Data*

We have extracted historical values from http://ichart.finance.yahoo.com for the year of 2012 for the stock of Infosys in NIFTY.

## V. EXPERIMENTS

The proposed system uses a predictor based on Neural Network. We train the Neural Network first. After training, the system is fed with historical stock prices and postings or text of the news articles about a particular firm as inputs. The neural network with 1 input layer, 2 hidden layers and 1 output layer have been used. Here are and it produces a numerical text sentiment measure as an output. We will show that produced text sentiment corresponds with future returns of the company's stock.

For extraction of news articles from money control, we coded a scraper in R to get the news articles for the year 2012 of INFY. We cleaned the news articles and we scaled the articles from -1 score to +1 score and -1 being the most negative article.

The score is calculated according to the following formula:

$$SCORE = \frac{\sum(positive\ matches) - \sum(negative\ matches)}{\sum(positive\ matches) + \sum(negative\ matches)} \qquad \ldots\ldots\ldots\ldots\ldots\ldots [2]$$

Stock values of stock for 4 consecutive days were taken as inputs .The fifth input will be the previous day's news article score. These values are fed into neural networks .We have used a java framework, Neuroph

[8] is used to train and predict the values. We have used the neural network with 1 input layer and 2 hidden layers and 1 output layer and the model is tested with different number of nodes in the hidden layers .The model predicts the rise or fall in the stock price based on which the investor will take a decision to buy or sell on a day to day basis that is for intraday trading. The model is predicting for the day i using the previous four days values for the company and the fifth input is the Sentiment value of i-1 day. If there is no news article for the previous day, the sentiment value is 0. The Neural network model used to predict the values is shown below.

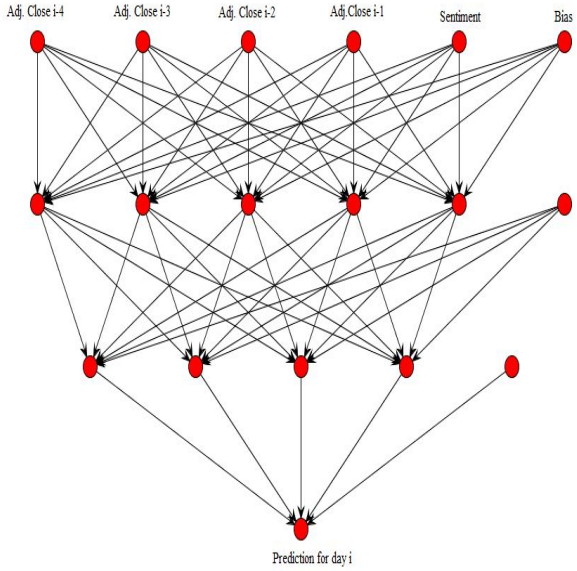


**Figure 3. Neural Network Model**

## VI. RESULTS

Based on the analysis of various neural network structures developed by changing the number of hidden layers and input layer nodes, the following results as shown in table I were obtained.

*Table I shows the accuracy of the predicted values*

| No of nodes in the 1st Hidden layer | No of nodes in the 2nd Hidden layer | Training set accuracy | Training set accuracy |
|---|---|---|---|
| 20 | 10 | 80.12% | 79% |
| 5 | 4 | 80.12% | 71% |
| 10 | 10 | 78% | 74.756% |
| 30 | 25 | 52% | 58% |

The correlation between the actual stock values and predicted stock values are shown in the graph below in figure.

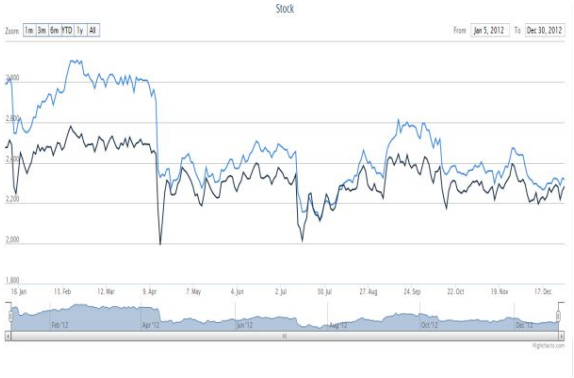


**Figure 4. Plotting of Stock values**

## CONCLUSION AND FUTURE WORK

The main contribution of this paper is to suggest a new method for automatically predicting the stock price. We have shown that stock prices predicted from historical prices and sentiments are significantly correlated with actual stock prices of a particular company. Future work would be extending these results by using Deep Multilayer Neural Networks with more than two hidden layers for determining text sentiment. One can go even further and use information of more companies. And also use complex algorithms like SVM, Naïve Bayes theorem to classify the sentiments.

## ACKNOWLEDGEMENTS

The authors wish to express their gratitude to Prof. Dilip K Sen, H.O.D., who was abundantly helpful and offered invaluable assistance, support and guidance. They also thank the management of SIR M.V.I.T., for their constant support.

## REFERENCES

[1] Caslav Bozic and Detlef Seese, *"Neural Networks for Sentiment Detection in Financial", JEL Codes: C45, D83, and G17,* Institute of Applied Informatics and Formal Description Methods, Karlsruhe Institute of Technology (KIT)

[2] Liang, X. 2005, *"Impacts of Internet Stock News on Stock Markets Based on Neural Networks"*, : ISNN 2005, LNCS 3497, pp. 897–903, 2005. © Springer-Verlag Berlin Heidelberg 2005

[3] Liang, X. and Chen, R.-C. 2005, *"Mining Stock News in Cyberworid Based on Natural Language Processing and Neural Networks"*, 13-15 Oct. 2005, Neural Networks and Brain, 2005. ICNN&B '05. International Conference

[4] Vivek Sehgal and Charles Song, *"SOPS: Stock Prediction using Web Sentiment"*, 2007 IEEE DOI 10.1109.

[5] Khurshid Ahmad and Yousif Almas, *"Visualising Sentiments in Financial Texts"*,

[6] Yang Gao, Li Zhou, Yong Zhang, Chunxiao Xing, *"Sentiment Classification for Stock News"*, 978-1-4244-9142-1/10/$26.00©2010 IEEE

[7] Susanne Glissman, Ignacio Terrizzano, Ana Lelescu, Jorge Sanz,*"Systematic Web Data Mining with Business Architecture to Enhance Business Assessment Services"*, 2011 Annual SRII Global Conference, 9 78-0-7695-4371-0/11 $26.00 © 2011 IEEE DOI 10.1109/SRII.2011.99

[8] Hailiang Chen, Prabuddha De, Yu (Jeffrey) Hu, and Byoung-Hyoun Hwang,"*SENTIMENT REVEALED IN SOCIAL MEDIA AND ITS EFFECT ON THE STOCK MARKET"* , IN 479072011,IEEE STATISTICAL SIGNAL PROCESSING WORKSHOP

[9] Yong LI; Jian WANG,*"Factors on IPO under-pricing Based on Behavioral Finance Theory: Evidence from China"*, 978-1-4577-0536-6/11/$26.00 ©2011 IEEE,

[10] Kaihui Zhang1, Lei Li2, Peng Li3 and Wenda, *"Stock Trend Forecasting Method Based on Sentiment Analysis and System Similarity Model"*,

[11] Binoy.B.Nair, V.P Mohandas, N. R. Sakthivel," *A Decision Tree- Rough Set Hybrid System for Stock Market Trend Prediction"*, International Journal of Computer Applications (0975 – 8887), Volume 6– No.9, September 2010

★★★