Recurrent Neural Network for Stock Market Forecasting using Long Short-Term Memory and an Analysis of How Social Media Affects Share Prices

SR Samarasuriya
Faculty of Science and Engineering,
University of Wolverhampton,
United Kingdom.

DVDS Abeysinghe
Faculty of Computing,
General Sir John Kotelawala
Defence University,
Ratmalana, Sri lanka

KGK Abeywardhane
Faculty of Computing,
General Sir John Kotelawala
Defence University,
Ratmalana, Sri lanka

ABSTRACT
With the increase in computing power and the popularity of machine learning (ML), it has become the norm to tackle more complex problems using ML. The stock market is known to be a highly volatile environment in which stock prices can fluctuate in an erratic manner. The main goal behind this study is to use a deep learning artificial intelligence model to understand and forecast future stock prices. An analysis was also done to assess the role of social media in the stock market price variation and to what extent, it impacts stock prices. The favored approach was to use a Recurrent neural network (RNN) composed of a Long Short-Term Memory (LSTM) model to predict the prices as it is the most suitable to work with time-series data. A successful model was deployed which showed a high level of accuracy and produced low values with regards to the loss function.

Keywords
Stock market prediction, RNN, LSTM

1. INTRODUCTION
The stock market can be considered as the bedrock of an economy and its success can greatly impact the lives of many people. Stock market prediction has long been an enticing prospect for shareholders and potential buyers alike, more so in recent times, as artificial intelligence has seen a rapid rise. Due to the inception of deep learning artificial intelligence models, the ability to incorporate non-linear data into such models and make predictions has become a reality. The stock market is an extremely volatile environment with unpredictable data gathering insights into such data and making predictions would be advantageous as it is likely to stimulate the economy by increasing the number of participants purchasing stocks. Moreover, an artificial intelligence model should not be used on its own but should rather be used to complement and reinforce existing tools and processes. Hence, the study attempts to create an AI model that is capable of predicting stocks with a high level of accuracy to aid the decision-making process when making financial decisions according to the projected predictions. Further, sentiment analysis has been conducted to assess the impact of social media and news on stocks and share prices in any way.

The proposed software model will be used in the stock markets to effectively predict future stock prices and how well it can be incorporated into existing statistical methods and tools to reinforce decision-making. With the rise of social media, more noise is added to stock prices as certain social events can either drive the price up or down therefore an analysis will be done as to the extent social media affects stock prices. Authors in [1] suggest in their work that there lies a significant impact of social media on the stock market. The primary source of data used in this study will be historical stock prices and social media news such as tweets from Twitter. Deep-learning artificial models will be associated with this project as they will be used to perform the stock price predictions and the sentiment analysis. A vital aspect of deep learning known as LSTM layers will be incorporated into the model due to their efficacy in working with time series data. The contribution of the study consists of a working model that will perform a future stock price prediction and a sentiment analysis using social media data. Users will engage with the final artifact through a web interface. The main programming language used is Python and the model will utilize existing frameworks and libraries within Python such as pandas, numpy, matplotlib, and scikit-learn. The TensorFlow library will form the core of the machine learning process and in-built methods and tools offered will be employed. A recurrent neural network will be used in tandem with LSTM layers to handle the time series data that is primarily used for the model.

2. LITREATURE REVIEW
Existing literature and studies carried out in the field of machine learning for stock prediction were widely available. The decision for choosing the most appropriate literature was based on matching certain criteria which is similar in nature to the proposed system. A focus was placed on choosing deep-learning stock prediction models. Snowballing was an effective method to extend the range of work covered and it allowed for a more well-rounded literature review to be conducted. A balanced number of studies carried out were reviewed in an objective manner, discussing both their benefits and shortcomings. Since the inception of deep learning, there have been many studies carried out that use machine learning algorithms in order to predict stock prices. A wide array of techniques has been used such as artificial neural networks, Support Vector Machines, and Genetic Algorithms in order to form predictions using historical stock price data. In the work of [2], authors have carried out a support vector machine implementation that predicts stock prices using input features such as price volatility, price momentum, sector volatility, and sector momentum, and a radial basis kernel was used to classify the data points in the hyperplane. Support vector machines are a supervised machine learning model that works by categorizing data in a high dimensional space despite the data not being linear. Once the categorization phase is complete, a hyperplane or separator is used to differentiate the data. Even though the

work shows promising results, there is very little indication as to the accuracy gained by the model with respect to the testing of the model – this brings up the question of whether or not the model was overfitted. It can also be argued that the input feature such as the volume of stocks traded on a daily basis could have been incorporated as this correlates to the movement of the stock price.

Authors in [3] used a support vector machine for the Indian Stock Market. A sample size of 990 of each stock within a specific time period was used – this time period is consistent with all the stocks. The predictive power of the model was used with statistical metrics such as NMSE (normalized mean squared error), MAE (mean absolute error), and DS (directional symmetry). Even though the model showed a reasonable predictive ability, it should be noted that most of the metrics used are more commonly used in regression analysis and that the noise of the stock market data can produce misleading results. The support vector machine approach tends to provide more generalization using the structural risk minimization principle [4].

Support vector machines were the favored approach to handle time-series data prior to the introduction of recurrent neural networks and long-short-term memory layers. The more recent work in the field has shown a shift in which recurrent neural networks are more dominant. One key disadvantage of support vector machines is that it has difficulty categorizing more noisy data and distinguishing such points in the hyperplane can be difficult leading to lower accuracy.

Authors in [5], carried out a specialized genetic algorithm on the Dow Jones index. A DSGA (Dynamic-radius Species conserving Genetic Algorithm) was used on the stocks. The training data consisted of the first quarter of 2011 and the test data was used against the second quarter to evaluate the model. The fact that only two quarters of data was used can lead to inconsistent results as several companies can perform better during the latter part of the year and this omitted data could have produced more accurate results.

In the work of [6], authors have created an artificial neural network that employs genetic algorithms to update the weights and biases and observed an increase in accuracy and a decrease in training the network. A fellow hybrid model that uses genetic algorithms in neural networks was used in feature discretization and connection weights rather than using genetic algorithms to optimize the model [7]. Genetic algorithms in general are computationally expensive and are not traditionally used to work with highly volatile data. It can be argued that certain classification rules used within genetic algorithms can omit useful dependencies thereby introducing certain inaccuracies in the predictions made.

There has been an increasing number of studies that hybridize existing algorithms and aim to produce a more accurate and efficient result. Authors in [8] used an artificial neural network supported with a particle swarm optimization in order to overcome the limitation of neural networks converging at a local minimum the particle swarm optimization led to greater accuracy as it assigned more appropriate weights and biases. Another hybrid approach was implemented using multilayer perceptron, genetic algorithms, and particle swarm optimization this approach predicts the direction of the stock indices rather than predicting the price.

Multilayer perceptron is a variant of artificial neural networks and is composed of an input layer, hidden layer and output layer. The hidden layer updates the weights and biases with the use of a backpropagation algorithm. Authors in [9] conducted a study in which multiple deep learning models were used and one architecture chosen for this was a multilayer perceptron neural network – it was determined that the convolutional neural network outperformed the multilayer perceptron. An identical study was done where several deep learning models were compared against each author in [10]. The disadvantage of multilayer perceptron's is the fact that it takes a lot of time to understand the relationship between the input features and targets in addition to relying on the training data more than other deep learning models.

Artificial neural networks are the most prevalent in stock market prediction. It has many different architectures and approaches that can be used to tackle complex problem domains and noisy data. Authors in [11] attempted to use an artificial network for stock prediction. In this study a multivariate approach was used with stocks in the Bombay Stock Exchange and a flexible neural network was produced. They claimed that increasing the input values would increase the accuracy however no consideration was made to overfitting the model. An LSTM (Long Short-Term Memory) model is a variant of neural networks and allows the ability to study sequences of data and their dependencies. This type of behavior is highly beneficial when studying the stock market which is sequential. Authors in [12] explored the suitability of an LSTM model with stock market data. Their work concluded that the LSTM model performed better when compared to specific baselines such as a multi-layer perceptron model, a random forest model, and a pseudo-random model. It can, however, be argued that more baselines should have been used to properly gauge the efficacy of the proposed model. Authors in [13] used a recurrent neural network with an LSTM using the root mean squared error to evaluate the stock price predictions. They showcased that increasing the number of epochs led to greater accuracy, however, they did not justify whether using the root mean squared error was the most optimal solution. A similar variant that is commonly used in forecasting is ARIMA (autoregressive integrated moving average). ARIMA works by finding a correlation between past historical data points and then the moving average of that data is incorporated together to produce the forecast.

Authors in [14] used an ARIMA model to perform forecasting of stock prices. Their work concluded by stating that satisfactory predictions were made but the predictions themselves were done in the short- term. Authors in [15] carried out a comparison between ARIMA and LSTM models to study the effectiveness of both and concluded that the LSTM and artificial neural networks are superior to ARIMA. One major drawback of ARIMA is the fact that it is computationally expensive, and it works by identifying patterns in the data. Since stock market data follows a random walk, it can be argued that an ARIMA model is not suitable for stock price predictions.

A variant of support vector machines using regression also known as, support vector regression, was carried out by [16] using data from the Hang Seng Index. The work primarily focuses on using different statistical margins to reduce the loss in the predictions. The historical data used in the experiment is quite low, only 104 days back worth of stock data was utilized and might not show the same accuracy should the input size be increased.

To conclude, numerous machine learning implementations have been performed on stock market indices. Most of the outcomes point towards positive results but it is difficult to

gauge which model is superior as they all have their own strengths and weaknesses. The vast majority of the studies show each model was successful in accurate predictions, however, very little information was provided as to the loss and validation loss produced in the training and testing processes. Such metrics will be extremely valuable when drawing more direct comparisons and can help in identifying which model performs better.

3. METHODOLOGY

The main outcome that is to be achieved from the study was to successfully make future stock market share predictions. In order to achieve this more information was gathered about past machine learning models being used to predict stock prices.

Initially, as a primary requirement gathering, a questionnaire was done with the use of a Google survey with a target number of respondents as 100. The gathered answers were varied, however, the majority of the respondents seemed to react positively towards the questions asked. In order to maintain relevance with the target audience, the targeted groups were individuals who had shown an interest in the stock market and were considered in the survey. The form was posted on Reddit and sent to individuals over specific distribution lists. Going over the responses, it was observed that for each question, over 60% of the respondents reacted in a positive manner.

Then further information was gathered using existing research carried out in the field and the type of machine learning models produced. The secondary requirements gathering was going over existing technical documentation and recording their methodology, tools, and processes used. Similar projects were assessed and the limitations in the research were identified. The focus of the existing research was to find out the technologies and tools used to implement the model. The vast majority of the machine learning implementations were done using Python but, in some cases, the programming language known as "R" was used. Python was chosen for this project due to the fact it has an easier syntax and its versatility in having many machine learning packages and data science related libraries. After analyzing the existing literature in the problem domain, APIs (application programming interfaces) were explored to see how the relevant data could be acquired and, in this case, it was historical stock market data.

Another secondary requirement gathering technique that was employed in the study and it was prototyping. In the prototyping phase, a baseline model was produced, and the efficiency of the model was observed without tuning it. This gave an idea about what type of hyperparameters needed to be adjusted in the actual development of the model as well as incorporating more complex loss functions and optimizers.

Then after the requirement gathering, functional and non-functional requirements for the proposed system were decided. Then the study was focused on the designing phase. The underlying design considerations are all based on a prototyping methodology. Prototyping was chosen over other longer, more traditional approaches due to its agile and lightweight approach. Choosing prototyping can be further justified as it does not have any implications on the schedule of the project. In the study, there were two main phases to be designed and implemented as below.

- Creation of predictive model
- Sentiment analysis from Tweets

In the first phase, initially, the data were gathered with the API, and then pre-processing took place after which all the input values were standardized. Finally, the model was created, fed the data and compiled. Then, the resulting metrics such as the loss were recorded and compared against subsequent iterations to improve the model. After the model was successfully trained and tested, it was hosted on a web server so the end-users could interact with it.

The proposed prediction model was implemented at this phase and initially, historical data was gathered and pre-processed. The data went through multiple transformations in order to be inserted into the neural network. Stock market closing prices were the targets that were chosen as the inputs for the model. Hence test and train data was split and train data was used with the RNN model and LSTM algorithm to train the model. An 80-20 split was done for the training and testing set respectively. Then the MinMaxScaler was used to scale the values between 0 and 1. The decision to choose this scaler implementation was because the stock data can have many different distributions. Then a sequential model was instantiated, and the layers were added. The model has three LSTM layers and one dense layer. The dropout rate is assigned to 0.2 to prune the network and prevent overfitting. A mean squared error loss function was chosen along with the 'adam' optimizer. The mean squared error was used as greater errors are balanced more efficiently by the weights. The model was then fitted with the training data and the targets and labels were evaluated. Finally, the test data was used to do the predictions and hence compared the results with the graphical representation of actual predictions made. Given below are the diagrams illustrating the proposed model (Fig 1). The architecture for the neural network consists of three stacked LSTM layers. The input layer feeds the input values to all the neurons in each layer. The return sequences are set to true for the first two layers as the hidden state of the input values is recorded in order to identify a pattern in the time series. The final LSTM layer then outputs the values to the dense layer. A dropout rate of 0.2 is assigned in order to prune the neural network to account for the overfitting of the data. Then in the second phase, tweets were gathered using tweepy API and then the data were preprocessed and analyzed with the NLTK Vader sentiment analyzer. Finally, the resulting polarity values were displayed using graphs.

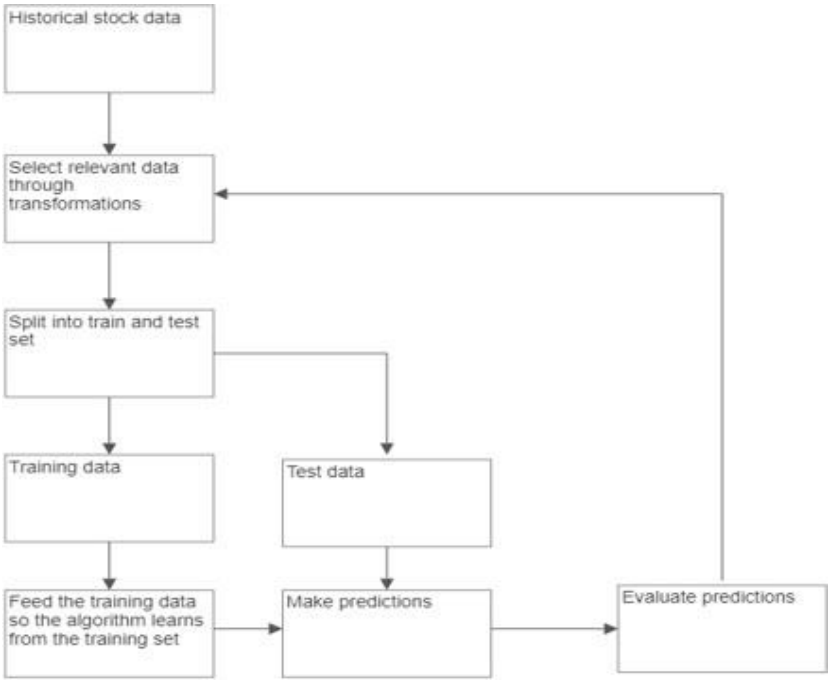

Fig 1: Proposed prediction model

4. ACADEMIC FINDINGS AND RESULTS

With the study the fact that an LSTM-based recurrent neural network is capable of predicting stock prices was identified and the prediction gave a result with a high degree of accuracy as shown in the figure given below (Fig 2). From Fig 2, it is clearly observed that the model has identified the pattern emerging from the input data and it was able to accurately predict the labels in the training phase.

The loss function that produced the lowest possible value for the predictions was the mean squared error. It was paired with the 'adam' optimizer and showed a gradual descent in producing the loss values and there was no evidence to suggest that the model was overfitted. The loss produced from the model is given below in Fig 2. In Fig 3, the graph clearly shows a steep decline followed by a gradual decrease showing that the loss was incrementally decreasing. The graph does not seem to fluctuate with respect to the loss values therefore we can conclude that the model was not overfitted. Supported by such a model, financial institutions can utilize it to complement their existing tools and processes to manage risks. Further, the sentiment analysis showed that there is little to no correlation of social media events affecting stock prices. However, in the past, certain events like the GameStop stock increase show that social media can contribute to black swan events and even trigger unpredictable phenomena in the financial world. In order to verify the model, stock market prices for well-known companies were predicted and further, twitter data sentiment analysis to evaluate the impact of social media on respective company's stock market prices for the considered time period was conducted. Results obtained from two cases have been illustrated in the below figures, Fig 4-11.

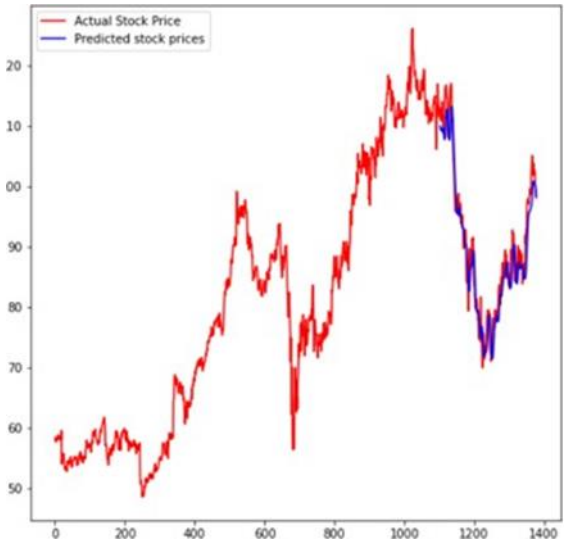

Fig 2: Stock market prediction from proposed model compared with actual historical data

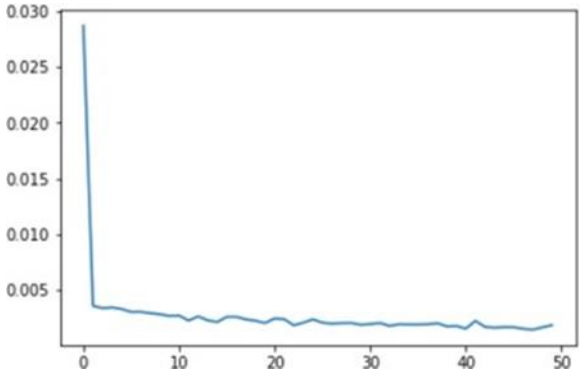

Fig. 3. Loss produced from the model

Despite promising results being shown by the neural network it is quite evident that stock prices are still too erratic and follow

a random walk. The neural network will always provide an approximation with regard to the prediction. It is precisely due to this fact that such a model should be used to anticipate either the negative or positive movement and not used to identify the exact increase or decrease in stock prices.

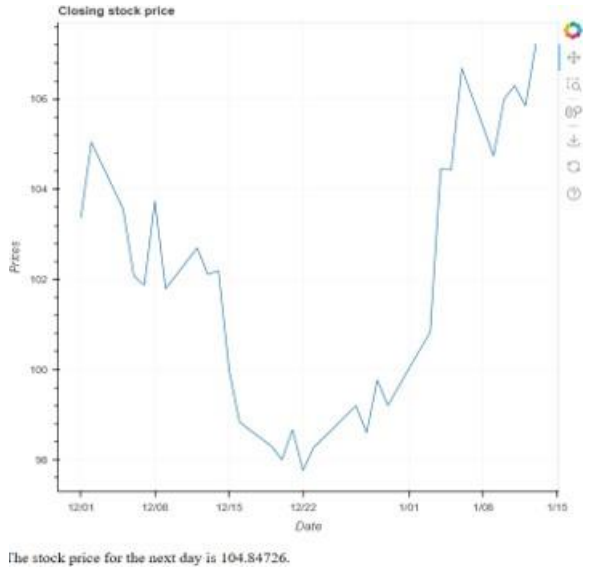

Fig. 4. Stock Market Prediction for Starbucks from the Model

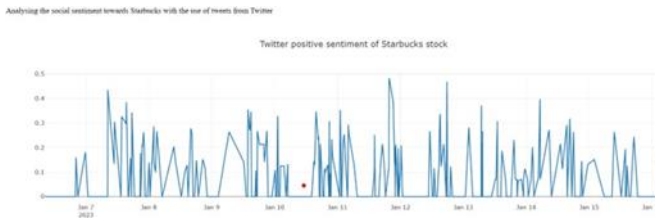

Fig. 5. Twitter Positive Sentiment of Starbucks Stock

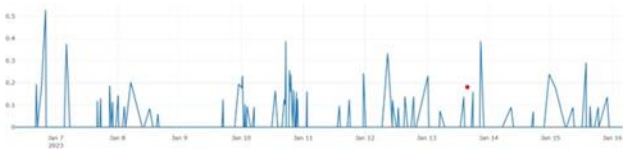

Fig. 6. Twitter Negative Sentiment of Starbucks Stock

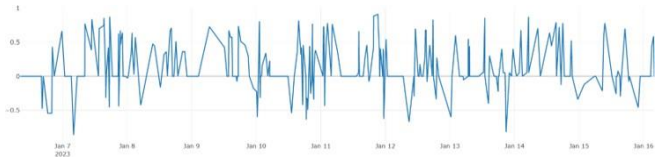

Fig. 7. Positive, negative, and Compound Sentiment Analysis for Starbucks from Twitter data

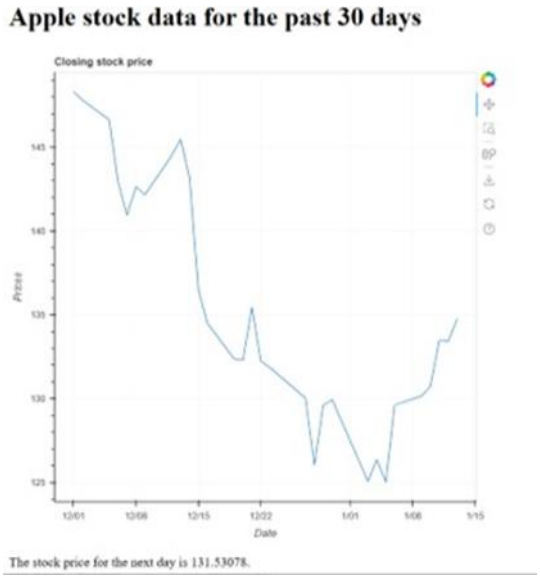

Fig. 8. Stock Market Prediction for Apple from the Model

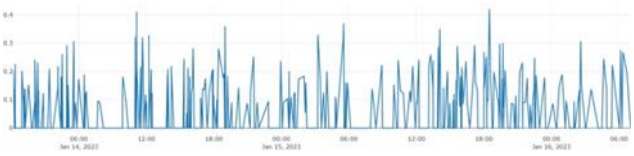

Fig. 9. Twitter Positive Sentiment of Apple Stock

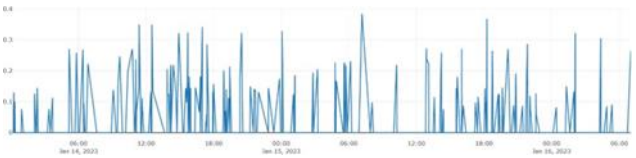

Fig. 10. Twitter Negative Sentiment of Apple Stock

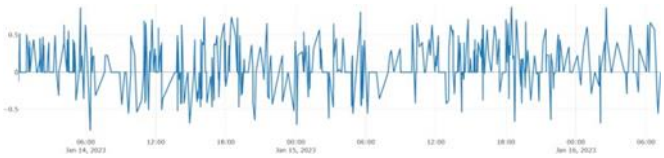

Fig. 11. Positive, Negative, and Compound Sentiment Analysis for Apple from Twitter Data

5. CONCLUSION

Neural networks are extremely powerful tools that are capable of solving complex problems. Machine learning algorithms have developed to such an extent that they can get good approximations with regard to the predictions made. Despite the model making good predictions, it should not be used in isolation to determine when to buy or sell stocks as the predictions themselves are just approximations and the model will never be able to predict what the actual prices may be. Instead, the model has great value as an indicator of whether a stock will have a positive or negative movement.

The sentiment analysis performed also showed great insight into how social media interacts with the stock market and the flow of information that is being passed around. Past events like the GameStop stock surge, were triggered solely by the use of social media a lone event like this unfortunately does little to suggest that the stock market prices can be manipulated through

social media.

The deployed model had difficulty learning long-term dependencies between the stock prices and increasing more time steps into the future. It ended up being overfitted with data and performed poorly on the validation data provided. A more reserved approach needs to be taken when increasing the training data and the model needs to have more effective pruning techniques to tackle overfitting. More exploration would need to be done and even custom loss functions or optimizers may have to be introduced when the scope increases. This type of implementation would have been difficult to implement given the time constraints and implications it would have had on the schedule. The trained model seemed to be more receptive to the mean squared error and adam optimizer and proved to be superior when compared to the others. This brings up the question of how time series data can be tackled in a more effective manner by creating guidelines to create a baseline model with these parameters. The fact that there are a multitude of implementations using different architectures means the research community has yet to decide on which model seems the most suitable. Efforts should be taken to identify the most superior model and attention needs to be paid to refining it. The future of LSTM-based neural networks looks promising and will hopefully be utilized to its full capabilities.

6. FUTURE WORKS

Enhancements to the existing system would be to increase the number of days the model predicts, currently, it predicts a one-time step into the future. The other refinement would be to do a more extensive sentiment analysis, in this case, NLTK vader's built-in sentiment analyzer was used. The Twitter API, tweepy, which was used offers a limit on the number of requests that can be made – this significantly falters the amount of data that can be gathered. The possibility of analyzing social media data from other mediums such as Facebook, Reddit, and other trending platforms will also need to be explored. In order to overcome difficulties such as reaching limit errors, it would be beneficial to incrementally add the data from the APIs to a database so that they can be queried later. This would also significantly increase the dataset that can be used for the model as well as the added advantage of performing some pre-processing from the database itself such as executing complex queries.

7. REFERENCES

[1] P. Jiao, A. Veiga, and A. Walther, Social media, news media and the stock market. Journal of Economic Behavior and Organization, 176, pp.63-90, 2020.

[2] S. Madge, and S. Bhatt, Predicting stock price direction using support vector machines. Independent work report spring, 45, 2015.

[3] S.P. Das and S. Padhy, Support vector machines for prediction of futures prices in Indian stock market. International Journal of Computer Applications, 41(3), 2012.

[4] K.J. Kim, Financial time series forecasting using support vector machines. Neurocomputing, 55(1-2), pp.307-319, 2003.

[5] M.S. Brown, M. J. Pelosi, and H. Dirska, Dynamic-radius species-conserving genetic algorithm for the financial forecasting of Dow Jones index stocks. In International Workshop on Machine Learn- ing and Data Mining in Pattern Recognition (pp. 27-41). Springer, Berlin, Heidelberg, 2013.

[6] M. Qiu and Y. Song, Predicting the direction of stock market index movement using an optimized artificial neural network model. PloS one, 11(5), 2016.

[7] K.J.Kim, and I. Han, Genetic algorithms approach to feature discretization in artificial neural networks for the prediction of stock price index. Expert systems with Applications, 19(2), pp.125-132, 2000.

[8] F. Ghashami, K. Kamyar, and S. A. Riazi, Prediction of stock market index using a hybrid technique of artificial neural networks and particle swarm optimization. Applied Economics and Finance, 8(1), 2021.

[9] M. Hiransha, E.A. Gopalakrishnan, V. K. Menon, and K. P. Soman, NSE stock market prediction using deep-learning models. Pro- cedia computer science, 132, pp.1351-1362, 2018.

[10] M. Usmani, S.H. Adil, K. Raza, and S.S.A. Ali, Stock market prediction using machine learning techniques. In 2016 3rd international conference on computer and information sciences (ICCOINS) (pp. 322-327). IEEE, 2016.

[11] A.V. Devadoss, and T.A.A. Ligori, Stock prediction using artifi- cial neural networks. International Journal of Data Mining Techniques and Applications, 2(1), pp.283-291, 2013.

[12] D.M. Nelson, A.C. Pereira, and R.A.De Oliveira, Stock market's price movement prediction with LSTM neural networks. In 2017 International joint conference on neural networks (IJCNN) (pp. 1419-1426). IEEE, 2017.

[13] M. Roondiwala, H.Patel, and S.Varma, Predicting stock prices using LSTM. International Journal of Science and Research (IJSR), 6(4), pp.1754-1756, 2017.

[14] A.A. Ariyo, A.O. Adewumi, and C.K. Ayo, Stock price prediction using the ARIMA model, UKSim-AMSS 16th international conference on computer modelling and simulation (pp. 106- 112). IEEE, 2014.

[15] Q. Ma, Comparison of ARIMA, ANN and LSTM for stock price prediction. In E3S Web of Conferences (Vol. 218, p. 01026). EDP Sciences, 2020.

[16] H. Yang, L. Chan, and I. King, Support vector ma- chine regression for volatile stock market prediction. In International conference on intelligent data engineering and automated learning (pp. 391-396). Springer, Berlin, Heidelberg, 2002.