Deep Learning Based Forecasting in Stock Market with Big Data Analytics

Gozde Sismanoglu
Istanbul Kultur University
Computer Engineering Department,
34158, Istanbul, Turkey
gozde.sisman@hotmail.com

Mehmet Ali Onde
Istanbul Kultur University
Computer Engineering Department,
34158, Istanbul, Turkey
malionde@gmail.com

Furkan Kocer
Istanbul Kultur University
Computer Engineering Department,
34158, Istanbul, Turkey
furkankocer.tr@gmail.com

Ozgur Koray Sahingoz
Istanbul Kultur University
Computer Engineering Department,
34158, Istanbul, Turkey
o.sahingoz@iku.edu.tr

Abstract— **In recent years, due to the technological improvements in computers' hardware and enhancements in the machine learning techniques, there are two increasing approaches for problem-solving as the use of "Big Data" and "Parallel Processing". Especially with the emergence of Deep Learning algorithms which can be executed parallelly on multi-core computing devices such as GPUs and CPUs, lots of real-world problems are resolved with these approaches. One of the most critical application areas in the Financial Market especially sits on Stock Markets. In this area, the aim is trying to predict the future value of a specific stock by looking at its previous financial data on the exchange process in the market. In this paper, we proposed a system that uses a Deep Learning based approach for training and constructing a knowledge base on a specific stock such as "IBM". We get time series values of the stock from the New York Stock Exchange which starts from 1968 up to 2018. Experimental results showed that this approach produces very good forecasting for specific stocks.**

Keywords—deep learning; big data; forecasting, stock exchange

I. Introduction

The most important role of investors is to analyze the movements of financial markets and to make an accurate prediction. Since decades, investors have tried to use some methodologies and techniques for increasing the profit. In the analysis of these movements was examined under two categories. Firstly, technical analysis can be explained as predicting targetable price movements based on past price movements. This type of analysis is the method of analysis used in the markets where volatility is high, such as FOREX market (Foreign Exchange Market). Secondly, fundamental analysis method predicts the expected price movements of financial data based on economic, environmental, political and other factors as well as statistical data in the analysis.

On account of the use of increasing and developing technology, the number of operations performed instantaneously increased in a direct proportion. Attempts to estimate stock prices along with the increasing number of transactions have also been the subject of research for a long time, and some methods have been proposed in various academic researches. However, results show that no method alone has achieved the desired success. In order to assist investors in general, the desired systems should advise on the best possible action as soon as possible. Therefore, some decision support system which is trained with some learning mechanism is a good solution for this.

In the computer science field, there are many works on this subject which use different learning approach for the training of the system, lots of them focused on the use of the neural network approach. However, in recent years, with the extended use of powerful computers and being able to access a huge size of data, Deep Learning is one of the most attractive research areas for using different real-world application areas.

Recurrent neural networks (RNN), which is one of the important deep learning models, has proven its strength in sequential data such as time series in many academic studies. Based on this knowledge, we have reached the best results with Long Short-Term memory which is one of the most successful RNNs architecture. Therefore, we used LSTM in our work.

The purpose of this project is to use the stock market data we have in order to estimate the high-volume financial time series on deep learning. Our main aim is to reduce RMSE loss to achieve a better-quality result.

Rest of the paper is organized as follows. In the next section, some background knowledge and literature survey are presented. Section 3 enlightens the details about the proposed system. The experimental results with the related parameters are shown in Section 4. Finally, the paper is concluded with showing some future works that are aimed to conduct.

978-1-7281-1013-4/19/$31.00 ©2019 IEEE

II. Background Knowledge

A. Machine Learning

Machine learning (ML) is a subset of artificial intelligence (AI). The main goal of machine learning is detecting, sensing and learning situation as human or better than the human in some cases. One of the key points of machine learning is achieving higher performance by minimizing human interaction. It is the field of computer science that can analyze and interpret data patterns and structures while making this decision. To explain it in a simpler language, if it is necessary to learn the machine, an algorithm is needed to analyze the data and make a conclusion with these analyzes. Any machine learning algorithm is a set of hypotheses that are used to find the most appropriate model without taking training data before it is retrieved.

B. Deep Learning

Deep Learning is a kind of machine learning algorithms that are inspired by the brain's structure and its functions which is called Artificial Neural networks. As in biological neurons, artificial neurons have received input signals, these signals are collected and processed, and the outputs are forwarded.

Especially with the development of the internet, millions of data are produced and stored in huge dimensions every second in the digital environment. These data are called "Big Data." Old learning algorithms have not been able to demonstrate performance in dealing with big data. Deep learning systems have achieved efficient results using these big data effectively. Different from the other fields, Deep learning learns to perform classification features and tasks directly from data. Deep learning models can be trained using many neural network architectures to achieve human-level learning and even exceed that level.

Deep networks allow the system to be trained with large data using high-end GPUs. As a result, there is a need for high-level hardware for deep learning. This approach is the best fit for especially big data analysis type problem solving as used in [10].

C. Related Works

The forecasting of financial time series is of primary importance in the economic world for investors. In recent years, there has been growing amount of literature in this field. The literature review in [1] shows that the proposed model basically creates a sale and buying strategy in the system called the A-Trader system. A-trader aims to make trading decisions in the Forex market. This system supports high-frequency trading. For comparison between classical neural networks and deep learning model predictions are examined to show the quality. Multi-Layer Perceptron (MLP) model uses the function of sigmoid activation and back- propagation learning algorithm. The result of the MLP model was not satisfiable; for this reason, a more complex model needed to be developed. As a result, the network architectures used in the A-Trader system based on Convolutional Neural Networks (CNN). In order to compare with other models, CNN resulted in the best error rate. Thus, this project has shown that CNN is better than MLP for this model.

With rapid growth of the internet and computer technologies, the thickness of trading in the stock market increased in seconds. In [4] authors proposed a general framework that uses LSTM and CNN for heavy training to make high-frequency stock forecasting.

This study demonstrates the loss of estimation error loss and direction estimation and demonstrates that productive reverse training can be used successfully to combine these estimation losses to produce satisfactory predictive results,

In some researches dataset is used as real time dataset with a high frequency data by getting into account changes in seconds even in milliseconds. With the use of this property, either the size of dataset or the requested execution time changes. As the number of data, which are generated every second is increasing rapidly, some authors implemented a Deep Neural network system as in [5]. Their dataset consists of tick by tick (TBT) data where one tic corresponds to one activity. It could be a transaction, a bid or ask price or any other activity. Approximately, 14 million data points per day has been used for their system. The paper analyses the performance of its models based on the estimation accuracy as well as the predictive speed for full-day trading simulations. In this research, the importance of having sufficient computational power to continue to update the model weights of each product in parallel while making predictions in real time was clearly stated.

Most studies like in [2], as well as current work, focus on presenting a high-frequency strategy based on Deep Neural Networks (DNNs). DNN estimations have been used to generate a stock merchandising strategy that makes buying or selling decision based on whether the next estimated final close price is below or above. Best available DNN has an aspect ratio of 66% accuracy. Backpropagation algorithm was used in the study. This research aims at overcoming the problem of exponentially dilution the error while passing through hidden layers which is the main problem of the back-propagation algorithm. Thus, this project has shown that DNNs can offer a potential solution for forecasting.

Some authors in [8] have also suggested that using the three models. Google Trends, which proved that not being effective factor in estimating the targeted data set price. Putting together the results obtained, this research approved the imbalance on the market and high variation of machine learning algorithms on estimating stocks. Not every algorithm gives the same results for each organization. The importance of the selection of the appropriate algorithm is emphasized in this work.

III. Proposed System

A. Used Dataset

We evaluate the performance of the proposed method based on the data collected from IBM stock prices [9]. The data in this work consists of the full historical daily price and volume data for all US-based stocks trading on the NYSE, NASDAQ, and NYSE MKT, ranging from January 2, 1968, to April 09, 2018. There are a totally 12648 trading days. We have not processed the last 300 data that we have. Thus we can perform our tests with that dataset. The sample size, which contains daily stock prices between 1968 and 2018, in this study dataset was not considered large enough for calling it Big Data. However, the evaluation of the data presented in this work shows that this is enough to get a sufficiently accurate solution with Deep Learning algorithms. In this dataset, we have 5 features to work as shown in Table I.

TABLE I. THE DEFINITION OF FEATURES

Feature	***Definition***
date	Day of the transaction
open	Opening price value of the specified day
close	Closing price value of the specified day
low	The lowest price value of the specified day
high	The highest price value of the specified day

B. *Recurrent Neural Networks*

Recurrent neural networks (RNN) are used to model sequential data such as sentences, audio, music and time series data. The main idea is to use sequential information. We can say that all inputs and outputs are independent of each other in a traditional neural network. The reason that they are called repetitive is that they perform the same task over and over for each element of an array. Another way to explain this structure is that they carry "memory" that collects data about what has been calculated so far. The diagram shown in Fig.1 shows the RNN structure that opens to a complete network.

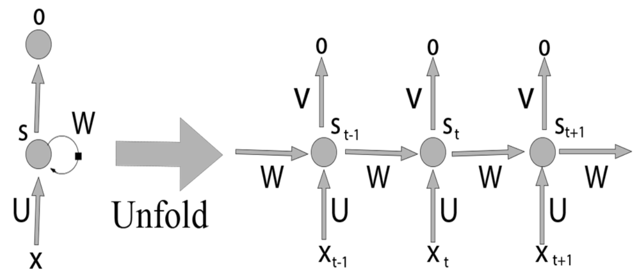

Fig. 1. Traditional RNN Schema.

C. *LSTM Cell*

LSTM structures have been developed to overcome the problem of vanishing / exploiting gradient. As stated in Hochreiter et al. [10]. This problem is a more difficult process to train recurrent neural network models than other models. After a while, RNN face with the network will begin to "forget" the first input, as information is lost at each step going through the RNN. Since the length of time series, the complexity of this problem is increasing. Owing to these problems, for our networks, some sort of "Long-term Memory" is needed.

In complex problems, the number of layers must be greater. However, undesirable results can be achieved in such models. The first layers of the model are slow to learn, and the later layers are quick to learn. On the contrary, it is called the Exploding Gradient when the first layers of the model are learned to be fast, and the next layers are slow to learn. Such situations cause the model not to achieve the expected results. LSTM layer structures have been developed to solve such problems and accurately create Recurrent artificial neural network models.

In the LSTM layers where the neurons have their own memory, the past time data is stored and used in the development process of the model which is shown in Fig. 2. In the training process, which data are stored in memory will be shaped. In this way, even the old data interacts with the new data and creates a network structure that produces more effective results.

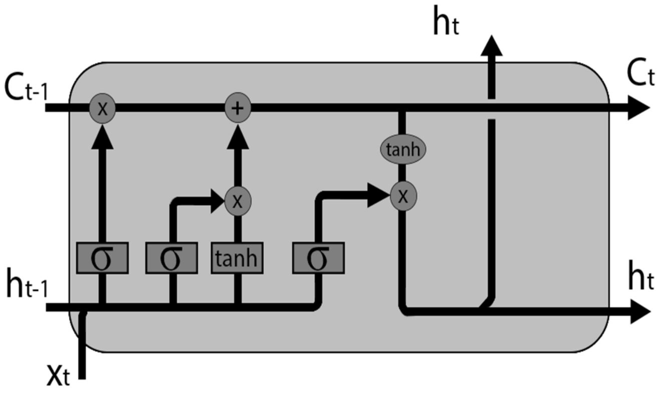

Fig. 2. LSTM Cell Schema.

IV. Experimental Results

The proposed model is tested in the hardware as follows. Hardware Components: 9 MB Cache, Intel 8 core, 2.3GHz processor with 8 GB RAM and Video card 4GB GDDR5 nVIDIA® GeForce® GTX1050 128-Bit DX12. Programming Language: Python (v .3.5.2) is chosen due to its libraries for machine learning tasks. Integrated Development Environment (IDE): The choice of IDE falls on PyCharm (3.6) and extended with Anaconda modules. Machine Learning and Supporting Libraries: Keras package with the Tensorflow, open-source framework developed by Google, backend

The proposed stock market forecasting model has been implemented in Python using the Keras library with the Tensorflow backend. The "close" feature used as the target value that we want to evaluate the performance of the forecasting. Data processed to focus on daily changes. We divided the data into two sections: training data and test data.

Time series prediction performance measures provide essential information about the capability of the forecast model that expected to make the predictions. As a performance analysis Root Mean Squared Error (RMSE) calculated by the mathematical formula as shown in Equation 1.

$$RMSE = \sqrt{\frac{1}{n}\sum_{j=1}^{n}\left(y_j - y_j\right)^2} \quad (1)$$

For testing the 300 days prediction the prediction and the original values are shown in Fig.3. If we compare the directional accuracy, we will get a better result from this graph.

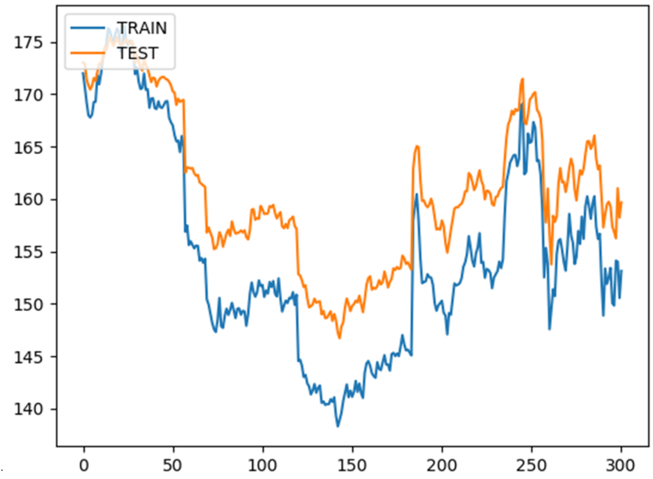

Fig. 3. Test data.

As a test scenario, firstly we trained our system with the previous data values, and by using the features as depicted above section, we checked the prediction accuracy by using the Root Mean Squared Error values by comparing real stock data with the predicted value. The reached RMSE values are depicted in Fig. 4.

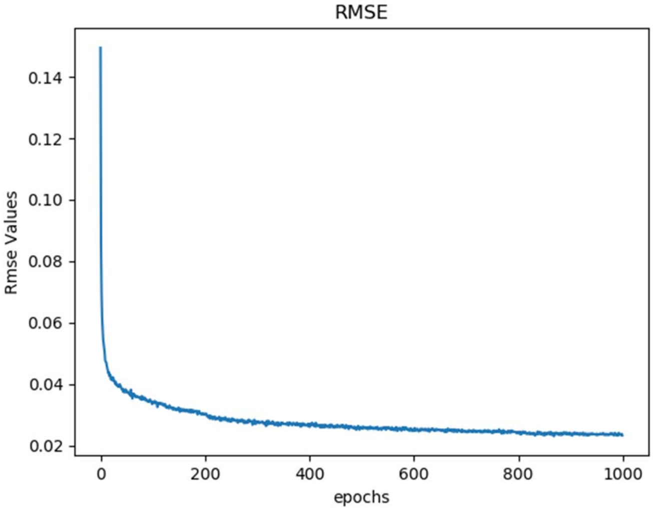

Fig. 4. RMSE changes.

Finally, the result of the test is briefly shown in Table II. As can be seen from this table the reached RMSE value is acceptable and the proposed model can be used in some real-world cases.

TABLE II. EXPERIMENT COMPONENTS

Epochs	*Batch Size*	*Duration*	*RMSE*
1000	32	9.49 minutes	0.04

V. CONCLUSION AND FUTURE WORKS

Making accurate forecasting in stock markets is a very challenging task due to the nonlinearity of the financial time series. Some researchers in this area say that stocks prices behave in a random walk manner. However, technical analysts insist that future prices can be predicted somehow by considering some current values. In this paper, as one of the deep learning approaches, LSTM network is applied to a large-scale stock market NYSE, NASDAQ, and NYSE MKT, ranging from January 2, 1968, to April 09, 2018 for predicting the future values. With the use of LSTM architectures, some hidden dynamics of the market can be captured, and efficient predictions can be possible. Proposed model results in an acceptable root mean square error (RMSE) value as 0.04.

As a future work, we aimed to take into consideration some new features for making more accurate predictions. Additionally, some parameter values can be set by using some optimization algorithms to increase the efficiency of the system. Finally, for using a huge parallelization, system execution can be transferred to GPU structures as mentioned in [11].

REFERENCES

[1] J. Korczak and M. Hernes, “Deep Learning for Financial Time Series Forecasting in A-Trader System,” *Proceedings of the 2017 Federated Conference on Computer Science and Information Systems,* 2017.

[2] A. Arévalo, J. Niño, G. Hernández, and J. Sandoval, “High-Frequency Trading Strategy Based on Deep Neural Networks,” *Intelligent Computing Methodologies Lecture Notes in Computer Science, pp. 424–436,* 2016.

[3] A. Hasan, O. Kalipsiz, and S. Akyokus, “Predicting financial market in big data: Deep learning,” *2017 International Conference on Computer Science and Engineering (UBMK)*, 2017.

[4] X. Zhou, Z. Pan, G. Hu, S. Tang, and C. Zhao, “Stock Market Prediction on High-Frequency Data Using Generative Adversarial Nets,” *Mathematical Problems in Engineering*, vol. 2018, pp. 1–11, 2018.

[5] P. Ganesh and P. Rakheja, "Deep Neural Networks in High-Frequency Trading," IEEE, 2018.

[6] H. Gunduz, Z. Cataltepe, and Y. Yaslan, “Stock market direction prediction using deep neural networks,” *2017 25th Signal Processing and Communications Applications Conference (SIU),* 2017.

[7] S. Pyo, J. Lee, M. Cha, and H. Jang, “Predictability of machine learning techniques to forecast the trends of market index prices: Hypothesis testing for the Korean stock markets,” *Plos One*, vol. 12, no. 11, 2017.

[8] B. Marjanovic, (11.10.2017). Huge Stock Market Dataset. Retrieved from https://www.kaggle.com/borismarjanovic/price-volume-data-for-all-us-stocks-etfs

[9] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," Neural Computation, vol. 9, no. 8, pp. 1735–1780, 1997.

[10] G. Karatas, O. Demir and O. K. Sahingoz, "Deep Learning in Intrusion Detection Systems," 2018 International Congress on Big Data, Deep Learning and Fighting Cyber Terrorism (IBIGDELFT), ANKARA, Turkey, 2018, pp. 113-116. doi: 10.1109/IBIGDELFT.2018.8625278

[11] S. I. Baykal, D. Bulut and O. K. Sahingoz, "Comparing deep learning performance on BigData by using CPUs and GPUs," 2018 Electric Electronics, Computer Science, Biomedical Engineerings' Meeting (EBBT), Istanbul, 2018, pp. 1-6. doi: 10.1109/EBBT.2018.8391429