Performance Evaluation of Machine Learning Classifiers for Stock Market Prediction in Big Data Environment

Sneh Kalra¹, Sachin Gupta², Jay Shankar Prasad³

¹ Research Scholar, Department of CSE, MVN University, Palwal, Haryana - 121102
² Associate Professor, Department of CSE, MVN University, Palwal, Haryana - 121102
³ Professor, Department of CSE, Krishna Engineering College, Uttar Pradesh - 201007

snehchhabra@gmail.com, sachin.gupta@mvn.edu.in, hello2jsp@gmail.com

http://doi.org/10.26782/jmcms.2019.10.00022

Abstract

Implementing machine learning models for the stock’s big data emerged as a component of algorithmic trading systems. This paper proposed a hybrid stock prediction model based on the collection of qualitative and quantitative data of particular stocks. In addition to tweets and news data, product reviews of the specific companies traded under National Stock Exchange are considered to analyze their effect on the stock movements. Historical Prices will be integrated with sentiment values generated from tweets, news and product reviews data to construct the amalgam model using Apache Spark and HDFS for storage of large data. The proposed model has been implemented in Google Cloud Platform with different cluster configurations. The paper compares the prediction accuracy based on various types of input data provided to the model using some popular machine learning algorithms.

Keywords: Supervised learning, Product Reviews, Google Cloud, Big data, Apache Spark.

I. Introduction

Due to divergent market conditions, it is quite complex to present a model that operate consistently under all possible conditions. In recent decades, credit goes to the development in science and communication technology, computational intelligence models started rising as a practical substitute to the conventional decision support systems. Former systems mostly rely on static rules and analyses methods, hence readily be obsolete.
Due to abundant use of social media, IOT devices, mobile apps a vast amount of data is generating every moment and this raw data may contain useful events, important information hidden inside it. The abundant exploitation of social media in the past decade has remodelled the approach of how investors forecast the stock exchange [VIII]. Social media provides massive volumes of real-time user view. To extract the patterns or events from this data, it needs to get analyzed. Investors can potentially exploit that information by applying machine learning and natural language processing techniques. For storage and analysis of huge data, researchers have exploited the Hadoop framework with its Map Reduce Applications. Map Reduce, works in a batch processing manner in which processes get executed one by one and takes a lot of time. Afterward, analysts move towards Apache Spark, Flume, and Kafka for fast and real-time processing of data [XV].

Real-time processing can be made through providing the data into a systematic analytical application, the instant it is obtained and immediately recorded observations return to the user to let him take an action. This type of processing is moderately adopted in Big Data applications such as recommendation system used by retailers, or for examining the performance of machinery in the manufacturing industry [VI].

The trouble is to augment the precision of forecasting stock market trends from huge data produced from resources like news channels, web applications and social media, load it on the storage system and to do parallel processing on it [IV]. In addition to tweets and news of a specific company, we have analyzed product reviews of companies to observe the impact of customer reviews for various products of a specific company for its stock price fluctuations. The model makes use of sentiment analyzer to find the polarity of the textual data produced from social media, online product reviews, and news websites and amalgamate the outputs with historical stock price values as an extra attribute. The main classification output is determined as binary categorization rise or fall and some popular supervised machine learning techniques are employed to augment the prediction accuracy.

The proposed hybrid model built on Google Cloud Platform using Hadoop Distributed File System (HDFS) as data storage and Apache Spark framework for parallelizing data processing, creating best utilization of resources to get precise prediction results.

Prediction accuracy has been computed with five different parameters including historical price, historical price and news sentiments (HN), historical price with tweet sentiments (HT), historical price with tweets and news sentiments (HNT), and finally historical data with news, tweets and product reviews sentiments (HNTR).
II. Related Work

The majority of the investors, stock analytics forecast stock close values using historical stock records, but it must utilize the massive amount of data resources to augment the accuracy of prediction and discover innovative ways to make a sound decision at right time. Though, it may be achievable by handling available data in different formats at the same point, and amazing data processing framework.

Study [I] analyzed the real-time tweets to find its relation with the share market movements. To classify the real-time fetched tweets, the model was trained with news data. Their proposed system worked in five phases named News Collection, Streaming data, Stock Data Retrieval, Streaming Pipeline, and Visualization. They achieved 77% accuracy for classification of tweets and reported 80% accuracy for predicting up/down trends in the big data environment.

Study [VIII] proposed a fast prediction model that works in 5 phases. The model forecasted the stock values by implementing the ARMA model for 20 top companies by reading bulk values for all stocks and streamed at a window of five thousand trades per second using Socket Text Stream. Their observations reported that predictions made with AR were best and ARMA predictions followed an exaggeration trend.

Study [V] implemented a forecast model using artificial neural network by considering technical indicators with historic data. To develop the model, first of all, financial data was converted in to buy/hold/sell signals layer. The model was trained with 10 years of stock data in Apache Spark big data framework. Their findings observed that in most cases NN model achieved comparable results against buy/hold strategy by selecting the most suitable technical indicators.

Study [III] implemented the model that worked in Offline mode and Real-Time mode. Data has been collected from Stockwits, News websites. To implement the model, sentiments of tweets and news were combined with extracted features of Historical data. To build the model, some supervised machine learning techniques were employed. Apache Spark used for handling real-time data streams and Hadoop HDFS for storing data in a distributed way. Offline mode worked by considering historical as well as today’s stock data after the market closing session and real-time mode worked with livestock data during the market session.

Study [IV] proposed a forecasting technique using sentiments scores of news data and historical stock dataset. Sentiment scores were calculated using Naive Bayes Classifier and they utilized the Hive ecosystem for pre-processing of news data. The proposed model implemented with Neural Network by providing sentiments and historic dataset as input to it. They trained the model with 3 years of news data and one
year of news data. Their findings suggest that the model will attain high accuracy if it is trained with data of recent years.

III. Proposed Work

The proposed work is designed using 4 main components named Data Acquirement, Data storage, Data analysis, and Trend Prediction as shown in Fig. 3.

For many years, Hadoop remained a best open source for Big Data analysis but recently advance Apache Spark framework has taken place over Hadoop. Spark replaces MapReduce for administrating the execution of tasks across a cluster of computers. However neither they perform the same tasks nor are they mutually exclusive but able to work together. Map-Reduce is an efficient parallel programming tool enabling fault tolerance, scalable, flexible, and cost-effective processing [II]. The main concern is the amount of computational time. Speed is a key component for large datasets, which are efficient for query processing and computation. The limitations faced by the Map-Reduce programming model, which are high disk rate, low throughput and diminishing performance of a cluster are addressed by Spark [II]. Spark framework was introduced for parallel distributed Systems. It is an independent technology and is not part of the Hadoop ecosystem. Spark does not have its own distributed storage system; this is the cause for installing Spark on the top of Hadoop. Spark has a powerful processing unit for fast computations and advanced methods like interactive query analysis and streaming over the MapReduce model. Spark supports applications with keeping scalability and fault tolerance as extended features. The significant feature of Spark is its in-memory cluster computing for increasing the processing speed of an application by using its RDDs. Fig. 1 and fig. 2 represents the Hadoop Map-Reduce architecture and Spark architecture respectively.

![Hadoop Map Reduce Architecture](https://beyondcorner.com)

Fig. 1 Hadoop Map Reduce Architecture (Source: https://beyondcorner.com)
The proposed model makes use of sentiment analyzer to find the polarity of textual data collected from tweets, news data, and product reviews about some products of a company and some popular supervised machine learning techniques for the prediction task. Some reputed companies in NSE like Maruti Suzuki, Dabur producing a vast amount of unstructured or structured data that require immense storage space to store enormous data and an effective processing system to analyze it in a limited time.

Historical price values are utilized as the key data resource for trend prediction algorithms. Yahoo finance is used to obtain stock data for Dabur and Maruti Suzuki companies.

Fig. 3 Proposed Hybrid Stock Prediction Model
III.1 **Data Acquisition**: Data has been collected from the various sources as described below:

**Tweets Collection**: - Twitter data is freely available to everyone but there is a limit set by twitter to download only last 7 days data. For this study, Tweets are downloaded every day with the help of a Rapidminer process [VII]. As the downloadable data was present in raw form and contains irrelevant and noisy data which is not directly suitable for the sentiment analysis process. To use them for sentiment purpose, fetched tweets were filtered using an own created dictionary of stock related keywords [VII] shown below.

| Keyword | Stock Related Keywords |
|---------|------------------------|
| Hit     | Down Depreciation Swell Trouble Share |
| Sell    | Earning Announce Soar Sector Accumulate |
| Bull    | Nifty Bear Wipe Down Hold |
| Profits | Q1 Estimate Rise Up Target |
| Plunge  | Q2 Market Quarter Dollar Miss |
| Results | Q3 Roadblock Hold Miss Sensex |
| Buy     | Q4 % Dive Loss Beat |
| QOQ     | Bear Rupee Target |

![Fig. 4](dictionary_of_keywords_for_stock_market.png)

**Market News Collection**: - Market News data has been collected from a number of online resources for last six months of the year 2018 using XML API in MS-Excel. Process to fetch the News Data on daily basis in MS-Excel explained below

1. Click on the source under Developer Tab.
2. Click on the XML Maps and a dialog box named XML Maps will open.
3. Click on Add button and enter the RSS feed of required online resource and change the file type as XML.
4. The RSS feed is added to the XML Maps and click Ok.
5. Import all the required fields in to the excel sheet one by one.

As shown in Fig. 5 the fetched news for a day contains all types of news, so for this study only news of required companies were filtered.

**Product Reviews**: - There are two types of companies named service-based and product-based. So in this paper, product-based companies are considered for the experimentation purpose. The product reviews of each of the company are collected from many sources [IX-XIV], [XVI],[XVIII] using a crawler written in Scala.
Historical Data Collection: Historic data of Dabur and Maruti Suzuki Company has been collected from Yahoo Finance [XIX] for a last 6 months of the year 2018.

Fig. 5 Sample of Fetched News Data

III.ii Data Storage (Hadoop Distributed File System)

All the fetched tweets, product reviews and collected news were store in a distributed manner in Hadoop System.

III.iii Data Analysis

Data analysis has been done in the Google Cloud by creating a cluster and uploading the required data. Analysis steps involved the following steps

Data Pre-Processing: Before passing the data to a sentiment analyzer, it has been pre-processed with lowercase conversion, and junk word removal and stop word removal steps carried out in Scala [VII]. In [VII] the author have pre-processed the data using Apache Spark framework and utilized that data for predicting next day close price of particular stocks.

Sentiment Analysis: Sentiment analysis has been carried out for tweets, news and product reviews data [VII]. A lexicon base approach has been employed for categorizing the data in to 1 and 0 classes. Fig. 6 shows sentiment analysis step for the pre-processed data in Apache Spark using Scala language.

III.iv Trend Prediction using Binary Classifier

Trend has been predicted by implementing some popular supervised machine learning algorithms named SVM, Naive Bayes, Random Forest and Logistic Regression in Apache Spark framework.
IV. Experimentation and Results

Data has been stored and analyzed in Google Cloud by creating a GCP project and some selected supervised machine learning algorithm has also been implemented in the Google cloud.

For the experimentation purpose, in addition to tweets and news of Dabur Company, product reviews have been collected for a few products like honey, chaywanprash, rose gulabari, almond oil, and tooth-paste and for Maruti, we have collected reviews for Maruti Suzuki - Alto, Baleno, Ciaz, Swift and Swift Dzire.

Initially, the pre-processing of all the available data has been done using Map Reduce and time taken by the Map-Reduce process is almost 8 times of the time taken by a Spark process with different cluster configurations. Machine learning algorithms have been implemented using Apache Spark-Scala process. Fig. 7 shows the time taken by the map-reduce process for pre-processing of tweets, news and product reviews for Dabur and Maruti Suzuki Company. For the experiment purpose, we have created a total of 8 clusters in the Cloud-Based Hadoop environment. Fig. 8 shows various clusters with varying number of nodes, executors, and cores.

Worker Nodes: - Worker node works as a slave node and execute the application code within the cluster. Master node is accountable for allocating tasks to a worker node and worker node actually does the assigned work. Master node schedules the various tasks based on resources and availability of resources are reported by worker node. Worker node processes the data stored on the node [XVII].

Executors:- It is a single Java Virtual Machine process and is accountable for executing the tasks and sustains all data in disk storage or memory. An executor alive

---

```scala
1. Load dictionary of positive and negative keywords.
2. Fetch the pre processed data from HDFS.
3. Count positive and negative classes for tweets, news and reviews.
   val neg_count= negWordFiltered.count
   val pos_count= posWordFiltered.count
   val df_tweets = Seq((pos_count,neg_count,z)).toDF("positive_tweets","negative_tweets","date_created")
   df_tweets.rdd.map(_.toString().replace("[","\n").replace("]","\n")).saveAsTextFile("/rs/hive/hdfc_tweets/date_created=\n+z+h")
   h=h+1
```

Fig. 6 Sentiment Analysis Process for the Pre-Processed Data in Apache Spark.
during the activation of a Spark Application and executes the tasks in multiple threads. It is initiated for a worker node. A lone (single) node able to execute multiple executors and an executor may also contain multiple worker nodes [XV].

Core: -A core is the basic computation unit of Central Processing Unit and a CPU can possess number of cores to perform the tasks at a given point of time. In spark, an executor can run how many parallel tasks are controlled by CPU [XV].

As it can be observed in fig 7 and fig 8, pre-processing time of tweets using map-reduce process is 6 times more than pre-processing time of tweets with a spark process. Fig 8 shows a varying number of nodes, executors, and nodes used to observe the processing time for Spark process and the pre-processing time is reducing at each step from cluster 1 to cluster 6, but after a saturation point due to small size of
datasets, processing time tend to again increase in both the cases. Fig 9 shows the graphical representation of pre-processing time for Dabur and Maruti Suzuki Company.

![Graph showing pre-processing time for Dabur and Maruti Suzuki Tweets in Spark](image)

**Fig. 9** Pre-processing time for Dabur and Maruti Suzuki Tweets in Spark

![Bar charts showing prediction accuracy for Dabur Company](image)

**Fig. 10** Prediction accuracy for Dabur Company

![Bar charts showing prediction accuracy for Maruti Suzuki Company](image)

**Fig. 11** Prediction accuracy for Maruti Suzuki Company

*Copyright reserved © J. Mech. Cont. & Math. Sci.*

*Sneh Kalra et al*
As it can be observed from fig 10 and 11, Naive Bayes classifier has less prediction accuracy for all types of input data. Initially, the executed algorithms have less accuracy with only historical data but when historical data is combined with sentiment values of news or tweets it has increased up to an extent. When historical prices are combined with sentiment values of news and tweets, the accuracy has been enhanced and finally, with all available data including historical prices, news, tweets, and product reviews sentiments has shown a big improvement in prediction accuracy.

V. Conclusion

The input data given to a system plays a vital role in its performance. In this paper, system performance has been improved by adding a new input dataset at every step. Finally, the model resulted in enhanced prediction accuracy with product reviews data in combination with news and tweets data. Among all the classifiers, random forest performed best and produced an accuracy of 87.02 and comparatively naive bayes generated a less accuracy of 60.23. Pre-processing time of available data with Map-Reduce and Spark has been compared and results show that spark executed the same task eight times faster than a Map-Reduce process. Future work will consider more data for processing and analysis. In addition real time streaming of data will be taken in to account to make the decision instantly and will be applied on some service based companies.

References

I. C. Lee and I. Paik, Stock Market Analysis from Twitter and News Based on Streaming Big Data Infrastructure, in Proceedings - 2017 IEEE 8th International Conference on Awareness Science and Technology, iCAST, Taichung, 2017, pp. 312-317.

II. J.V.M. Lakshmi, A Framework Model on Big Data Analytics using Machine Learning Techniques for Prediction on Datasets”, Ph.D. dissertation, Dept. Comp. Sci. and App., Sri Chandrasekhar Univ., Enathur, Kanchipuram, 2018.

III. M. M. Seif et al, Stock Market Real Time Recommender Model Using Apache Spark Framework, Springer AMLTA 2018, pp. 671–683, 2018,https://doi.org/10.1007/978-3-319-74690-6_66.
IV. M. Shastri, S. Roy, M. Mittal, Stock Price Prediction using Artificial Neural Model: An Application of Big Data, EAI Endorsed Transactions on Scalable Information Systems, 2019, vol- 6, issue 20.

V. O. B. Sezer, A. M. Ozbayoglu, An Artificial Neural Network-based Stock Trading System Using Technical Analysis and Big Data Framework, ACMSE 2017, Kennesaw State University, GA, U.S.A., April, 2017, DOI -10.1145/3077286.3077294.

VI. R. T. Llame et al, Big Data Time Series Forecasting Based on Nearest Neighbours Distributed Computing with Spark, Knowledge Based Systems (2018), DOI: 10.1016/j.knosys.2018.07.026

VII. S. Kalra, S. Gupta, J. S. Prasad, Sentiments Based Forecasting for Stock Exchange using Linear Regression, unpublished.

VIII. O. B. Sezer, A. M. Ozbayoglu, An Artificial Neural Network-based Stock Trading System Using Technical Analysis and Big Data Framework, ACMSE 2017, Kennesaw State University, GA, U.S.A., April, 2017, DOI -10.1145/3077286.3077294.

IX. V. K. Menon et al, Bulk Price Forecasting Using Spark over NSE Data Set, International Conference on Data Mining and Big Data, DMBD 2016, pp 137-146.

X. https://www.amazon.in/ONGC-Natural-Corporation-Ltd-2019/product-reviews/9388426983

XI. https://www.autocarindia.com/car-reviews/2018-maruti-suzuki-alto-review-test-drive-412662

XII. https://www.autocarindia.com/car-reviews/2018-maruti-suzuki-ciaz-15-diesel-review-test-drive-412307

XIII. https://www.auto.ndtv.com/maruti-suzuki-cars/baleno/reviews

XIV. https://www.carwale.com/maruti-suzuki-cars/baleno/userreviews

XV. https://data-flair.training/forums/topic/what-is-worker-node-in-apache-spark-cluster/

XVI. https://www.mouthshut.com/product-reviews/Dabur-Vatika-Hair-Oil-reviews-925004768

XVII. http://site.clairvoyantsoft.com/understanding-resource-allocation-configurations-spark-application/

XVIII. https://www.snapdeal.com/product/dabur-chyawanprash-50g/657387760199/reviews?page=3&sortBy=RECENCY

XIX. https://www.yahoofinance.com