Stock Market Price Forecasting using Parallel Recurrent Neural Network

by

Md Nahid Hasan Shuvo
Department of Computer Science
College of Information Science and Technology
University of Nebraska at Omaha
Omaha, NE 68182-0500
nshuvo@unomaha.edu

For

CSCI 8590 Fundamental of Deep Learning
Spring 2022

Date: Saturday, May 4, 2022

Introduction

The stock market has grown in popularity in recent years as a result of its high returns. Despite the considerable risk, several institutions and innovators continue to invest in the stock market. The study of stock market prediction has been ongoing for some years, but due to its complexity, the number of variables and sources considered are limited. It has been established that this is an extraordinarily difficult task. Early awareness of ever-changing stock values gives the investor a decision-making advantage. There are around 60 major stock exchanges worldwide working currently. The stock exchange's global market value has increased steadily throughout the years, from $2.5 trillion in 1980 to $68.65 trillion at the end of 2018. Even during the pandemic's crisis era, the stock market remained functional and actively supported the economy, and by the end of 2020, the worldwide market capitalization had surpassed $100 trillion. Stock markets operate because of investor confidence and a consistent profit return on investment. Stockholders must be aware of ever-changing stock price movements in order to do this. This can be accomplished by building a solid framework for stock price prediction.

It's worth remembering that in 2012, it was anticipated that algorithms could handle approximately 85 percent of stock market transactions in the United States. Numerous methods for forecasting stock prices have been developed, including classical models and machine learning-based approaches. Statistical models such as AR, ARIMA, and GARCH are able to compute linear correlations, however time series data such as stock forecasting are non-linear. And to address this, the neural network model is extremely effective [1]. Indeed, forecasting the financial markets is a difficult undertaking due to the sequence's extremely nonlinear, dynamic, and noisy nature.

RNNs have been demonstrated to be one of the most effective models for processing sequence information [2], capable of recognizing sophisticated non-linear correlations.
However, pure RNN channels have some difficulty with long term memory; in this circumstance, RNN subtypes such as LSTM (Long Short Term Memory) and GRU (Gated Recurrent Unit) are now the most effective. LSTM and GRU are identical in operation, but GRU incorporates a more efficient and faster memory system by substituting for a few gates in the LSTM network. Although LSTM appears to be more efficient than GRU, there is still controversy over which algorithm to utilize in which situation. Another type of RNN is BiLSTM or BiGRU. These are similar to raw LSTM and GRU, but have two identical based

models oriented in opposite directions, effectively increasing the amount of information available on the network. As a result, BiLSTM has demonstrated superior performance over the LSTM network [3].

However, to the best of our knowledge, none of the papers so far has shown the application of parallel recurrent Neural network for stock market prediction. To achieve the forecasting efficiently and quickly, we have chosen the LSTM and GRU based network in parallel to forecast the time series on this project. As we know, there are several cases, like weather forecasting, Cloud workload prediction, even for the cyber security the DDoS attack prediction, this type of forecasting technology might be very useful.

Related Works

Accurate stock market prediction is a long-desired field in statical analysis pedagogy. Modern deep learning algorithms have started to excel at many autoregressive tasks. These algorithms have become so expressive that they started to recognize patterns on highly complex regression task as the stock market analysis. As a result, the deep learning research has transpired a boom in deep learning-based time series forecasting research. In this section, we will briefly review some of these related works.

Some of the early work on stock market predictions were done by MLP's which due to high number of parameters were often prone to overfitting issues. Simple methods like linear regression and KNN have shown some prediction efficacy in limited regression scenarios. Decision trees [4] have demonstrated better accuracy when employed with advanced feature engineering to techniques. On the contrary, SVM's, while being simple, have proven very versatile and significant on accurately forecasting time series data. But due to their limited expressivity, SVM's [5] are not capable of correlating highly unstructured data introduced by stock markets. Additionally, such data requires an element of memory component to factor the previous important signals to make an informed prediction on current situation. While many RNN's have excelled with sequential data, most of them don't embed any memory. Introduction of LSTM [6] networks started to mitigate the issue of long-term memory dependencies. Shao et al. [7] introduced a framework that can forecast available parking spaces in multi-steps ahead using LSTM model. Yuming et al. [8] proposed the GRU-ES combining gated neural recurrent unit model with exponential smoothing method to predict cloud resource usage. The proposed method is able to outperform SOTA models in both single and multi-step predictions. LSTM networks have formulated to adopt this general framework for learning stock market forecasting task which is currently employed by many well know research articles. Sadman et al. [9] introduced LSTM based RNN with extensive hyperparameter tuning to maximize accuracy for predicting DSE data. Kai et al. [10] showed the improvement in accuracy of LSTM model compared to other regression models through their research. Sliding window based CNN [11] networks have also proven to be a good estimator for accurate time series data prediction. This technique is widely used to build efficient sentiment analysis and classification applications. Inspired by similar technique, Sreelekshmy et al. [12] applied LSTM, and CNN-sliding window methods for predicting NSE listed companies stock price. Arif et al. [13] adopted Bi-directional LSTM to increase the accuracy of the forecasting where objective is to minimize the RMSE value.

Methodology

A. Feature Extraction by 1DCNN:

Convolution layers have a very high ability to discover features, hence they were employed to learn local patterns from the input information, even though Dense Networks learn distinct patterns from the input feature space. Time series data can be condensed down to a much more manageable length with ConvNet. Figure 1 depicts a simplified representation of a 1-dimensional CNN in action. A kernel with a predetermined window size is used to filter out unwanted sequences at the outset. A 1D convolution is used to separate the input sequences into their individual parts. The retrieved sequences are used to weight the layer in dot production, and the output sequences are then obtained. This output sequences

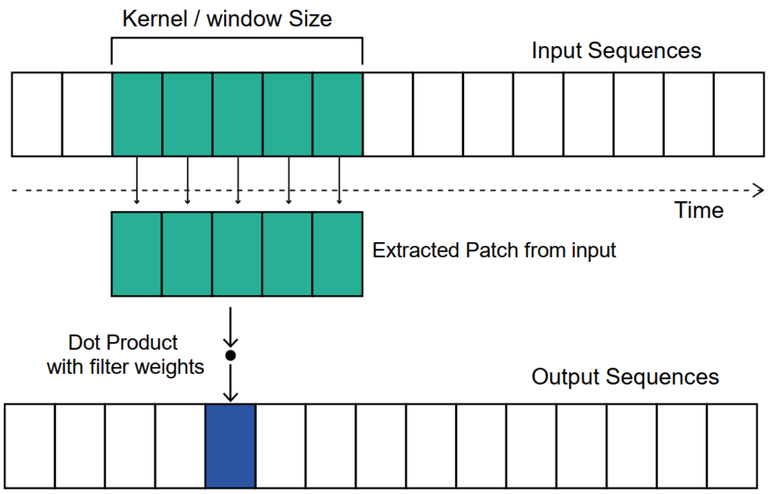


Figure 1: Working procedure of 1D CNN

contain high-level features that can be used as the input to any neural network system's dense layer. The architecture of the LSTM cell illustrated in figure 1.

B. Long Short-Term Memory (LSTM):

Long-short term memory is a type of RNN that is capable of learning to solve problems involving long-term dependencies. According to one study, LSTM is capable of properly dealing with time series data that exhibits long-term dependency. An input gate, a forget gate, and an output gate are all included in LSTM neurons. LSTM controls data flow through these three functional gates, which also serve as storage for historical data. The figure 2 illustrates the architecture of the LSTM cell. The memory cell is the core component of the LSTM. It contains crucial information that has been accumulated over time, and the new information is transformed at each step by the three gates. As a result, LSTM is superior to RNN in learning long-term dependencies. From those three gates, the keep gate specifies how much information should be retained from the previous cell, and the write gate determines how much information should be sent to the memory state, so resolving the vanishing gradient problem.

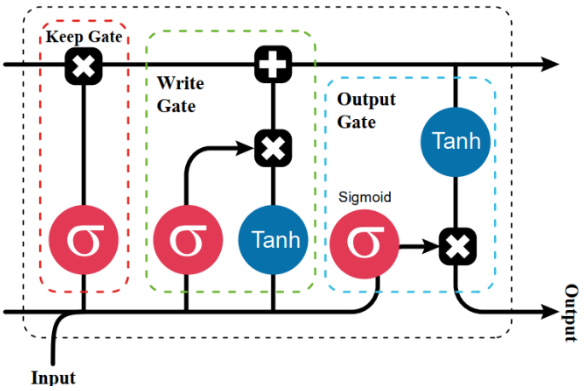


Figure 2: LSTM cell

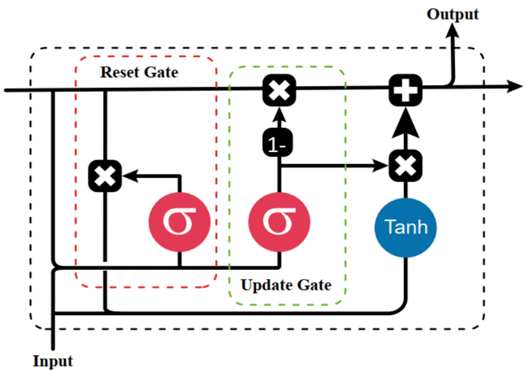


Figure 3: GRU cell

The memory cell is the LSTM's central component. It contains crucial information that it has accumulated over time, and the new information is transformed at each step by the three gates. As a result, LSTM is much more adept at learning long-term dependencies than RNN. From those three gates, the keep gate specifies how much information to retain from the previous cell, and the write gate decides whether to write information to the memory state, so resolving the vanishing gradient problem.

C. Gated Recurrent Unit (GRU):

In figure 2 GRU cell architecture illustrated, which is a variant of LSTM introduced in 2014. Since LSTM is relatively complex and thus it requires more training time, therefore, GRU was introduced. GRU method optimizes the input and forget gate by the update gate. Her the update gate acts as like as input gate in lSTM for making the decision of keeping necessary information. Along with update gate GRU unit has another gate for deciding how much past data need to throw away. Therefore, the cell structure changes into two gates instead of three gates. Though theoretically LSTM contains longer sequences in memory than GRU, but GRU have higher computational efficiency with good accuracy due to less component per cell.

D. System Architecture:

Here in this work, I proposed an optimized RNN based model, Her the data are inputted to the 1D convolution neural network, then the output of the 1D CNN is splits and passed to the two different networks, one goes to the LSTM network and another goes to the GRU network. LSTM network, retain the information for longer periods of time and the GRU

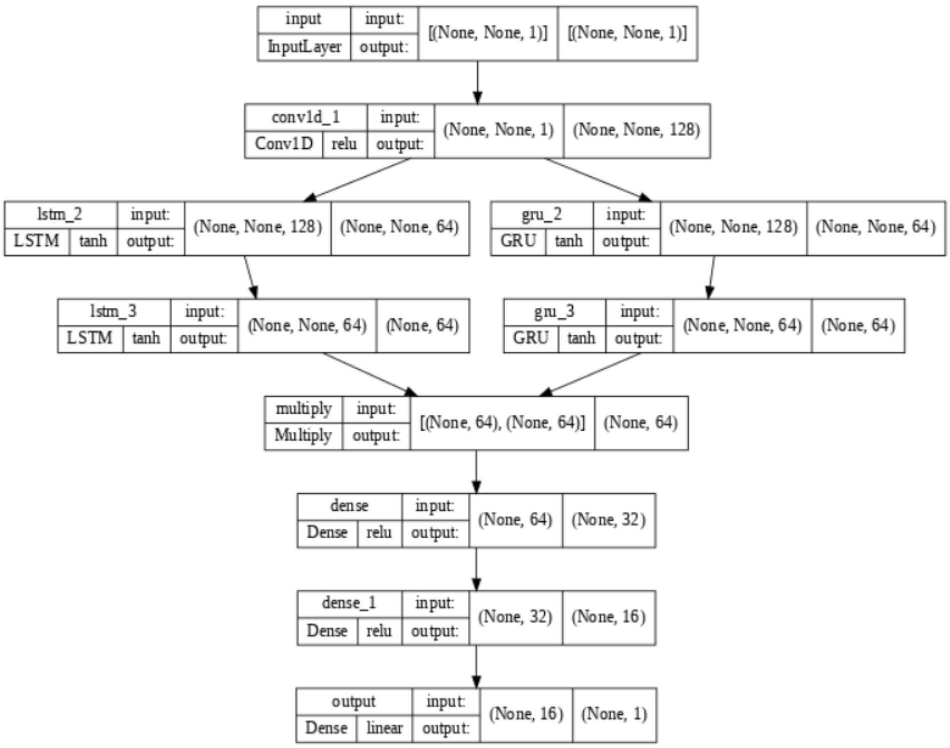


Figure 4: Our proposed parallel RNN based model

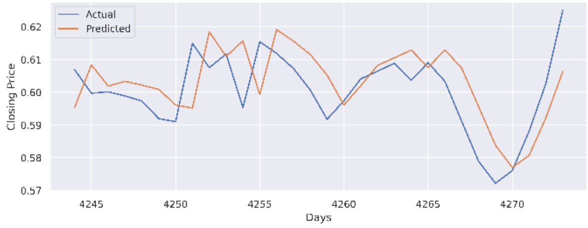
have easy computation efficiency thus prediction from each module passed into a single dense layer, which then predict the final output. In the optimized model the number of neurons for each RNN was 64, We used 'tanh' activation on each RNN layers and for dense layers we used 'ReLU' activation function. Figure 4 shows the architecture.

Experiment

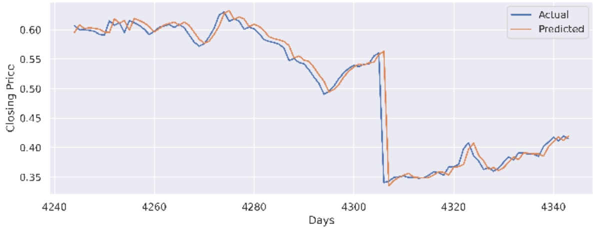
A. Dataset:

We used a genuine dataset from the Kaggle dataset repository for this study, which contained stock price data.The data originates from India's National Stock Exchange (NSE), and it contains the price history and trading volume of the NIFTY 50 index's fifty constituent stocks [14]. All datasets are collected daily as beginning in January 2000 and ending in April 2021. We selected stock price data for BPCL (Bharat Petroleum Corp Ltd.) from this dataset. Stock's closing price for training and validating our model. The dataset was splitted into 8:2 manner. Figure 5 shows the close price of BPCL stock before and after normalization of the data.

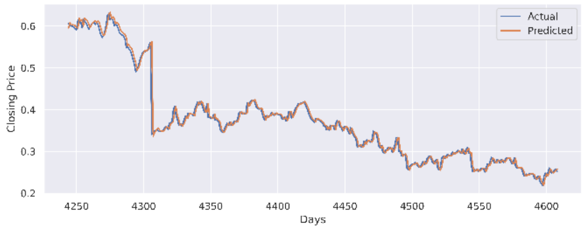
Figure 5: BPCL stock's Closing price



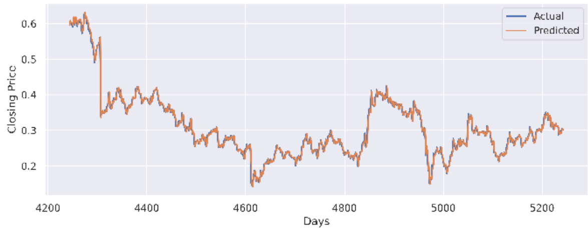
(a) 30 days ahead prediction



(b) 100 days ahead prediction



(c) 365 days ahead prediction



(d) 1000 days ahead prediction

Figure 6: Actual vs predicted Closing stock price by our approach

B. Result and Analysis:

For conducting this experiment, we used google Colab platform, with python Keras and TensorFlow library. We have seen that BiLSTM works better that LSTM thus we take BiLSTM as base model and compare that with our approach. In figure 6, we can see that our model can predict 30,100,365, and 1000 days ahead prediction. If we can see the result closely, we can see that for a single day the error of our model is very low.

In fact, figure 7 shows us the training vs validation loss, and from there we can see that

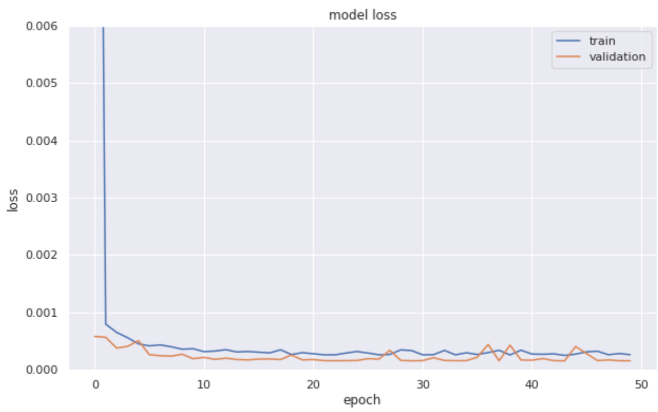


Figure 7: Training vs Validation Loss

Table 1Error Analysis among Bi-LSTM and Our Approach

	RMSE		**MSLE**		**R2 Score**	
	Train	Test	Train	Test	Train	Test
Our Approach	0.01515	**0.01233**	0.00011	**0.0000846**	0.9940	**0.9821**
Bi-LSTM	0.01627	0.1323	0.00013	0.000099	0.9930	0.09794

after a very few epochs the loss have been minimized and almost saturated. However, more smoothing can be possible here by more hyperparameter training, which we will do in our future work.

We also implement Bi-LSTM model and compare the result with our approach. Table 1 shows us the comparison between them, using Root Mean Square Error (RMSE), Mean Squared Logarithm Error (MSLE), and coefficient of determination error (R^2). From the table we can see that for RMSE, and MSLE error, our method get smaller value and for R^2 score it gets higher value, which shows that, our model perform better that the Bi-LSTM model.

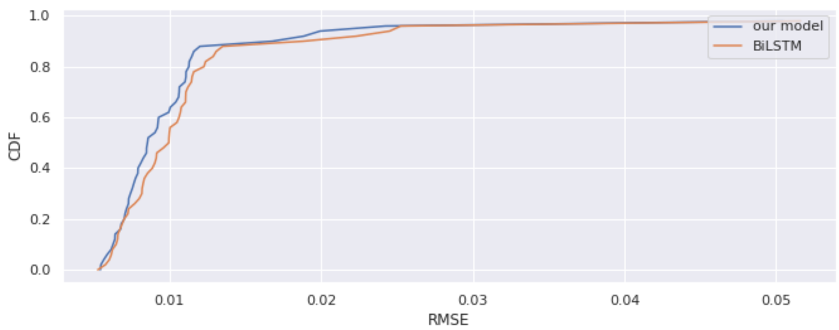


Figure 8: CDF analysis of our model vs Bi-LSTM

Moreover, we also analyzed the Cumulative Distribution Function (CDF) for RMSE prediction error to find out the stability and harmony of our approach with compared to the Bi-LSTM approach. In figure 8, we can see that, our approach has curve closer to the y-axis than the Bi-LSTM model, which denote that our model have superior performance.

Conclusion

Investors are increasingly looking for accurate and dependable stock market forecasting tools. In this work, we see that RNN based model can extract the long -term dependencies present on time series data. We propose a parallel RNN based architecture, which utilize the power of both LSTM and GRU and combined them together to forecast the final output. In fact, it shows better performance than state-of-the-art Bi-LSTM model by means of different error (RMSE, MSLE, and R^2) analysis. Also, from CDF analysis, we can see that, our model has better stability. Also, both models have almost equal number of trainable parameters, so the runtime of each model is almost equal. Thus, our model can be utilized for the stock market price prediction by any individuals or organizations.

In future, we will analyze more stock market data and cryptocurrency price data with this mode. And we will also go deeper into this model to train it in a away, so that we will be able utilize it for other time series applications like, datacenter load forecasting, or Weather forecasting.

Runnable Code Link:

https://colab.research.google.com/drive/1t9J9wCY5au8JYvF9XhtF97a3SL8yMA1O?usp=sharing

Reference

[1] Ferreira, Tiago AE, Germano C. Vasconcelos, and Paulo JL Adeodato. "A new intelligent system methodology for time series forecasting with artificial neural networks." *Neural Processing Letters* 28.2 (2008): 113-129.

[2] Pham, Dinh Tuan. "Stochastic methods for sequential data assimilation in strongly nonlinear systems." *Monthly weather review* 129.5 (2001): 1194-1207.

[3] Siami-Namini, Sima, Neda Tavakoli, and Akbar Siami Namin. "The performance of LSTM and BiLSTM in forecasting time series." *2019 IEEE International Conference on Big Data (Big Data)*. IEEE, 2019.

[4] Nair, Binoy B., V. P. Mohandas, and N. R. Sakthivel. "A decision tree-rough set hybrid system for stock market trend prediction." *International Journal of Computer Applications* 6.9 (2010): 1-6.

[5] Lin, Yuling, Haixiang Guo, and Jinglu Hu. "An SVM-based approach for stock market trend prediction." *The 2013 international joint conference on neural networks (IJCNN)*. IEEE, 2013.

[6] Yu, Yong, et al. "A review of recurrent neural networks: LSTM cells and network architectures." *Neural computation* 31.7 (2019): 1235-1270.

[7] Shao, Wei, et al. "Parking availability prediction with long short term memory model." *International Conference on Green, Pervasive, and Cloud Computing*. Springer, Cham, 2018.

[8] Cheng, Yuming, et al. "Gru-es: Resource usage prediction of cloud workloads using a novel hybrid method." *2019 IEEE 21st International Conference on High Performance Computing and Communications; IEEE 17th International Conference on Smart City; IEEE 5th International Conference on Data Science and Systems (HPCC/SmartCity/DSS)*. IEEE, 2019.

[9] Islam, Sadman Bin, Mohammad Mahabubul Hasan, and Mohammad Monirujjaman Khan. "Prediction of Stock Market Using Recurrent Neural Network." *2021 IEEE 12th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON)*. IEEE, 2021.

[10] Chen, Kai, Yi Zhou, and Fangyan Dai. "A LSTM-based method for stock returns prediction: A case study of China stock market." *2015 IEEE international conference on big data (big data)*. IEEE, 2015.

[11] Sadouk, Lamyaa. "CNN approaches for time series classification." *Time Series Analysis-Data, Methods, and Applications* (2018): 1-23.

[12] Selvin, Sreelekshmy, et al. "Stock price prediction using LSTM, RNN and CNN-sliding window model." *2017 international conference on advances in computing, communications and informatics (icacci)*. IEEE, 2017.

[13] Sunny, Md Arif Istiake, Mirza Mohd Shahriar Maswood, and Abdullah G. Alharbi. "Deep learning-based stock price prediction using LSTM and bi-directional LSTM model." *2020 2nd Novel Intelligent and Leading Emerging Sciences Conference (NILES)*. IEEE, 2020.

[14] Vopani, "Nifty-50 stock market data (2000 - 2021)," May 2021, [Online]. Available: *https://www.kaggle.com/rohanrao/nifty50-stock-market-data*