International Stock Index Prediction Using Artificial Neural Network (ANN) and Python Programming

Ashkan Safari
Faculty of Electrical and Computer Engineering
University of Tabriz
Tabriz-Iran
Ashkansafari.unv@gmail.com
orcid.org/0000-0002-1780-7615

Amir Aminzadeh Ghavifekr
Faculty of Electrical and Computer Engineering
University of Tabriz
Tabriz-Iran
Aa.ghavifekr@tabrizu.ac.ir
orcid.org/0000-0002-0345-4150

2021 7th International Conference on Control, Instrumentation and Automation (ICCIA) | 978-1-6654-0350-4/20/$31.00 ©2021 IEEE | DOI: 10.1109/ICCIA52082.2021.9403580

***Abstract*— The stock market is one of the best channels for financial development that requires a high accuracy prediction of the trades. This subject needs some technical skills and experience to achieve the best result. This paper represents a tuned Python console program based on the Neural Network (NN), and the Artificial Intelligence (AI) to predict future price in a qualified and quantized way with high accuracy and close to real. New ideas implemented in this paper are combining AI and NN model in the Python console system with a security shell that works with voice and PIN to authenticate the user. It has Cross-Platform capability and supports cryptocurrencies price and their predictions. This program enables the user to have a duplication of the final data in his/her given email. The proposed approach presents the influence of AI and Machine learning in nearly future predictions. This system can be used in the all kinds of subjects that include past time databases.**

***Index Terms*-- Artificial Intelligence, Artificial Neural Network (ANN), International Stock Prediction, LSTM, Python Programming.**

I. INTRODUCTION

Recent advances in the stock markets have caused significant effects on finance which can be more complicated to predict the indexes. Nowadays, most of people are directly or indirectly related to this subject and the more technology is developing the more they need to know and predict the indexes and this makes them be interested in index prediction. However, due to the quick changes in stock price, the prediction of stock price becomes a challenging task. Moreover, effects of the cryptocurrencies have increased this complexity [1]. These factors cause traders to go through using intelligent systems rather than using fundamental analysis to predict the price. Accordingly, traders can sell the index before value decline or buy before the price rises and this causes the trader to have much more profit. Also, it seems unbelievable for traders to replace their experience and professionalism with intelligent systems, but due to the remarkable amount of data and technological advancements of intelligent systems, algorithms, pattern recognition and Artificial Intelligence, it seems appropriate to use and even, combine them with the experience and professionalism. Since the significance of accurate information, Neural Networks (NN) have become one of the successful and efficient algorithms and models that are being used for modeling stock market behavior [2]. Artificial Neural Network (ANN) is a popular method which also incorporate technical analysis for making predictions in financial markets. One of the most practical methods in this area is Long Short Term Memory (LSTM) [3]. Pattern recognition is another method to predict the repeated patterns of a stock index over a period of time in the future [4].

In stock trading, it is very decorous that a model like NN provides a prediction nearly to the real price. Predicting the stock market is one of the processes that requires experience and reacquaint to have an accurate prediction. However, this process is qualitative and cannot be a complete prediction. AI and NN convert it to a quantity that means it can be used with mathematical approaches and results in a scalar number form that gives an amount with high accuracy and small Root Mean Squared Error (RMSE) values for future that is more reliable than qualitative predictions[5, 6]. Although this approach is useful, all users cannot apply it due to it is not implemented on an OS program. One of the Neural Networks is Real-Time Recurrent Learning (RTRL) network that is practical and able to store the information for later use to enhance the efficiency and a better way for modeling [7]. In [8] it is proved that the algorithms based on RNNs can be useful in financial market prediction. The program which is presented in this paper can be flexible and dynamic that means user can change the mark in a fraction of a minute. When it is incorporated with Python programming AI can be more advantageous than previous methods, and a wide range of people can use it. The NN is a means of performing machine learning, in which a computer learns to perform some tasks by analyzing training examples, and that is the base case of prediction, and the approach to quantitatively predict the stock market. This simple feature is used in this paper to make complicated approaches for accurate prediction close to the real value.

The organization of the paper is as follows: In section 2, we provide preliminaries and tools. The proposed framework is presented in section 3. The results are presented in section 4. Finally, section 5 draws a conclusion and further improvements.

978-1-6654-0350-4/21/$31.00 ©2021 IEEE

Authorized licensed use limited to: BOURNEMOUTH UNIVERSITY. Downloaded on June 30,2021 at 20:27:21 UTC from IEEE Xplore. Restrictions apply.

II. Neural Networks and Programming Language

The basic idea for prediction is usually based on the information from previously analyzed and confirmed data. Therefore, outputs will be useful to predict future data. When a neural network is used for predicting, results tend to be closer to the actual value, since the model can be frequently acclimated through model performance with a reduction of errors, and tolerances.

Our program is run under Python. That operates a wide range of applications such as business, scientific and numeric and educational applications, software, web and internet development, and desktop graphical user interfaces (GUIs). Keras is a Neural Network library written and developed in Python. It is qualified to superordinate Tensor Flow and designed to operate fast trial with artificial, and Deep Neural Networks. SKlearn is a machine learning library for Python programming. It accentuates prismatic classification, regression, and constellated algorithms including vector machine support, and database scan, and designed to sync, and perform with the Python numerical-scientific libraries for instance, NumPy.

NumPy is a professional mathematical library for Python programming that supports large, multi-dimensional, complicated arrays and matrices. Including a wide collection of high-level mathematical functions to operate on the arrays and matrices. Pandas is a software library written for Python and mostly used for data structures.

Tkinter is Python's de-facto standard GUI (Graphical User Interface) package and object-oriented interface. It is commonly used in GUI and object-oriented apps. Wx Python is a wrapper for the cross-platform GUI API (Graphical User Interface Application Programming Interface) (toolkit) and its widgets (which are written in C++) for the Python programming language. It is one of the alternatives to Tkinter GUI and implemented as a Python extension module (native code). Matplotlib is a plotting library for the Python programming language and its numerical mathematics and scientific packages *'NumPy'* and *'SciPy'*. It provides an OO (Object-Oriented) API for embedding plots into applications using GUI toolkits such as Tkinter, wxPython. Mplfinance is the Matplotlib utility package for the visualization and visual analysis of financial data. It's particularly used for candlestick plots in Python.

The OS (Operating System) module, provides functions for interacting with the operating system in the Python programming language. It comes under Python's standard utility modules. This module provides a portable way of using Operation System dependent functionality. The 'OS' and 'OS.path' modules include many functions to interact with the file system and the interaction with the Python environment.

III. Proposed Framework

This program proposes an attention-based short-term and memory model to predict the International Stock Price trend and all system runs on Python. The model consists of five layers: Shell Layer, Input Layer, Hidden Layer, Attention Layer and Output Layer. The shell layer authenticates the user. The input layer reads the input data that meet the input requirements. The hidden layer is correlated to the linear network thorough the LSTM unit. The attention layer prepares future amounts based on the predictions that are performed in the hidden layer. The output layer receives the final measured results to show for the user. The proposed framework is illustrated in Fig. 1 and the diagram of the LSTM method is shown in Fig. 2.

According to Fig. 2, after the user enters the input requirements, the data is received from the database with *Pandas* and put in the order with package *NumPy*. Then, it is rounded and analyzed with the Python Artificial Intelligence packages (*Keras, TensorFlow, and SKlearn*) and prediction is performed on it according to the LTMS algorithm. As a result, the train plot, prediction plot, and the future price will be shown to the user.

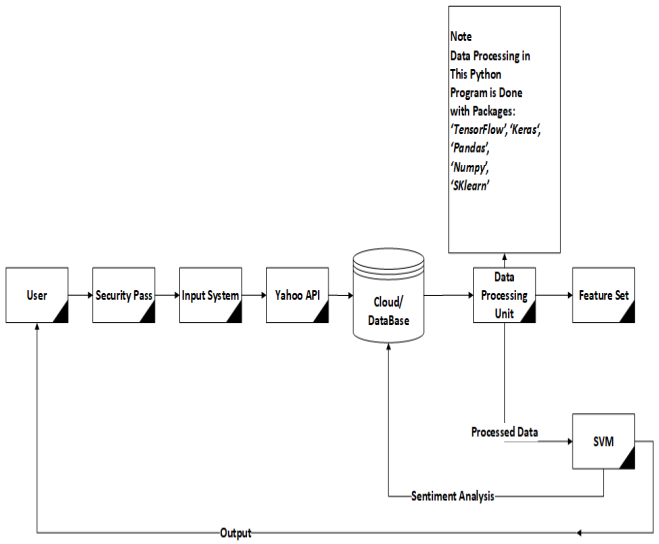


Fig. 1, Proposed Framework that illustrates the process from the initial

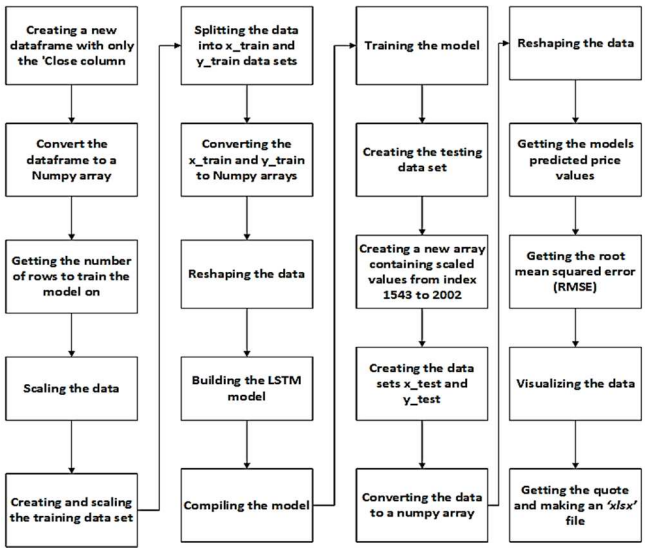


Fig. 2, Diagram of the prediction method based on the LSTM.

Authorized licensed use limited to: BOURNEMOUTH UNIVERSITY. Downloaded on June 30,2021 at 20:27:21 UTC from IEEE Xplore. Restrictions apply.

A. Shell Layer

In the shell layer, the user is authenticated with one of two optional approaches (PIN, and Passvoice), and after successful authentication, moves to the next layer. The diagram of this model is shown in Fig. 3.

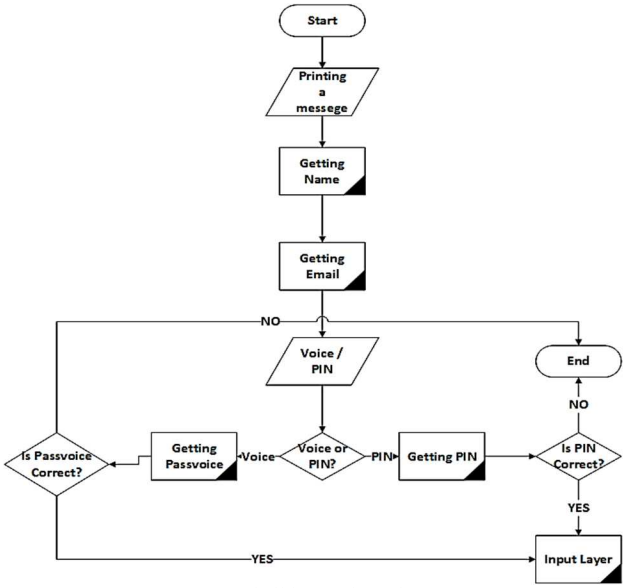


Fig. 3, Shell layer algorithm

B. Input Layer

After the shell layer, the user moves to the input layer and enters data: initial data, final data, future data to predict, primary credit. User's job finishes here, and according to the entrance, data is received from the database with package Pandas and put in the order with package NumPy. Then, it split to the train set and test set to be transferred to the next layer. The diagram of the layer is illustrated in Fig. 4.

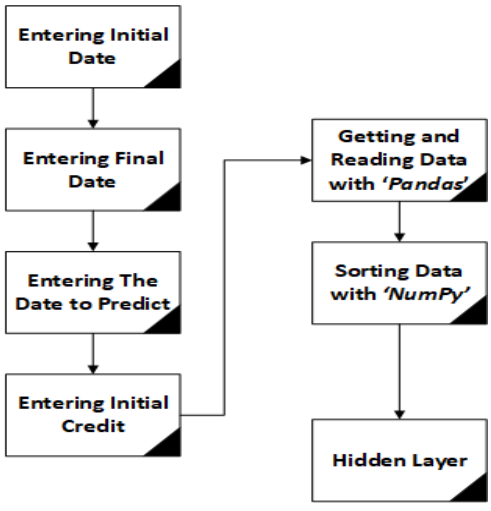


Fig. 4, Input layer block diagram

C. Hidden Layer

The Hidden Layer is the main and the most significant part of the program. The whole analysis and predictions performed with the LTMS algorithm, and RMSE equation are in this layer. The diagram of the hidden layer is shown in Fig. 5.

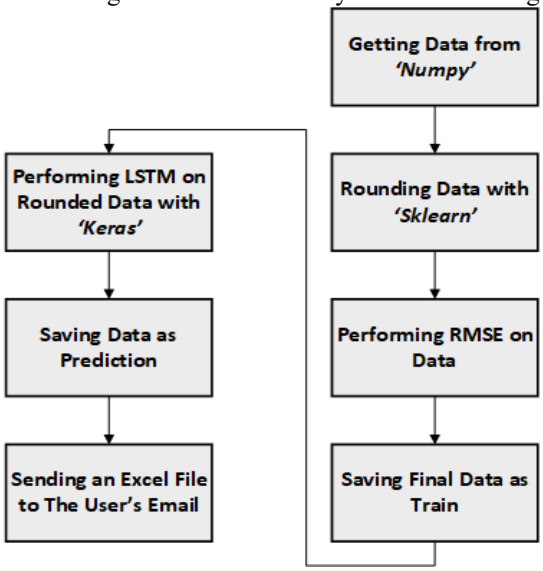


Fig. 5, Hidden layer block diagram

Depending on Fig. 5, when the calculating process finished, an Excel file including prices, and dates is sent to the email that the user had entered in Shell Layer. It makes the hidden layer more efficient and improved. According to Fig .4, RMSE is used in predictions for tolerance and error. The RMSE is defined as[9]:

$$\boldsymbol{RMSE} = \sqrt{\frac{\sum_{i=1}^{n} \left(Y_{i,Predicted} - Y_{i,Actual}\right)^2}{n}} \qquad (1)$$

According to Eq (1), n is the total data sample. $Y_{i,Predicted}$ and $Y_{i,Actual}$ respectively are predicted price, and actual price. The RMSE closer to 0 predicts less error. When the difference between entered initial data, and final data came bigger, RMSE tends to zero. As a result, the prediction is closer to the actual. This concept is proved in Experiment Results section. The RMSE in Python is calculated as:

```
rmse = np.sqrt(np.mean(((predictions - y_test) ** 2)))
```

D. Attention Layer

In the attention layer, the final calculated data received from the hidden layer is analyzed. The approach is performed by analyzing input data in the attention layer, the model, and the program can observe and learn the input data. The higher weights of data observed by the model training, the closer result is for the target value. As a result, the Price of the future date is predicted. Depending on performed experiments, and the weights of analyzed data, it predicts with a high accuracy

Authorized licensed use limited to: BOURNEMOUTH UNIVERSITY. Downloaded on June 30,2021 at 20:27:21 UTC from IEEE Xplore. Restrictions apply.

near to actual. The diagram of the attention layer is illustrated in Fig. 6.

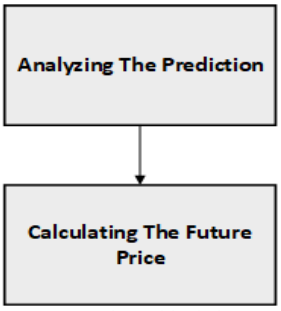


Fig. 6, Attention layer block diagram

E. Output Layer

Output Layer is the last layer of the system, it illustrates related plots, future price, profit, and future worth. The profit and future worth are calculated as:

$$i = ln\left(\frac{train["close"][int(End\_Date)]}{train["close"][int(Start\_Date)]}\right) \quad (2)$$

$$F = (i + p)^t \quad (3)$$

Here, F is the future worth, P, i, and t respectively are Primary credit, compound interest, and the number of periods. Periods are intended annual and the number of periods (t) is measured as:

$$t = (Year_{final}) - (Year_{initial}) \quad (4)$$

According to the code, the data is visualized and ready to be processed with *matplotlib* package and plotted. The diagram of the Output Layer is shown in Fig. 7.

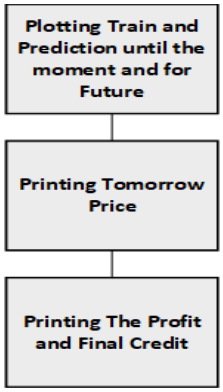


Fig. 7, Output layer block diagram

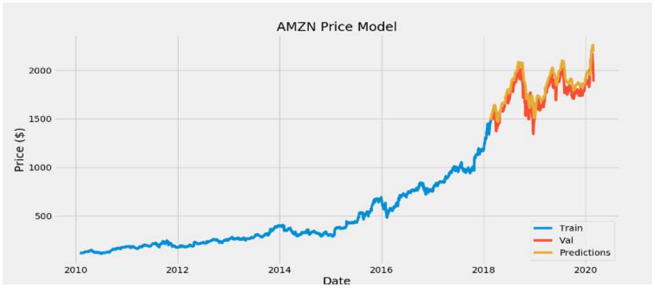
As well as the plot results, the others such as quote, interest, Future worth are shown in this layer.

IV. EXPERIMENTAL RESULTS

The model, and program are tested three times with different data (Shown in Table 1) for stock *AMZN*. The results are illustrated in Figs.9, 10, 11, 12, and 13.

Table 1

Primary year	Final year	Prediction Date	Range
2010	2020	2020/02/24	10
2018	2020	2020/02/24	2
2019	2020	2020/02/24	1



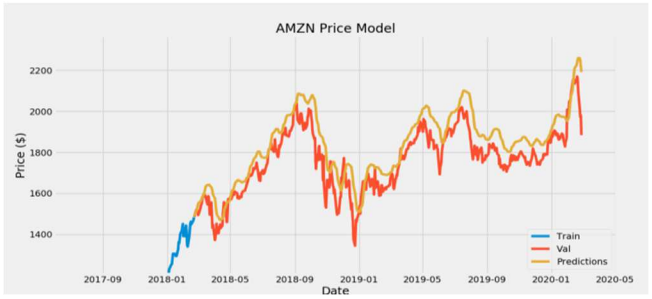
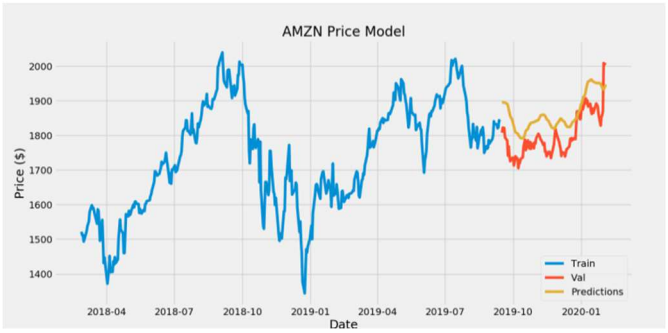


Fig. 8, Periodic Graph for $t = 10$ years

Authorized licensed use limited to: BOURNEMOUTH UNIVERSITY. Downloaded on June 30,2021 at 20:27:21 UTC from IEEE Xplore. Restrictions apply.

Fig. 9, Candlestick Graph for *t = 10* years

Fig. 11, Candlestick Graph for *t = 2* years



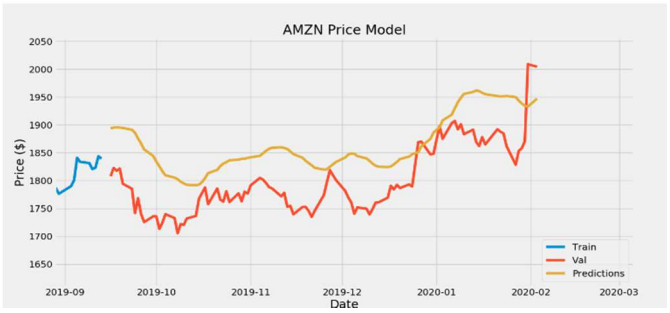
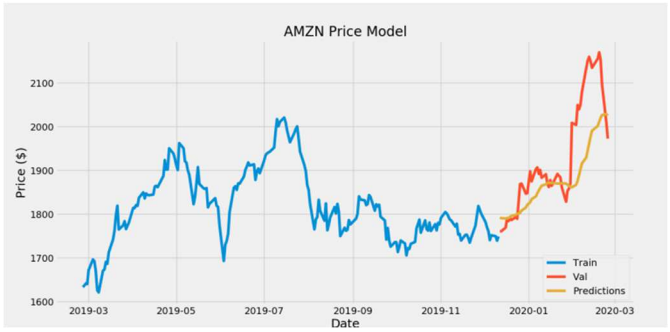


Fig. 10, Periodic Graph for *t = 2* years



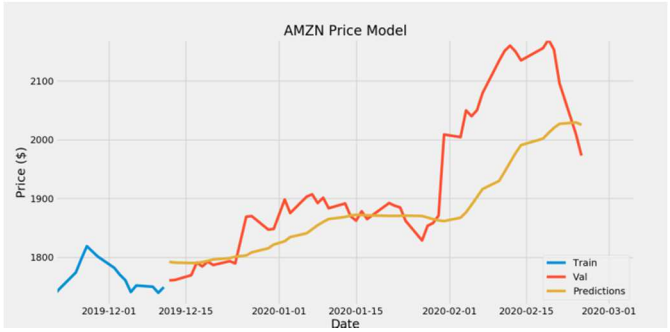


Fig. 12, Periodic Graph for *t = 1* year

Authorized licensed use limited to: BOURNEMOUTH UNIVERSITY. Downloaded on June 30,2021 at 20:27:21 UTC from IEEE Xplore. Restrictions apply.

Fig. 13, Candlestick Graph for *t = 1* year

According to Figs. 9 and 10, due to the high amount of received data, RMSE is low. In consequence, due to the difference of entered years is high (2010-2020), the plot of previous years (2010-2019) is not in a towering situation. In order, the prediction of the LSTM algorithm is extremely near to the actual and dependable. Thus, the price and prediction are nearly congruous.

Depending upon Figs. 11, and 12, due to the normal amount of received data, RMSE is normal. The difference of entered years (2018-2020). Therefore, the plot of previous years (2018-2019) is in a towering situation. In order, the prediction of the LSTM algorithm is normally near to the actual. As illustrated in Figs.11, 12, due to the less amount of received data, RMSE is high. Since the difference of entered years is low (2019-2020), the plot of the previous year (2019) is in a towering situation. In order, the prediction of the LSTM algorithm is lower than the actual.

In comparison to other ways and models, the presented model consists of two additional features that increase its efficiency. Moreover, the model is implemented on a Python console program that means it can be a cross-platform application. Additional information is shown in Table 2.

Table 2: Comparison of the proposed approach and the others

Item	Proposed Program	Other ones
An email of duplicated data	Sends to user	Not able
Cross-Platform capability	Able and Lighter	Able and Heavier
Operating System	Python 3.7	various
Security Layer	Included	Not Included
An Excel data file	Generates	Do not Generate

V. CONCLUSION

In this paper, for Stock Prediction based on the LSTM machine learning, the AI, Python Programming, and International Stock Market was carried out on analysis with the deep learning and AI frameworks in the Python operating system. As an experiment, the program used to predict the future price and the graph of the prediction plots in three different times for stock AMZN.

In this order, the results validate the possibility and correctness of the program and the prediction. The experiment compared with the prediction that indicated with classic ways, validates the correctness of the proposed Python program. Therefore, this method can be extended to different applications such as financial applications and educational applications.

The innovation of this paper can be summarized in 7 aspects:
(1) The proposed program can predict the price of cryptocurrencies such as Bitcoin (BTC) and Ethereum (ETH).
(2) The model presented in this paper is tuned with a flexible method that its accuracy increases when the user enters a wide range of data.
(3) In this paper, the model is approved with a secure layer that called Shell Layer, thus the user should be authenticated with voice or PIN.
(4) The proposed program has the ability to plot the candlestick graphs for the given data.
(5) When the data is converted to information with processing, and analyzing, an Excel file with *'xlsx'* format is sent to the email that the user has entered.
6) The Python program that is made with the paper method can be used by all kinds of users, for instance, a stock expert, or a scientific one.
(7) The proposed program can predict all kinds of subjects that include previous numeric data such as a house, or car pricing.

VI. REFERENCES

[1] D. Wei, "Prediction of Stock Price Based on LSTM Neural Network," in *2019 International Conference on Artificial Intelligence and Advanced Manufacturing (AIAM)*, 2019, pp. 544-547.

[2] M. Abdechiri, K. Faez, and H. Bahrami, "Neural network learning based on chaotic imperialist competitive algorithm," in *2010 2nd International Workshop on Intelligent Systems and Applications*, 2010, pp. 1-5.

[3] T. Mankar, T. Hotchandani, M. Madhwani, A. Chidrawar, and C. Lifna, "Stock market prediction based on social sentiments using machine learning," in *2018 International Conference on Smart City and Emerging Technology (ICSCET)*, 2018, pp. 1-3.

[4] L. Sayavong, Z. Wu, and S. Chalita, "Research on Stock Price Prediction Method Based on Convolutional Neural Network," in *2019 International Conference on Virtual Reality and Intelligent Systems (ICVRIS)*, 2019, pp. 173-176.

[5] M. Billah, S. Waheed, and A. Hanifa, "Stock market prediction using an improved training algorithm of neural network," in *2016 2nd International Conference on Electrical, Computer & Telecommunication Engineering (ICECTE)*, 2016, pp. 1-4.

[6] H. Abbasimehr, M. Shabani, and M. Yousefi, "An optimized model using LSTM network for demand

Authorized licensed use limited to: BOURNEMOUTH UNIVERSITY. Downloaded on June 30,2021 at 20:27:21 UTC from IEEE Xplore. Restrictions apply.

forecasting," *Computers & Industrial Engineering,* p. 106435, 2020.

[7] M. T. Motlagh and H. Khaloozadeh, "A new architecture for modeling and prediction of dynamic systems using neural networks: Application in Tehran stock exchange," in *2016 4th International Conference on Control, Instrumentation, and Automation (ICCIA)*, 2016, pp. 196-201.

[8] A. Moghar and M. Hamiche, "Stock Market Prediction Using LSTM Recurrent Neural Network," *Procedia Computer Science,* vol. 170, pp. 1168-1173, 2020.

[9] M. Vijh, D. Chandola, V. A. Tikkiwal, and A. Kumar, "Stock Closing Price Prediction using Machine Learning Techniques," *Procedia Computer Science,* vol. 167, pp. 599-606, 2020.

Authorized licensed use limited to: BOURNEMOUTH UNIVERSITY. Downloaded on June 30,2021 at 20:27:21 UTC from IEEE Xplore. Restrictions apply.