HIGH SCHOOL EDITION
Journal of Student Research
Volume 12 Issue 4 (2023)

Comparative Analysis of LSTM, GRU, and ARIMA Models for Stock Market Price Prediction

Rushil Yavasani[1] and Frank Wang[#]

[1]Morris Hills High School, USA
[#]Advisor

ABSTRACT

This study delves into the efficacy of various machine learning and statistical models that have captured the attention of financial analysts. Two of them, Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU) are variations of Recurrent Neural Networks while the Autoregressive Integrated Moving Average (ARIMA) is a statistical model. These models will be used to forecast stock market data across different economic sectors. In the dynamic landscape of financial markets, accurate forecasting is crucial. This research paper contributes to quantitative finance by conducting a comprehensive comparative analysis of these models on historical stock market data from three sectors: extraction, manufacturing, and service (which are considered the primary, secondary, and tertiary sectors of the economy respectively). The models' performances are evaluated using mean squared error (MSE) on six selected stocks representing these sectors. Results reveal the power of recurrent neural networks in capturing intricate patterns. Moreover, the results will explore whether or not the efficacy of each model is impacted by the sector of the economy that it is forecasting data for.

Introduction

The dynamic nature of financial markets and the intricacies of stock price movements have long captivated researchers and practitioners in the realm of quantitative finance. As the world of investments becomes increasingly competitive and complex, accurate and efficient forecasting of stock market data is crucial for investors and analysts. To this end, predictive models have emerged as valuable tools in the quest to decipher and anticipate financial market trends.

Over the years, quantitative finance research has witnessed remarkable progress, and various innovative methodologies have been developed to better model financial data. Notably, advancements in machine learning techniques, the advent of deep learning algorithms, and the growth of big data analytics have all contributed to more sophisticated forecasting models.

This research paper delves into the efficacy of three prominent models, namely Long Short-Term Memory (LSTM), Gated Recurrent Unit (GRU) and Autoregressive Integrated Moving Average (ARIMA), for forecasting stock market data from companies operating in various industries. These models will be tested on historical stock market data from companies that are representative of each sector in the three sector economics model – agriculture, manufacturing, and service. Each of these models brings a unique set of strengths and limitations, making the investigation of their performances crucial for the advancement of quantitative finance.

In the quantitative finance literature, LSTM has garnered substantial attention due to its ability to capture long-term dependencies and nonlinear patterns in time series data. LSTM has been shown to be proficient in capturing intricate stock price movements across different industries, highlighting its potential for achieving accurate predictions (Sako et al., 2022).

The Gated Recurrent Unit (GRU) is an altered version of LSTM, but has been seen to be more computationally efficient. Its ability to anticipate trends based on long term data has captured the attention of researchers as well, demonstrating the model's potential.

Contrasting the machine learning neural networks like LSTM and GRU, ARIMA serves as a powerful statistical model in time series analysis. ARIMA's simplicity, interpretability, and applicability to various economic and financial datasets have solidified its presence as a staple in forecasting literature (Nau, 2020).

Essentially, this research paper aims to contribute to the existing body of knowledge in quantitative finance by conducting a comprehensive comparative analysis of LSTM, ARIMA, and GRU models for forecasting stock market data across diverse industries. After analyzing the work of previous researchers in a thorough literature review, we aim to outline the methodologies we used for data collection and analysis, and then showcase the results and findings of our tests. Finally, there will be a formulated conclusion that aims to shed light on the strengths and weaknesses of each model and provide insights that can aid investors and financial analysts in making informed decisions in the unpredictable world of finance.

Literature Review

For this project, a thorough literature review was completed to analyze the past research on these statistical models. This would provide context, give guidance as to how to run the experiment, make it clear which models to select, and serve as a baseline that could be used to compare our results to.

The first few sources that were analyzed were specifically looking into the ARIMA model, and its applications on time series data. We started off with a paper that applied an ARIMA model to stocks in Yahoo finance, which found that the model was particularly effective in forecasting data in the short term (Dong et al., 2017). Another source did a more interesting study, conducting a sector-specific analysis with a diverse range of fifty-six Indian stocks (Mondal et al., 2014). As expected, the results indicate that the ARIMA model is impressively accurate, but these results varied based on the stock. Stocks belonging to specific sectors such as FMCG (Fast Moving Consumer Goods) showed the best accuracy, while the banking and automobile sectors had relatively lower accuracy, suggesting the need for better models in those sectors.

Many of these sources stated that future research should be focused on applying other potential models for stock market analysis, especially to account for any flaws or weaknesses the ARIMA model may possess.

After looking at these sources, our focus turned to Recurrent Neural Networks, which were a form of deep learning that has emerged as a powerful approach for analysis in time-series data, eliminating the need for costly hand-crafted features and expert knowledge (Gamboa, 2017). Recurrent Neural Networks come in many variations, each with their own strengths and weaknesses. A few in particular were highly regarded in a variety of research articles, specifically the LSTM (Long Short Term Memory) version and the GRU (Gated Recurrent Unit) version.

In fact, a research project actually created a hybrid model, which synthesized LSTM and GRU cells, finding that with experiments on real financial data, this model was suitable for diverse datasets and capable of efficient signal capture and price movement forecasting (Li & Qian, 2022). These authors were not alone in their general findings, as others found in a comparative analysis of Recurrent Neural Networks, that LSTM and GRU outperformed the simple RNN. Specifically, GRU performed better for stocks with high fluctuations, while LSTM excelled when the price patterns were moderately variable (Dey et al., 2021).

The last notable insight derived from this literature analysis was that a sector specific analysis could be worth pursuing. A paper which compared both ARIMA and LSTM models mentions how both models have varying effects depending on the stock analyzed, hinting that a sector specific analysis may yield results that indicate a significant difference in price prediction between sectors. Supposedly, LSTM performs better at predicting stock prices and expressing price changes, while ARIMA is easier to apply due to its shorter training time and fewer parameters (Xiao et al., 2022).

Methods

In order to perform this experiment, yahoo finance historical stock data was collected in csv format, which displayed the closing prices of a specific stock over a 6 year time frame. Once this data was downloaded, there needed to be a program run that utilized each statistical model and forecasted the future prices of each stock.

Utilizing reference code, these models were built in google colab with various python libraries such as pandas, keras, and pyplot. These notebooks preprocessed the yahoo finance data and then applied the statistical model/machine learning model upon the data. This preprocessing involved splitting the data into two sections: testing and training. Training data is used to fine tune the models parameters, and fit the model so that it can accurately predict time series values for the desired stock. After the model trains, it is asked to make predictions for the stock price on days after the training dataset. These predictions are then evaluated against the testing data, which is the actual stock price. The following paragraphs will describe how these models function.

ARIMA Model

The ARIMA model is a statistical model without the use of traditional machine learning, meaning one simply must input three parameters and then they can run the model. First, however, the data must be considered stationary in order for the ARIMA model to perform effectively. Stationary data means that the variance of the data remains constant and the data fluctuates around a certain mean value.

The process of differencing solves this problem. If t represents time, the independent variable, and $y(t)$ represents the dependent variable, the process of differencing would do the following.

$$y(t) = y(t) - y(t-1).$$

The number of times a differencing operation must be conducted before the data becomes stationary is denoted as d, one of three parameters which must be used in performing the ARIMA model.

Once the data is considered stationary, the autoregressive (AR) and moving average (MA) components must have their appropriate parameters. These parameters are the p and q values. The p value represents the number of autoregressive terms. Autoregression is essentially when a model uses past time series values as input, and uses it to predict future time series values. A model could have multiple autoregressive terms, with each term corresponding to a different time series value in the past. For example, if an ARIMA model were to have 2 autoregressive terms and 0 differencing or moving average terms, then the model would be as follows.

$$\hat{y} = \mu + \Phi\, y(t-1) + \Phi\, y(t-2) + \omega_t$$

In this equation, $\hat{y}$ is the predicted time series value, μ is the mean value of the data (it is stationary so there should be a set mean), while ϕ is a coefficient. The ω_t term is an error term, which accounts for any fluctuations that are accounted for simply by random chance. It is simply a random value that is chosen from a normal distribution.

Now there is only one parameter left, the q value or the component of the moving average term. The MA (Moving Average) model is based on the principle that future time series values are based on past error (ω_{t-k}) terms.

The q value represents the amount of past error terms in the ARIMA model. Here is what two moving average terms would look like (a q value of 2).

$$y(t) = \mu + \omega_t + \theta_1\, \omega_{t-1} + \theta_2\, \omega_{t-2}$$

Here the coefficients are θ, and there are two terms, as the model is based on the error values of the time series function at t-1 and t-2.

In the python code, a method from the statsmodel library automates the process of finding optimal parameters for the ARIMA model. This method uses the training data to do so, and it will output a p, q, and d value that we can use to build our model. After this step, we are able to call a simple model.fit() and model.predict() method in order to forecast future values.

LSTM and GRU Models

The LSTM and GRU models require a different approach. First, data will be preprocessed using pandas, and then it will be split into training and testing data. The training data is used first to allow the model to fine tune its parameters so that it can effectively model that particular stock.

Both the Long Short Term Memory (LSTM) and Gated Recurrent Unit (GRU) models are recurrent neural networks. Recurrent neural networks are composed of layers, which in turn are composed of neurons. Individual neurons will apply mathematical functions on whatever input is given, and it will yield an output that is given to the neuron in the next layer. After layers upon layers within the neural network finish processing inputted values, it yields a singular output – the predicted stock price at a given time. Then, this process is repeated in order to produce several forecasted values, which represent the stock price over a period of time. However, the model needs to have its neurons and layers properly fit and optimized to the stock that it is predicting. This is where training comes in, and it is the most critical portion of creating effective time series models.

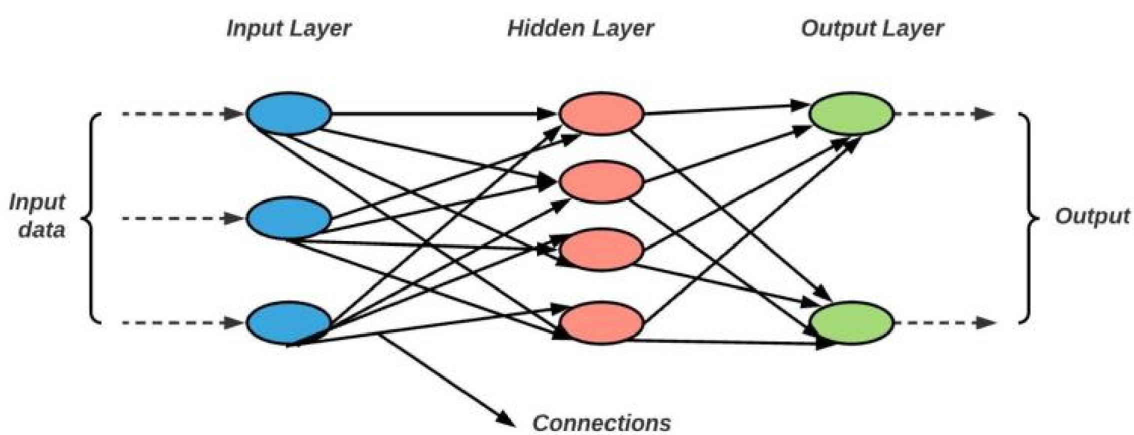

Figure 1. Diagram of a Neural Network

The mathematical functions that the neurons within a neural network do are determined in this training phase. This training stage takes place after the data is preprocessed. Here, the recurrent neural network adjusts the weights and biases of its neurons. Weights and biases are basically mathematical operations that are done by each neuron in each layer of the neural network. The process that these neurons use to fine tune their parameters and select the optimal mathematical functions determines whether they are classified as a LSTM or a GRU cell.

After the model goes through the training data, adjusting its parameters (weights and biases) each time, it now has the optimal set of parameters for this specific stock, meaning it can now recognize patterns in data. This pattern recognition can be applied to the testing data now, and using these patterns, the model will be able to now forecast the time series data.

Each version of a recurrent neural network operates slightly differently, as the cells in the neural network operate with a different architecture for LSTM and GRU.

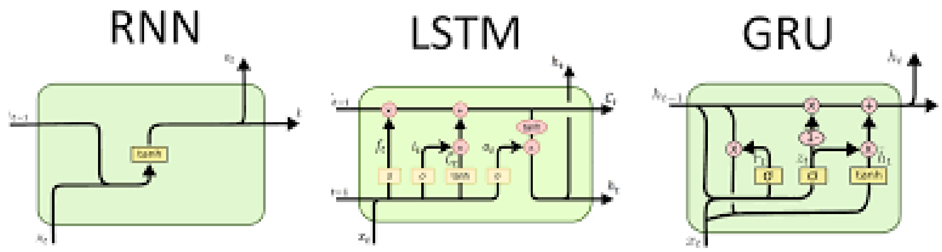

Figure 2. Diagram of architecture of LSTM, GRU, and standard Recurrent Neural Networks

They both attempt to solve the vanishing gradient problem that occurs with the base version of the recurrent neural network. This is done by designing each neuron so that important information or data points are 'remembered' and accounted for in forecasted values. Through the use of gates like the update gate and the reset gate for GRU and the forget gate and the output gate for the LSTM, these versions of RNNs are able to account for long term patterns of trends in data. This is unlike the base version of the Recurrent Neural Network, which is prone to 'forget' past patterns and rather makes its decisions based on its short term 'memory'.

Fortunately, this tedious process is simplified through keras – a python library which can create deep learning models. The LSTM and GRU layers exist in the Keras library, so when designing the model we can simply declare whether we want each layer to be LSTM or GRU. After designing this keras model, we can call model.fit() and model.predict() with the appropriate inputs to give us our intended output. Then, pyplot was used to create a graph to visualize how the predicted price data compared to the actual price data.

Lastly, as part of our methodology we ensured that for each model being run, the mean squared error will be calculated as a standard metric for determining the efficacy of a model in forecasting a stock's price. In addition, each model will predict the price of stocks that represent three different sectors of the economy: agriculture, manufacturing, and service, in order to determine if a model is more effective at predicting stock data that belongs to a certain economic sector.

Results

The models were tested for 1 month (2023, July) and trained for 6 years (2017 June 30 to 2023 June 30). Two companies were chosen for each economic sector in order to encompass a diverse range of stocks. The primary sector included extraction companies, most notably oil companies, which is why Exxon and Chevron were chosen. The secondary sector includes manufacturing companies, which could come in many forms but for the purposes of this study larger, more stable stocks are advantageous, so Apple and Pfizer stock were chosen. Note that Apple could fall into the third category as well, the tertiary sector. The tertiary sector are service companies, and for this category large technology consultant service companies were chosen – IBM and Oracle.

Primary - Extraction
Exxon
Chevron

Secondary - Manufacturing
Apple

Pfizer

Tertiary - Service
IBM
Oracle

Table 1. Mean Squared Error for predicted stock price against actual stock price

Sector	Stock	ARIMA	LSTM	GRU
Primary	Exxon	9.63	4.37	2.64
Primary	Chevron	9.10	5.23	4.01
Secondary	Apple	12.36	9.90	7.69
Secondary	Pfizer	0.38	0.26	0.20
Tertiary	IBM	17.93	2.74	1.69
Tertiary	Oracle	3.81	2.74	2.02

After applying these time series models on the stock data, the mean squared error for each stock was calculated. The table above separates each stock by the sector of the economy, and in order to make claims based on this data, we will be comparing the mean squared errors for each statistical model against each other for every stock selected.

Here is an example of final predictions using the Apple stock.

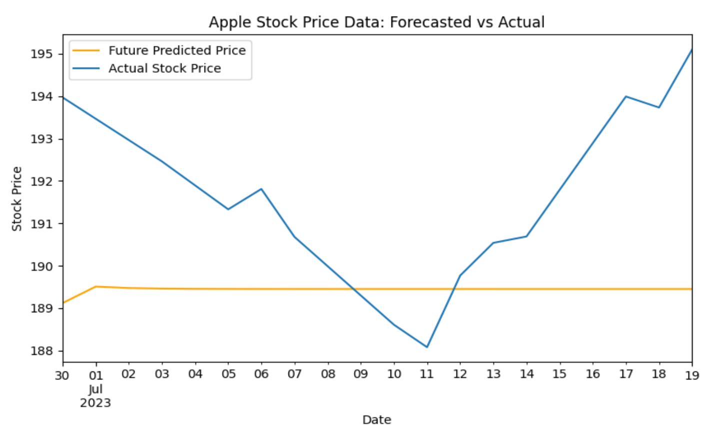

Figure 3. Apple Stock Price Prediction with the ARIMA model

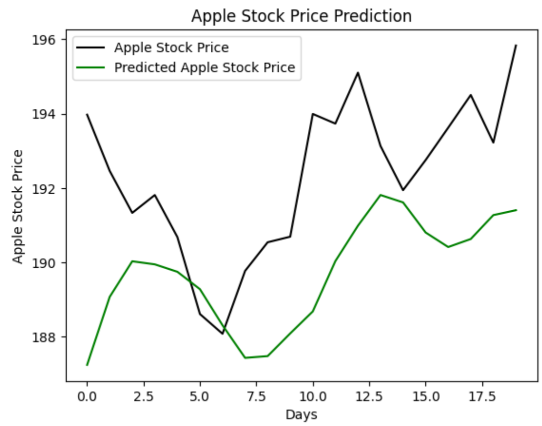

Figure 4. Apple Stock Price Prediction with the LSTM model

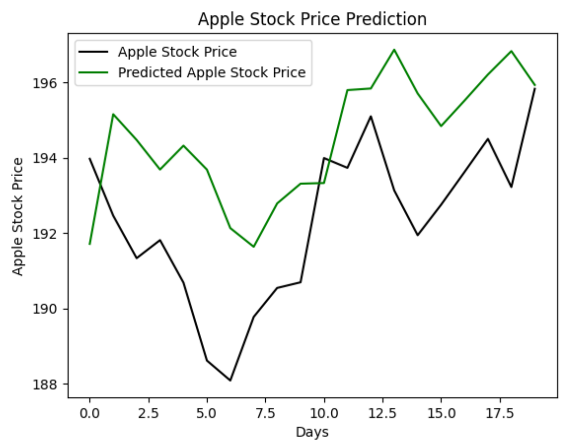

Figure 5. Apple Stock Price Prediction with the GRU model

Discussion

Essentially, what we have found is that the model that performed the best is the Gated Recurrent Unit (GRU) with the LSTM in a close second. While these differences were often minimal between GRU and LSTM, clearly GRU outperformed LSTM in nearly every metric, demonstrating the capabilities of this incredible machine learning model.

What is interesting to note is that ARIMA performed more poorly than the other two models, especially for certain stocks like that of Exxon and IBM. As a regression model, ARIMA only predicted linear trends, and it often failed to predict any stock prices that were volatile and showed any up and down movement. However, the overall conclusion is that these recurrent neural networks are more powerful in terms of analyzing time series data.

My findings were relatively consistent with other studies. With the ability of neural networks, like LSTM and GRU models to take into account patterns in the past, and incorporate them into forecasting future values, they are more sophisticated than ARIMA. Recurrent neural networks are also extremely effective when it comes to large sets of training data, which was used here. In this case, the training data was around 6 years, which would give an advantage to machine learning neural networks, rather than a regular statistical model.

There was no significant difference in the efficacy of these models across economic sectors. In fact, according to the table, the economic sector of the stock was a non factor. We can therefore conclude that the efficacy of these models is independent of whether the stock was part of the primary, secondary, or tertiary sector of the economy.

It is important to note that just because a stock price yielded a larger mean squared error when all 3 models were applied to it, it does not mean that it was significantly harder to predict or more volatile. The mean squared error as a metric, is unfortunately vulnerable to the fact that some stocks are priced much higher than others. If a stock is priced at such a high number, like Apple, then even the slightest deviation in its prediction would result in a large residual, and therefore a large mean squared error. The opposite of this effect is true, and clearly Pfizer's relatively low stock price made it much easier to predict.

However, we were still able to evaluate which models work better for which type of stock. and if a model seemed particularly effective in forecasting one sector of the economy than the other, then it would have been apparent in the data. The clear conclusion is that, despite the sector of the economy, or even despite the stock itself, the GRU model consistently yielded a lower mean squared error value than the LSTM and ARIMA models.

Conclusion

To conclude, this research paper has undertaken an extensive and rigorous comparative analysis of three pivotal forecasting models - Long Short-Term Memory (LSTM), Gated Recurrent Unit (GRU), and Autoregressive Integrated Moving Average (ARIMA) - to discern their effectiveness in predicting stock market data across diverse economic sectors.

The results highlight the effectiveness of the Gated Recurrent Unit (GRU) model in forecasting stock market data. Its remarkable capacity to capture intricate long-term dependencies and patterns in time series data positions it as a powerful instrument for accurate predictions. Simultaneously, the Long Short-Term Memory (LSTM) model demonstrates a competitive edge, closely shadowing GRU in predictive performance. Conversely, the conventional Autoregressive Integrated Moving Average (ARIMA) model registers relatively higher mean squared error values, indicating its limitations in effectively encapsulating the multifaceted nature of stock price fluctuations.

In an era where financial markets are characterized by relentless change and heightened complexity, the ability to make precise predictions assumes paramount significance. While the quest for precise predictions remains a dynamic pursuit, this study helps decision-makers navigate the intricate landscape of modern finance.

Acknowledgments

I would like to thank Haoxiang Wang of King's College for his help in mentoring me throughout this project. His guidance and aid have made this research project possible. He has provided various research articles, code snippets, and other resources that played an important role in developing this report. His guidance has enhanced my working knowledge of these complex topics.

References

Adusumilli, R. (n.d.). *Machine learning to predict stock prices*. Retrieved August 31, 2023, from https://towardsdatascience.com/predicting-stock-prices-using-a-keras-lstm-model-4225457f0233

Advances in Economics, Business and Management Research. (n.d.). https://doi.org/10.2991/iemss-17.2017.140

Dey, P., Hossain, E., Hossain, Md. I., Chowdhury, M. A., Alam, Md. S., Hossain, M. S., & Andersson, K. (2021). Comparative Analysis of Recurrent Neural Networks in Stock Price Prediction for Different Frequency Domains. *Algorithms*, *14*(8), 251. MDPI AG. Retrieved from http://dx.doi.org/10.3390/a14080251

Gamboa, J. (n.d.). *Deep learning for time-series analysis*. Retrieved August 31, 2023, from https://arxiv.org/abs/1701.01887

Li, C., & Qian, G. (2022). Stock Price Prediction Using a Frequency Decomposition Based GRU Transformer Neural Network. *Applied Sciences*, *13*(1), 222. MDPI AG. Retrieved from http://dx.doi.org/10.3390/app13010222

Mondal, Prapanna & Shit, Labani & Goswami, Saptarsi. (2014). Study of Effectiveness of Time Series Modeling (Arima) in Forecasting Stock Prices. International Journal of Computer Science, Engineering and Applications. 4. 13-29. 10.5121/ijcsea.2014.4202.

Nau, R. (n.d.). *Statistical forecasting: Notes on regression and time series analysis*. Retrieved August 31, 2023, from https://people.duke.edu/~rnau/411home.htm

Pedamallu, H. (n.d.). *RNN vs GRU vs LSTM*. Medium. Retrieved August 31, 2023, from https://medium.com/analytics-vidhya/rnn-vs-gru-vs-lstm-863b0b7b1573

Sako, K., Mpinda, B. N., & Rodrigues, P. C. (2022). Neural Networks for Financial Time Series Forecasting. *Entropy (Basel, Switzerland)*, *24*(5), 657. https://doi.org/10.3390/e24050657

Xiao, R., Feng, Y., Yan, L., & Ma, Y. (n.d.). *Predict stock prices with ARIMA and LSTM*. Retrieved August 31, 2023, from https://arxiv.org/abs/2209.02407v1