INTERNATIONAL JOURNAL OF ADVANCE RESEARCH, IDEAS AND INNOVATIONS IN TECHNOLOGY

ISSN: 2454-132X
Impact factor: 4.295
(Volume 5, Issue 3)
Available online at: www.ijariit.com

Stock market prediction using RNN and sentiment analysis

Ambati Venkata Malla Reddy
ambativenkatamallareddy@gmail.com
SRM Institute of Science and Technology, Chennai, Tamil Nadu

Dinesh Y.
dineshyoga4@gmail.com
SRM Institute of Science and Technology, Chennai, Tamil Nadu

Vamsi Krishna
krishna975420@gmail.com
SRM Institute of Science and Technology, Chennai, Tamil Nadu

Soundarya Miranam
miranam1997@gmail.com
SRM Institute of Science and Technology, Chennai, Tamil Nadu

ABSTRACT

Predicting the stock market Price is a challenging task. With an increase in data collection through the internet, data scientists try to extract valid data points for the prediction. However, these data points obtained in studies are usually only one way based upon data source and thus may not cover all the factors affecting the stock market. Therefore, to improve the prediction for stock market index movements, we use real-time tweets from the twitter using their API and sentiment analysis to predict the stock market moments by correlating them with the Existing data set used to train a recursive neural network model Evaluations on the data from the year 2004 to present upon Bombay stock exchange to demonstrate our model. Hence making the predictions almost accurate.

Keywords— *RNN, Twitter, Sentiment analysis, LSTM, BSE, Stock price*

I. INTRODUCTION

Anticipating money markets developments is a critical and testing assignment. As the Web data develops, scientists start to remove viable markers (e.g., the occasions and feelings) from the Web to encourage the expectation. Be that as it may, the markers acquired in past investigations are typically based on just a single information source and accordingly may not completely cover the elements that can influence the share trading system developments. In this work, to enhance the forecast for the stock exchange, we use the textures among various information sources and build up multi-source numerous cases demonstrate that can successfully consolidate occasions, assumptions just as the quantitative information into a thorough structure. To viably catch the news occasions, we effectively apply a novel occasion extraction and portrayal technique.

Assessments on the information from the year 2004 show the adequacy of our model. In expansion, our methodology can consequently decide the significance of every datum source and recognize the critical info data that is considered to drive the developments, making the forecasts interpretable.

Hence using existing data across the web can be used to train an effective model that can predict the stock based upon the past data. Although the prediction may be efficient, the recent studies have shown that the investment in stocks is based upon the sentiment of the investors, hence trying to do a sentimental analysis using the tweets about the company to invest and public tweets of investors across the globe can give an effective chance for our model to get better at predicting the stock prices.

2. LITERATURE SURVEY

Many models that have been used to predict the stock market prices or its movements were based upon a few methods only. The most common methods used are Statistical approach and the Artificial intelligence approach. Debadrita Banerjee's, Statistical data approach is done by trying to find a conclusion based upon analysing the data of a company or an organization statistically. Most of the time the analysis involves visualization of stock scores in the form of various plots and prediction of the scores using a time series model. Data might be decades or months old. This analysis may help them to use the required Algorithm that may be useful to predict the Stock movement. While the second approach is to not rely completely on the existing data and instead use real-time updating data related to the organization by acquiring them directly from the internet. Real-time data updating methods to fetch the data are used to predict the stock movement changes using various deep network algorithms.

Researchers also proposed a strategy utilizing the sentiment theme for securities trade speculation. They proposed two strategies to extract these subject-based assessment affiliations. One is JST-construct strategy that depends with respect to the current models, the other one is Aspect-based sentiment system where the themes and opinions are distinguished by their proposed technique. The work is well connected over different

stocks at the same time but there is a problem when there are heavy fluctuations in the stock prizes resulting in a downgrade of accuracy.

In the implemented work, five models have been developed and their performances are compared in predicting the stock market trends. These models are based on five supervised learning techniques i.e., Support Vector Machine (SVM), Random Forest, K-Nearest Neighbour (KNN), Naive Bayes, and SoftMax. The proposed architecture for the implemented work mainly consists of four steps: feature extraction from the given dataset supervised classification of the training dataset, supervised classification of the test dataset, and result evaluation. The dataset being larger in size can only be predicted while if the data is live stream then it is not at all possible for the algorithms to give a properly accurate answer.

The Stock Market is too volatile and current algorithms are not able to predict it when there is a change in multiple stocks which are dependent on each other.

The slight fluctuations can make a big difference in the stock market so there needs to be a clearer vision to correct it.

Overcoming the larger dataset with smaller with not degrading with accuracy is a must too.

3. RELATED WORK

Let's get an overview of the event extraction methods. Firstly, we convert the tweets extracted from the twitter into paragraph vectors and then model using past events or data with LSTM based upon the opening and closing prices of stocks of Bombay Stock Exchange. A temporary sentimental index function is used to extract most of the events. Then the Related Tweets are analysed using topic modelling to understand its context. Once these events are extracted, they are trained to correspond with the old available Bombay stock data using Deep Neural Network to model the influences of the events. Investors emotions play a major role in the stock investments, public moods derived from Twitter may help us to predict the stock price. We can know if the tweet is positive or negative according to the stock moments, hence aggregating sentiment per day may give us required results. However, this method alone cannot be enough to predict the stock movement, hence we also implement the RNN based approach and later combine these two approaches to predict the final value. We May call this as Multi-source multiple instance model.

4. MULTI-SOURCE MULTIPLE INSTANCE MODEL

In this section, we are going to propose the Multi-Source Multiple Instance (M-MI) framework.

4.1 Existing system

Stock Market depends upon various factors, such as Trades, events and the emotions of public or investors. Thus, using a single data source may not be enough to predict accurate prices. Hence, we decided to use multi-source data model approach. Especially relying on Social network posts and historical trading data. We also try to obtain the key factors that may have a vast influence on the market movement. These factors help us for further analysis during our process of developing this system.

There already existing various Stock market prediction machine learning projects but they are all based upon single source to predict the prices. Those systems consist mostly of one approach to problem-solving. They are mostly based upon neural networks or machine learning algorithms or upon normal analysis of data. This approach works well for a few instances of time but might not be conservative with growing data and real-time changes in events in the world. Hence, we propose multi-source, multiple instance model.

Although these two methods can work, they are partially efficient. Therefore, we are going to combine these both approaches to predict an accurate or a better Stock price movement.

4.2 The proposed approach

The inputs of the framework are the stock quantitative data and the tweets. We first use the sentiment analyzer to obtain the required sentiments from twitter. Then the extracted sentiments and the stock quantitative data are fed into the M-MI model. The M-MI model consists of given group labels, which are assumed to be an association function of the instance-level label. The goal is to predict the label for the multi-source group that indicates the rise or decline of the stock market.

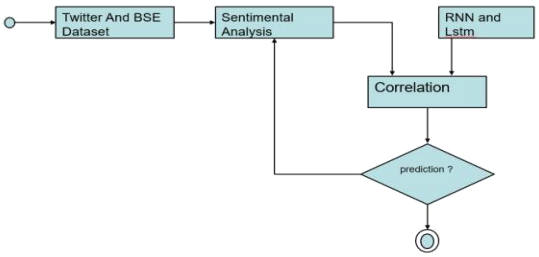

Fig. 1: Architecture diagram of our proposed model

The above figure explains the working pre-plant of the model.

5. SENTIMENT EXTRACTION

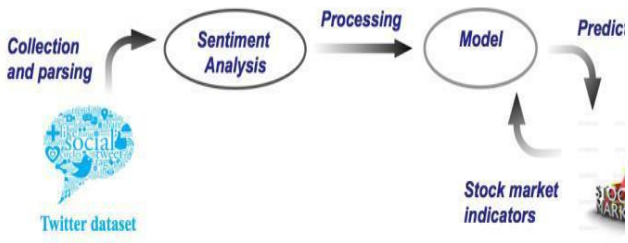

Fig. 2: Sentiment extraction

Twitter is a treasure trove of sentiment, people around the world output thousands of reactions and opinions on every topic it's like one of the big psychological databases that's constantly being updated and we can use it to analyse sentiments using sentimental analysis well when we receive tweet as input text. Then we will first want to split it up into several words or sentences, this process is called tokenization because we're creating small tokens from big texts we can just Count the number of times each word shows up once a text is tokenized this is called the bag of words model then we could look up the sentiment value for each word from a sentiment lexicon that has all pre-ordered to classify the total sentiment value of our tweet. As a process, we need to register to Twitter to access twitters API. An API is a programming interface gateway that lets you access some server's internal functionality in our case Twitter so we'll be able to read and write tweets. Hence able to authenticate or verify our identity with Twitter. Then we are going to use required dependencies such as text blob. This will help us to perform actual sentimental analysis using natural language processing techniques at the background. Sentiment attribute of the analysis using text blob will let us know the relativity of the

text and also its subjectivity measure showing how much opinion is versus the factual break.

It is recommended that we extract all the hashtags related to stock market movements and the data clean them properly to use them to fetch the right tweets from twitter and later use sentimental analysis to predict the emotions.

6. RECURSSIVE NEURAL NETWORKS

We're going to build a deep learning model to train a model that uses historical data to train itself with the above approach to predict the stock movement. For our training data will be using the BSE closing and opening prices from January 2002 until the present.

Using Yahoo Finance API just like Twitter we can get a copy of Historical data od BSE in CSV format to ease things up while training a model. This dataset consists of a series of data points indexed in time order or a time series. As our goal will be predicting the closing price for any given date after training. We can load the data as an array of values after normalizing them rather than being those values directly into our model, this will improve the convergence. Since this will reflect percentage changes from the starting point, we will divide each price by initial price and subtract one when our model later makes a prediction using normalization. In Real World, we first initialize a sequential value to a linear stack of layers and then we will add our first layer which is an LSTM layer. As the words are usually learned in a sequence it's called conditional memory which means going back word may be hard but not impossible. Feedforward neural nets don't accept thick sigh vectors as an input. So feed-forward neural Nets the hidden layers wait are based only on the input data but in a recurrent that the hidden layer is a combination of input data at the current timestamp and the hidden layer at the previous time step is constantly changing as it gets more inputs and the only way to reach these hidden states are with the correct sequence of inputs, this is how a memory is incorporated in and we can model this process mathematically so this hidden state at a given time that is a function of the input at that same time step modified by a weight matrix like the ones using feed-forward Mets added his State of the previous time step x its own hidden state to hidden state matrix otherwise known as a transition matrix and because it's feedback loop is occurring at every time step in the series each hidden state has traces of not only the previous hidden state but also of all of those that preceded it that's why we call it recurrent in a way we can think of it as copies of the same network each passing a message to the next. memory cells each cell has been implicated in output gate and an internal state that feeds into itself across time steps with a constant weight of one this eliminates the vanishing gradient problem since any gradient that flows into the self-recurring units during backdrop is preserved indefinitely since errors x 1 still have the same value HP is an activation function like signaling during the forward path to implicate learns when to let activation packed into the cell and the output learn to let activation pass out of it during the backward pass the output get learns when to let error flow into the cell and implicate one's going to let it flow out of himself through the rest of the network so despite everything else in a recurrent that staying the same doing this more powerful update equation for our hidden state results in our network being able to remember long term dependencies using LSTM. Later using LSTM we will set our input dimension to 21 to return sequences to true means these layers output is always set into the next layer all its activations can be seen as a sequence of predictions the first layer has made from the input sequence will add twenty percent drop out to this layer then initialize our second layer as another LSTM with 100 units and its output is only fed to the next layer at the end of the sequence doesn't help put a prediction.

for the sequence instead a prediction vector for the whole input sequence will use the linear dense later to aggregate the data from the prediction into one single value then we can compile our model using a popular loss function called mean squared error and use gradient descent our optimizer labelled RMS prop will train our model with the function then we can test it to see what it predicts for the next 50 steps at several points in our graph and visualize it using that.

Table 1: Shows the Glimpse of historical data collected from yahoo finance website using its API.

Serial No.	High	Close	Open
0	0.012630	0.01262	0.01263
1	0.012640	0.01264	0.01256
2	0.012594	0.01258	0.01255
3	0.012530	0.01253	0.01250
4	0.012530	0.01251	0.01252

Let's now understand RNN and LSTM mathematically:

Elman-network:

$$h_t = \sigma_h(W_h x_t + U_h h_{t-1} + b_h)$$
$$y_t = \sigma_y(W_y h_t + b_y)$$

Jordan network:

$$h_t = \sigma_h(W_h x_t + U_h y_{t-1} + b_h)$$
$$y_t = \sigma_y(W_y h_t + b_y)$$

LSTM with a forget gate:

$$f_t = \sigma_g(W_f x_t + U_f h_{t-1} + b_f)$$
$$i_t = \sigma_g(W_i x_t + U_i h_{t-1} + b_i)$$
$$o_t = \sigma_g(W_o x_t + U_o h_{t-1} + b_o)$$
$$c_t = f_t \circ c_{t-1} + i_t \circ \sigma_c(W_c x_t + U_c h_{t-1} +$$
$$h_t = o_t \circ \sigma_h(c_t)$$

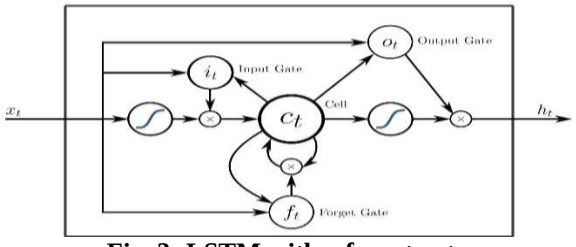

Fig. 3: LSTM with a forget gate

7. COMPARISON METHODS

Now after completing the sentimental analysis and Usage of Neural networks to train a model it's time to compare both the outputs in order to predict the final stock market movement.

For this approach, we have used support vector machines and it has been tested upon using only this algorithm. Although various other methods can be tried.

SVM: the standard support vector machine is used as a basic prediction method. During the training process, the label assigned to each instance and each group is the same as its multi-source supergroup label. During the prediction phase, we obtain the predicted label for each of the instances and then average the labels as the final label of the super group. After this process is completed, we would be able to see a graph just like RNN but with a slightly better result than the previous one.

8. PREDICTION RESULTS

After the Usage of all the above-stated algorithms and models, figure 4 shows how our end result of the predicted stock movement of BSE would look like with comparison of these models. This graph may change depending upon the change in the current sentiment of tweets from twitter although the historical data remains constant**.**

Fig. 4: Showing the end result

9. CONCLUSION

In this paper, a Multi-source Multiple Instance model using sentimental analysis and RNN is proposed which can predict the stock market movement. Although Mi model is used it might not predict accurate price prediction at the end though there are multiple instances-based approaches used as There may be problems such as overfitting or under fitting of data which might lead to Noise in data set and also affect the cost function values. Hence the accurate results cannot be expected.

10. REFERENCES

[1] Eugene F Fama. The behavior of stock-market prices. The Journal of Business, 38(1):34–105, 1965.

[2] Sanjiv R Das and Mike Y Chen. Yahoo! for Amazon: Sentiment extraction from small talk on the web. Management Science, 53(9):1375–1388, 2007.

[3] Jianfeng Si, Arjun Mukherjee, Bing Liu, Qing Li, Huayi Li, and Xiaotie Deng. Exploiting topic based twitter sentiment for stock prediction. In Proceedings of The 51st Annual Meeting of the Association for Computational Linguistics (ACL 2013), pages 24– 29, 2013.

[4] William Yang Wang and Zhenhao Hua. A semiparametric Gaussian copula regression model for predicting financial risks from earnings calls. In Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (ACL 2014), pages 1155–1165, June 2014.

[5] Shimon Kogan, Dimitry Levin, Bryan R Routledge, Jacob S Sagi, and Noah A Smith. Predicting risk from financial reports with regression. In Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics, pages 272–280, 2009.

[6] Ronny Luss and Alexandre d'AZAspremont. Predicting abnormal returns from news using text classification. Quantitative Finance, 15(6):999–1012, 2015.

[7] Robert Rougelot Prechter. The wave principle of human social behavior and the new science of socionomics, volume 1. New Classics Library, 1999.

[8] John R Nofsinger. Social mood and financial economics. J. Finance, 6(3):144–160, 2005.

[9] Jinbo Bi and Xin Wang. Learning classifiers from dual annotation ambiguity via a min-max framework. Neurocomputing, 151:891–904, 2015.

[10] Sihong Xie, Wei Fan, and Philip S Yu. An iterative and re-weighting framework for rejection and uncertainty resolution in crowdsourcing. In Proceedings of the 2012 SIAM International Conference on Data Mining pages 1107–1118, 2012.

[11] Quoc V Le and Tomas Mikolov. Distributed representations of sentences and documents. In Proceedings of the 31st International Conference on Machine Learning (ICML-14), pages 1188–1196, 2014.

[12] Frederik Hogenboom, Flavius Frasincar, Uzay Kaymak, and Franciska De Jong. An overview of event extraction from text. In Workshop on Detection, Representation, and Exploitation of Events in the Semantic Web (DeRiVE 2011) at Tenth International Semantic Web Conference (ISWC 2011), volume 779, pages 48–57. Citeseer, 2011.

[13] Ryo Akita, Akira Yoshihara, Takashi Matsubara, and Kuniaki Uehara. Deep learning for stock prediction using numerical and textual information. In Computer and Information Science (ICIS), 2016 IEEE/ACIS 15th International Conference on, pages 1–6, 2016.

[14] Thin Nguyen, Dinh Phung, Brett Adams, and Svetha Venkatesh. Event extraction using behaviors of sentiment signals and burst structure in social media. Knowledge and information systems, pages 1–26, 2013.

[15] Xiao Ding, Yue Zhang, Ting Liu, and Junwen Duan. Using structured events to predict stock price movement: An empirical investigation. In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 1415– 1425, 2014.

[16] Xiao Ding, Yue Zhang, Ting Liu, and Junwen Duan. Deep learning for event-driven stock prediction. In

[17] Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence (IJCAI 2015), pages 2327–2333, 2015.

[18] Johan Bollen, Huina Mao, and Xiaojun Zeng. Twitter mood predicts the stock market. Journal of computational science, 2(1):1–8, 2011.

[19] Masoud Makrehchi, Sameena Shah, and Wenhui Liao. Stock prediction using event-based sentiment analysis. In Web Intelligence (WI) and Intelligent Agent Technologies (IAT), 2013 IEEE/WIC/ACM International Joint Conferences on, volume 1, pages 337–342, 2013.

[20] Thien Hai Nguyen and Kiyoaki Shirai. Topic modelling based sentiment analysis on social media for stock market prediction. In Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics (ACL 2015), pages 1354–1364, 2015.

[21] Qing Li, LiLing Jiang, Ping Li, and Hsinchun Chen. Tensor-based learning for predicting stock movements. In The Twenty-Ninth AAAI Conference on Artificial Intelligence (AAAI-15), pages 1784– 1790, 2015.

[22] Thomas G Dietterich, Richard H Lathrop, and Tomás Lozano-Pérez. Solving the multiple instance problems with axis-parallel rectangles. Artificial intelligence, 89(1):31–71, 1997.

[23] Jaume Amores. Multiple instance classification: Review, taxonomy and comparative study. Artificial Intelligence, 201:81–105, 2013.

[24] Guoqing Liu, Jianxin Wu, and Z-H Zhou. Key instance detection in multi-instance learning. In Asian Conference on Machine Learning, pages 253– 268, 2012.

[25] Dimitrios Kotzias, Misha Denil, Nando De Freitas, and Padhraic Smyth. From group to individual labels using deep features. In Proceedings of the 21st ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD 2015), pages 597–606, 2015.

[26] Ji Feng and Zhi-Hua Zhou. Deep miml network. In The Thirty-First AAAI Conference on Artificial Intelligence (AAAI-17), pages 1884– 1890, 2017.

[27] Yue Ning, Sathappan Muthiah, Huzefa Rangwala, and Naren Ramakrishnan. Modelling precursors for event

forecasting via nested multi-instance learning. In Proceedings of the 22Nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD 2016), pages 1095–1104, 2016.

[28] Geoffrey E Hinton and Ruslan R Salakhutdinov. Reducing the dimensionality of data with neural networks. science, 313(5786):504– 507, 2006

