DAL 2023 Final Exam: Stock Price Prediction using LSTMs

Arun Palaniappan
Department of Mechanical Engineering
Indian Institute of Technology Madras
Chennai, India
me20b036@smail.iit.ac.in

***Abstract*—This paper provides an overview of the mathematical formulations and applications of Long Short Term Memory(LSTM) networks. First, the fundamentals of LSTM are explored, as well as the specifics necessary to comprehend the research. The problem statement and data-set are introduced in a detailed way. The data-set under consideration is one of the stock market prices, to explore how the attributes of the stock contribute to the model's closing value prediction. To accomplish this efficiently, the data has been thoroughly evaluated, and both visual and quantitative insights have been provided in this paper. Finally, the results of the LSTM model are reported in a detailed manner.**

***Index Terms*—LSTM, stock, R^2-score, RMSE, mean, RNN, visualization**

I. Introduction

LSTM, or *Long Short Term Memory*, is a powerful and special kind of Recurrent Neural Networks(RNNs). RNNs are a class of artificial neural networks designed for processing sequential data by incorporating feedback loops to capture temporal dependencies. LSTMs were designed to overcome the shortcomings of conventional RNNs in identifying and extracting knowledge from long-range dependencies in sequential data. Hence, they have been used extensively in a wide range of applications, from natural language processing to time-series analysis and stock price prediction.

LSTMs can be used to selectively store and retrieve data over extended periods, which helps to mitigate the vanishing and exploding gradient issues that standard RNNs face frequently. Due to their distinct architecture, LSTMs are especially good at tasks requiring them to understand temporal dependencies as well as complex patterns and relationships within sequential data. To test how our model will perform on unforeseen data-sets in the future, we build a train-validation split, then train the model (minimize the loss) on the train split, and then check the prediction accuracy on the validation split.

Stock prices are an important indicator of the state of the economy because they are a reflection of market dynamics. Stock price variations result from several factors, including a company's financial health, economic trends, and global events, influencing how investors perceive and value a business. Stock price fluctuations create a dynamic market in which buyers and sellers interact with one another. In the financial markets, the capacity to forecast future stock prices is critical because it enables traders, investors, and financial institutions to manage risks, make well-informed decisions, and optimize their investment strategies. Precise forecasts can steer portfolio management, facilitate identifying profitable possibilities, and reduce potential losses.

In this paper, a comprehensive understanding of LSTM fundamentals are provided. The paper also undertakes a detailed exploration of the targeted problem and elucidates how the resulting insights are supported by a combination of visual representations and quantitative evaluation.

II. LSTMs

In this section, we will describe the mathematical details of LSTMs. LSTMs is a supervised learning algorithm used for modelling the relationship between two types of variables, the target(dependent) and features(independent). The following is an overview of the jargons and mathematics utilised in the LSTM approach:

- *Cell State:* In an LSTM, the cell state is a long-term memory unit that runs through the entire sequence. It addresses the vanishing gradient issue by being selectively updated, enabling the model to maintain significant context over long stretches.
- *Hidden State:* An LSTM's hidden state is comparable to a conventional RNN's output. It adds to the model's comprehension of the sequential data by encapsulating and summarizing information from both the previous hidden state and the current input.
- *Input Gate:* In an LSTM, the input gate determines how much new information should be stored in the cell state. To control the flow of information, it employs a sigmoid activation function.
- *Forget Gate:* In an LSTM, the forget gate determines which information from the cell state should be discarded. Using a sigmoid activation function, it generates a forget vector that scales the current cell state.
- *Output Gate:* An LSTM's output gate controls the amount of data that should be transferred from the cell state to the hidden state. It regulates information flow using a hyperbolic tangent (tanh) and a sigmoid activation function.
- *Tanh Activation Function:* The hyperbolic tangent (tanh) activation function is commonly used to compress input

values between [-1,1]. It is used in LSTMs to prevent gradients from exploding or vanishing.

$$Tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \tag{1}$$

- *Sigmoid Activation Function:* The sigmoid activation function maps input values between [0,1]. In LSTMs, it is used in the gates to control the flow of information.

$$\sigma(x) = \frac{1}{1 + e^x} \tag{2}$$

A. Assumptions in LSTMs

LSTM is a powerful technique, but its validity and reliability rests upon certain assumptions that must be met for accurate and meaningful results. The features must be scaled before passing on to the network to prevent exploding or vanishing gradients. Another obvious assumption is that, LSTMs are used to model data's sequential dependencies. They assume that the information at a given time step is relevant and dependent on previous time steps. LSTMs also assume the availability of sufficient labeled data for training as insufficient data may lead to overfitting.

B. Algorithm

We will now explain the steps involved in modelling an LSTM:

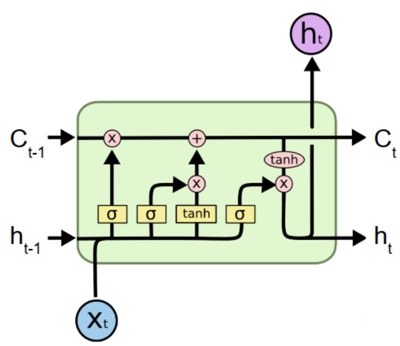

Fig. 1. A single cell of LSTM, source: adapted from [2]

1) Forget gate:

$$f_t = \sigma(W_f \cdot h_{t-1} + U_f \cdot x_t + b_f) \tag{3}$$

- Where "h_{t-1}" denotes the hidden state from the previous cell, "x_t" denotes the current input, "W_f" denotes the weight matrix associated with hidden state in the forget gate, "U_f" denotes the weight matrix associated with input in the forget gate, and "b_f" denotes the bias term of the forget gate.
- As there is a sigmoid activation function (σ), the forget gate outputs a value between 0 and 1 for each member of the previous cell state "C_{t-1}", a value of "1" means the previous cell state is very important and "0" means it's not useful.

2) Input gate:

$$i_t = \sigma(W_i \cdot h_{t-1} + U_i \cdot x_t + b_i) \tag{4}$$

$$\tilde{C}_t = tanh(W_C \cdot h_{t-1} + U_C \cdot x_t + b_C) \tag{5}$$

- Where "W_i" denotes the weight matrix associated with hidden state in the input gate, "U_i" denotes the weight matrix associated with input in the input gate, "b_i" denotes the bias term of the input gate and "W_C", "U_C" & "b_C" denote the weight matrices and bias terms associated with hidden state and input while generating new intermediate cell state ($\tilde{C}_t$) values.
- The "$tanh$" activation function creates a new candidate of cell state values, a fraction of which will be added to the previous cell state in the next step based on the value of i_t, which will range from 0 to 1.

3) Updating cell state:

$$C_t = f_t \odot C_{t-1} + i_t \odot \tilde{C}_t \tag{6}$$

- The new cell state is determined by multiplying the previous cell state "C_{t-1}" with the forget gate value "f_t", essentially forgetting all the unnecessary information from the previous cell state and also adding the intermediate cell state "$\tilde{C}_t$" multiplied by the input gate value "i_t", essentially remembering only the important information from the intermediate cell state.

4) Output gate:

$$o_t = \sigma(W_o \cdot h_{t-1} + U_o \cdot x_t + b_o) \tag{7}$$

$$h_t = o_t \odot tanh(C_t) \tag{8}$$

- Where "W_o" denotes the weight matrix associated with hidden state in the output gate, "U_o" denotes the weight matrix associated with input state in the output gate and "b_o" denotes the bias term of the output gate.
- The final output of the LSTM is determined by filtering the cell state through a sigmoid layer to select parts to output, then applying a "$tanh$" function to scale these values, which are ultimately multiplied by the output gate's value(o_t) to produce the final result.

5) Gradient descent optimization: We will now define a loss function for the model and then update the weight matrices using gradient descent optimizer to reduce the loss as much as possible.

Loss function:

As we are working with continuous value of stocks, we will use the Mean Squared Error(MSE) loss function in this case. It is defined as,

$$J_{W,U,b} = \frac{1}{N}\left[\sum_{i=1}^{N}(y_i - \hat{y}_i)^2\right] \tag{9}$$

Where y_i are the ground truth values and $\hat{y}_i$ are the values predicted by the model.

Gradient Descent:

Gradient descent is an optimization algorithm used to minimize the loss function by iteratively moving the weight

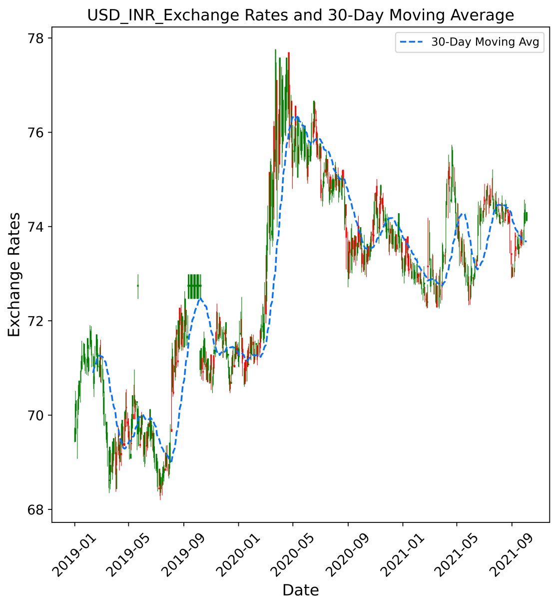

Fig. 2. USD-INR Exchange rates

Fig. 3. Cognizant Stock Price Variation

matrices and bias terms in the direction of the steepest descent as defined by the negative of the gradient. The process is repeated until the loss function converges to a minimum. It can be mathematically represented as,

$$W_i^{new} = W_i^{old} - \alpha \frac{\partial J_{W,U,b}}{\partial W_i} \tag{10}$$

$$U_i^{new} = U_i^{old} - \alpha \frac{\partial J_{W,U,b}}{\partial U_i} \tag{11}$$

$$b_i^{new} = b_i^{old} - \alpha \frac{\partial J_{W,U,b}}{\partial b_i} \tag{12}$$

Where α denotes the learning rate, a positive value determining the step size we take on each iteration.

C. Evaluation Metrics

We now define two quantities which can be used to evaluate the model.

- *Root Mean Squared Error:* It is defined as the square root of the mean sum of squares of the residuals. Let y_i be the ground truth values and $\hat{y}_i$ be the values predicted by the model. Then,

$$RMSE = \sqrt{\frac{1}{N}\left[\sum_{i=1}^{N}(y_i - \hat{y}_i)^2\right]} \tag{13}$$

- R^2 *score:* It is also known as *coefficient of determination* or *goodness of fit*. It quantifies the ratio of variance explained by the model being tested. Let $\bar{y}_i$ be the mean of ground truth values. Then,

$$R^2\text{-score} = 1 - \frac{\sum_{i=1}^{N}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{N}(y_i - \bar{y}_i)^2} \tag{14}$$

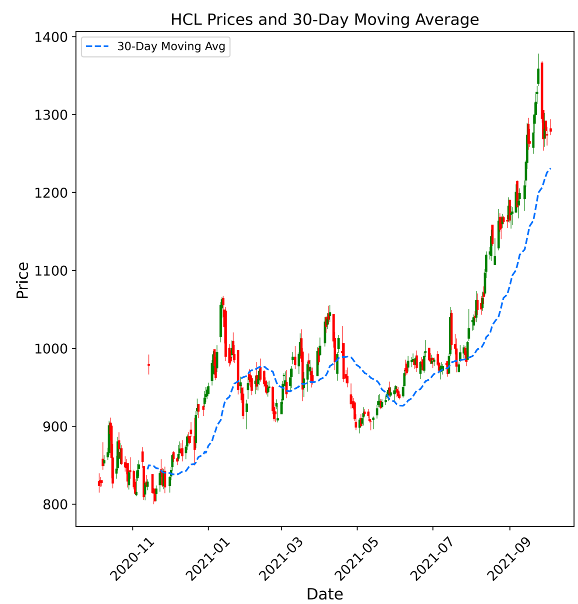

Fig. 4. HCL Stock Price Variation

D. Advantages and Disadvantages of LSTMs over RNNs

- *Advantages:* Firstly, they're really good at dealing with long-term patterns in data because they can retain information for a longer period than RNNs. Secondly, LSTMs are better at avoiding the vanishing gradient problem. Lastly, LSTMs excel at comprehending complex data sequences. They can identify the main patterns and structures in data, making them useful for tasks involving a large amount of sequential information.
- *Disadvantages:* Firstly, they are more complicated than traditional RNNs and are data-hungry models, i.e., they

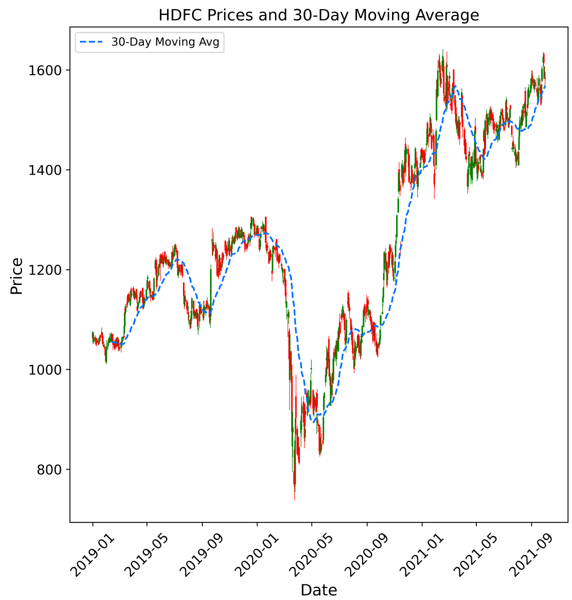

Fig. 5. HDFC Stock Price Variation

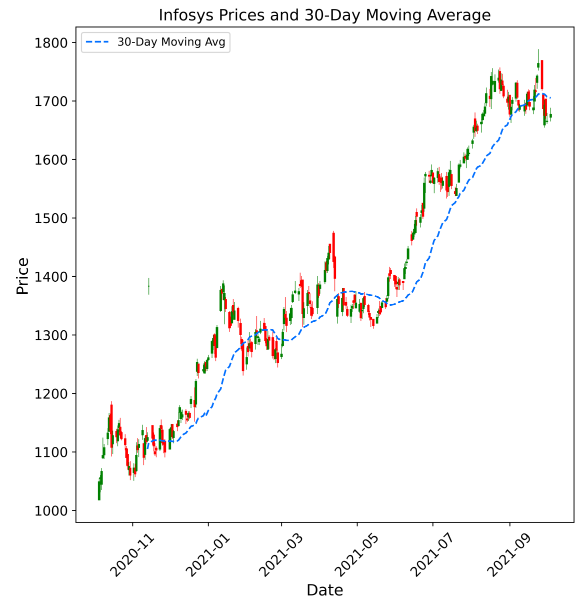

Fig. 7. Infosys Stock Price Variation

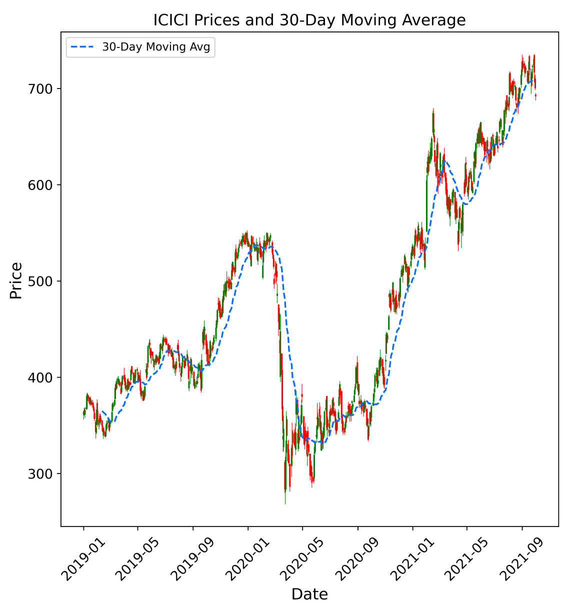

Fig. 6. ICICI Stock Price Variation

Fig. 8. SBI Stock Price Variation

require more training data in order to learn effectively. They also need a lot of computation power and time to train efficiently.

III. The Data

A data-set with information about various attributes of stock prices of 6 different companies{*Cognizant, HCL, HDFC, ICICI, Infosys, SBI*}, such as the date, the volume moved, opening price, highest price of that day, lowest price of that day and the closing price are provided to us. The ultimate aim of the study is to predict the closing price of the stock based on its attributes.

A. Data description

In this section, we describe the structure of the data-sets. Each data-set has been split into train(80%)-validation(20%) data-set. The columns from the data-set are listed below:

- *Date:* This column represents the date corresponding to each entry in the stock data-set.
- *Open:* It represents the opening price of a stock on that given day.
- *High:* It indicates the highest price of a stock on that given day.
- *Low:* It represents the lowest price of a stock on that given

day.

- *Close:* It indicates the closing price of a stock on that given day.
- *Volume:* It represents the total number of shares traded for the given stock on that particular date.

B. Data cleaning

While cleaning the data, we have to handle missing values. They were dealt in the following way:

- The data-set contained missing values which were imputed using respective mean, as all of the percentages of missing data are 3% or less.
- Then the data-set was standardized using the Gaussian standardization before feeding to the LSTM model.

C. Data Visualization

1) USD-INR Exchange:

- Before early 2020, the rates fluctuated but seemed relatively stable as evident from Fig. 2.
- As we approach early 2020, there appears to be a sharp increase in the exchange rate. This peak most likely corresponds to the COVID-19 pandemic's initial impact, which caused significant volatility in global currency markets.
- Then after the pandemic subsided, the rates went back to normal fluctuations.

2) Cognizant:

- The closing price of Cognizant fell significantly in March 2020 as a result of the pandemic's initial impact, but later recovered to near pre-pandemic levels as evident from Fig. 3.
- In certain cases, there seems to be an inverse relationship between the share price and the exchange rate.

3) HCL:

- HCL stock prices in the pre-pandemic and during the pandemic's first wave are absent.
- From Fig. 4, we can conclude that HCL stock value was steadily increasing as of September 2021.

4) HDFC:

- As evident from Fig. 5, there is a big dip in the stock values during the initial wave of the pandemic.
- We can also notice that post-pandemic, HDFC stock prices have recovered really well and are steadily increasing.

5) ICICI:

- As observable from Fig. 6, there is a sharp decline in ICICI stock closing price around March 2020 due to the pandemic, but eventually, the stock price rose again.
- We can also notice a negative correlation between the currency exchange rates and the ICICI stock value.

6) Infosys:

- Infosys stock prices in the pre-pandemic and during the pandemic's first wave are absent.
- We can also infer from Fig. 7 that Infosys stock prices are rising steadily irrespective of the exchange rates.

7) SBI:

- Similar to most of the companies seen above, the stock prices of SBI declined in pandemic times, but then rose back to above pre-pandemic levels as evident from Fig. 8.
- Additionally, we observe a negative relationship between the value of SBI's stock and currency exchange rates.

IV. The Problem

A. Outline

We have done the analysing, cleaning and visualizing the data. A train-set and a validation-set are created from the data-set. The validation-set is used to make predictions, after the LSTM model has been trained using the train-set. The validation-set predictions are used to calculate two different metrics to assess model performance.

B. LSTM modelling

The LSTM model is fitted to the training data by using *keras* module in *tensorflow* package in Python. We need to specify the parameters and hyper-parameters of the LSTM, which are listed below:

- *LSTM units:* It denotes the number of repetitive LSTM cell units in the model. After trying out different unit values, we have taken 100 units for our data.
- *batch size:* It is the number of samples that will be fed through the network before the model's weights are updated. We have taken the industry standard of 32 as the batch size.
- *epochs:* It denotes the number of times the entire data is passed through the model. As we are training an LSTM model, we training it for 100 epochs.
- *optimizer:* It is the optimizer for the learning rate of the gradient descent. We have taken the industry standard *Adam* optimizer.
- *loss function:* As defined in the previous sections, we have taken Mean Squared Error(MSE) as the loss function.

We now instantiate an LSTM model, with these parameters, for predicting the closing price. We use $RMSE$ and R^2-score as our evaluation criteria on the validation data-set. After testing on the Validation Split, the model performance is reported in the next subsection.

C. Accuracy metrics

The $RMSE$ and R^2-score of the LSTM model fit to the 6 different companies are tabulated below in TABLE I.

Company	R^2-score	RMSE	Range of stock value
Cognizant	0.99	0.39	~ 70
HCL	0.30	115.59	~ 1500
HDFC	0.96	11.19	~ 1400
ICICI	0.95	10.93	~ 700
Infosys	0.33	33.39	~ 1800
SBI	0.92	9.92	~ 400

TABLE I
Accuracy metrics

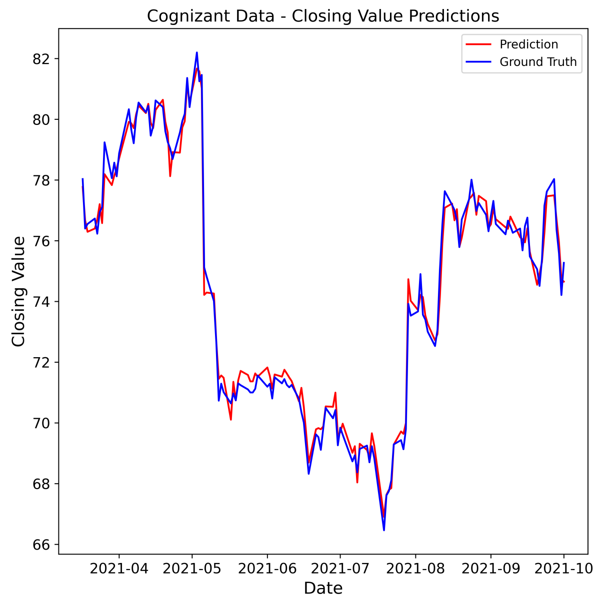

Fig. 9. Cognizant Data- Closing Value Predictions

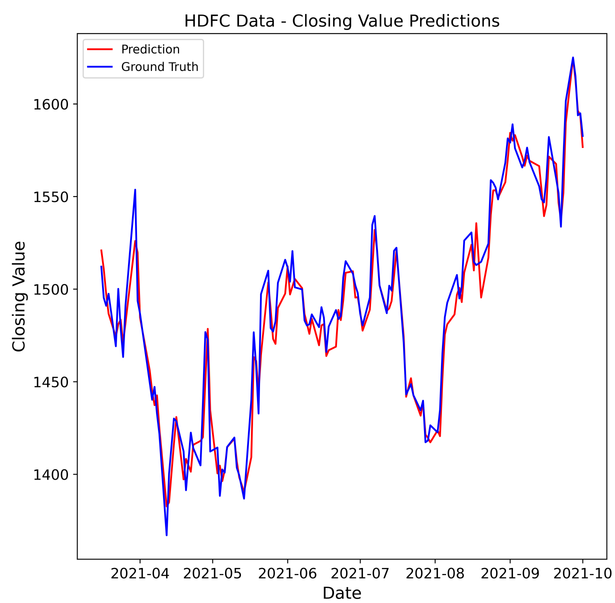

Fig. 10. HDFC Data- Closing Value Predictions

As we can observe, the model has performed exceptionally well with an average R^2-score of 0.84 across six different companies. Fig. 9, Fig. 10 & Fig. 11 are plots of prediction vs ground truth of some of the best performing LSTM models.

V. Conclusion

The conclusions that can be drawn from all the above analysis are:

- Businesses suffered greatly due to the pandemic's effects, resulting in a decline in stock prices and a general economic slowdown.

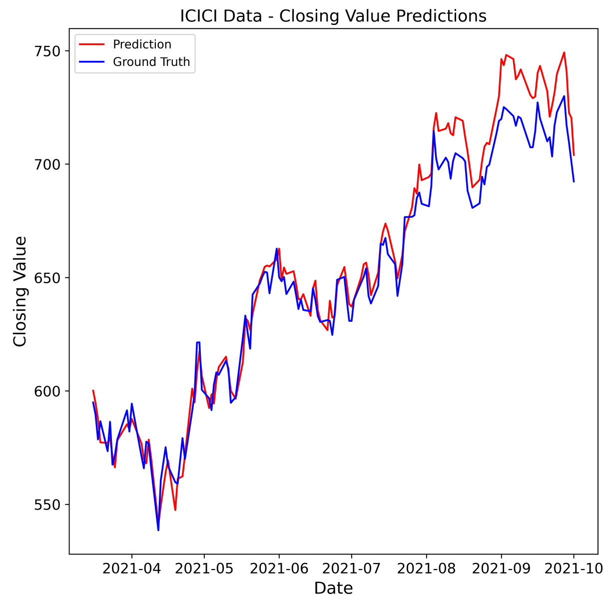

Fig. 11. ICICI Data- Closing Value Predictions

- As the income of citizens reduced during the pandemic, the spending rate also reduced. Hence the conversion rate of USD to INR also piqued.
- A visual examination of stock closing prices and exchange rates revealed discernible short-term and long-term trends.
- Models like LSTMs can be used by traders, investors, and financial institutions to manage risks, make well-informed decisions, and optimize their strategies.
- We have predicted the stock closing prices of six different companies using LSTM models effectively, and displayed the results in both quantitative and visual methods.

Future research could use similar time-series forecasting models like GRUs(Gated Recurrent Units) or ARIMA(Auto-Regressive Integrated Moving Average). We can compare the performance of these models against the currently used LSTM model, and finally, we can have an ensemble of three different models for robust predictions. We could also implement attention mechanisms in LSTMs, so that the model gets better context from the closing price of past days.

References

[1] "Long short-term memory." Wikipedia, Wikimedia Foundation, 17 Nov. 2023, *https://en.wikipedia.org/wiki/Long_short-term_memory* (accessed 20 Nov. 2023)

[2] Understanding LSTM Networks. (n.d.). *https://colah.github.io/posts/2015-08-Understanding-LSTMs/* (accessed 20 Nov. 2023)

[3] Gareth James, Daniela Witten, Trevor Hastie, Robert Tibshirani. *An Introduction to Statistical Learning : with Applications in R*. New York :Springer, 2013.