Machine Learning Approaches in Stock Price Prediction

D. Shravani[1], P. Shashi Rekha[2], Y. Aparna[3], N. Keerthi[4], G. Surekha[5]

[1,2,3,4]Student, Computer Science Engineering, Vidya Jyothi Institute of Technology
[5]Assistant Professor, Computer Science Engineering, Vidya Jyothi Institute of Technology

Abstract

One of the convoluted things to do in the stock market is to make a prediction or do an analysis of the stock price. Market volatility and several inferior and sovereign factors are impacting stock value. As a result, any stock market expert will have difficulty predicting the growth or collapse of a particular stock. Earlier methods of stock price prediction involved the use of artificial neural networks (ANN) and convolutional neural networks (CNN), which have an average error loss of 20%. In this activity, we will implement a model using long short-term memory (LSTM), after which we will compare and evaluate its results with those of other machine learning techniques, which will help us predict values with a lower percentage of error. Here, we will be using the Tata Global Dataset. So here we will be constructing the stock price prediction model for TATA Beverages Limited. The stock price data will be supplied as a comma-separated file (.csv) that may be opened and analyzed in Excel or a spreadsheet.

Keywords: Open Value, High Value, Low Value, Last Value, Close Value, Total Trade Quantity, Turnover (Lacs), RMSE, LSTM.

1. Introduction

Stocks are purchased by investors in the hopes of making a profit. This type of income is sometimes referred to as "stock returns," and it might include gains from stock trading or dividends paid. Dividends may be distributed to shareholders on a quarterly, semi-annual, or annual basis from profits earned. Stock prices and returns are bound to be impacted by various kinds of risks within a country as well as global events. Many experts have studied the stock market scientifically and meticulously to produce regulations (Selvamuthu, 2019, 5–16) for its operation. The study's findings, however, suggest that stock market fluctuations are unconnected. As a result, there has been an increase in demand for associated financial services, and stock price forecasting has become a topic that professional analysts and investors regard highly. Many factors, including but not limited to economic fundamentals, have been linked to stock price volatility, including market expectations and faith in the company's management and operations. (Nayak,2016,441-449). Because of technological advancements, the general public now has access to a vast amount of information in a much shorter period. As a result, stock research has become more challenging, as a large volume of data must be analyzed in a short amount of time.

As AI technology advances, deep learning techniques are increasingly being used in a wide range of research fields and practical scenarios (Parmar, 2018; 574-576). Natural language processing, picture recognition, medical forecasts, and other applications are examples. As a result of the advent of deep learning, the neural networks utilized in various application areas have likewise evolved and improved. A highly exact prediction of a future trend is crucial in a financially volatile market like the stock market. Because of the financial crisis and the demand for earnings, it is vital to have a credible stock price prediction. The use of sophisticated machine learning algorithms is required to predict a nonlinear signal. So, in this case, we've chosen an advanced machine learning model called LSTM to build a stock prediction model.

The following is the project flow: fundamental analysis, technical analysis, and model deployment. The first phase, called "fundamental analysis, evaluates the company's future profitability while taking the latest trends and set of circumstances into account. The second phase of our proposal consists of developing a dashboard that includes reading charts, statistical information, and current stock market trends. The final stretch is Production/Deployment, which entails migrating our custom ML model into any cloud.

2. Literature Survey

Many studies agencies are investigating using social media analytics to expect inventory marketplace trends. Bollen' s book is the most well-known in this field (2011). They looked into whether the collective mood states of the public (happy, calm, and anxious) derived from Twitter feeds are related to the Dow Jones Industrial Average value. For their prediction, they used a fuzzy neural network. According to their findings, public mood states on Twitter are strongly correlated with the Dow Jones Industrial Average. [4] has shown a survey on stock market prediction using various ML techniques and even discussed the latest techniques. In this study, they used ANN variations to forecast stock prices. However, the efficiency of ANN forecasting is dependent on the learning algorithm used to train the ANN. The Levenberg-Marquardt (LM), Scaled Conjugate Gradient, and Bayesian Regularization algorithms are compared. Observing this paper, the loss rate is high, and even so, it is not highly recommendable for continuous data; hence, the development of a stock model using this model is not recommended. In two ways, [1] forecasted stock prices The first is a daily prediction model, which forecasts the future trend for the following day, and the second is a monthly prediction model, which forecasts the trend for the following month using only historical data. Chartist or technical theories are used, as well as foundational or innate value analysis. The proposed method is founded on technical theories. The basic premise of this theory is that history tends to repeat itself. Based on historical data, a forecast could be used to predict future trends. This paper concentrates mainly on history rather than observing all the internal factors of a company that is responsible for a stock's rise or fall. So, there are chances that the stock price predicted may be wrong, as all the time it can't be raised or lowered depending on its previous history; instead, it mostly observes all the factors such as economic growth, investor resources, exchange rates, etc. [9] predicted stock closing prices for five different companies in different sectors of operations, where he evaluated models using strategic indicators such as RMSE and MAPE. This project is completely vague, as he randomly evaluated the models and did not consider particular parameters. [10] predicted stock price based on deep learning technology, where he took social media information and used the Doc2Vec version to construct lengthy textual content characteristic vectors from social media. Mostly, the Doc2vec feature is used when we need to implement feature engineering or extraction to our text, and here we have used it for the same to build long text feature vectors and then decrease their dimensions, and there is a chance of missing text sometimes here in the process of reducing it may be a single letter, but that can even change the context of the sentence and lead to wrong predictions. That's why Doc2Vec is not suitable and is not considered in most cases. [5] and Shay B. Cohen used a pre-selected stock collection to predict stock movement using tweets and historical prices. They offered a hybrid objective with a temporal auxiliary to capture predictive dependencies flexibly. [2] focused on the use of regression techniques to predict stock market price values. Factors considered were open, close, low, high, and volume. They focused on the use of LSTM and regression-based machine learning models. Future computer modelling stock market prediction research directions were identified in [7]. To discover relevant consensus journal papers from the previous two decades and classify research that employs comparable methodologies and situations, a comprehensive literature methodology is applied. Four categories emerged from the research: artificial neural networks, support vector machines, optimization strategies integrated with other techniques, and mixed or other artificial intelligence systems. Each category's research is analyzed to discover common results, noteworthy findings, restrictions, and areas that require additional exploration. [8] has used some ANN-heuristic algorithms and time series models like ARIMA and ARIMA to forecast stock prices and finally compared results with each other. Here by observations, we can conclude that they used the ARIMA time series model, which is not recommended for stock price models because it can't identify the correct model from the possible models. [6] employed sentiment analysis to collect meaningful information from a variety of textual data sources, as well as a blended ensemble deep learning model to forecast future stock movement. Sentiment analysis and neural network-based concepts were applied to Twitter tweets, and the correlation between a company's stock market performance and sentiments in tweets was examined [3].

2.1 Feasibility study

Technology Side

Define the Problem: The problem is to accurately predict stock prices using machine learning.

Data Collection: Collect historical stock price data, company financial data, news articles, social media sentiment, and other relevant data sources.

Data Preparation: Clean and pre-process the data, remove missing values, handle outliers, normalize the data, and select relevant features. Model Selection: Select an appropriate machine learning model, such as regression models, time-series models, or deep learning models.

Model Training and Evaluation: Train the selected model on the prepared data and evaluate its performance using various metrics such as accuracy, precision, recall, and F1 score.

Implementation and Deployment: Implement and deploy the machine learning model on a web platform, a mobile application, or desktop software.

Economic Side

Business and Financial Analysis: Conduct a business and financial analysis to evaluate the potential profitability of the project. Estimate the revenue generated from the product, the costs involved in its development, and the potential return on investment.

Legal Side

Data Collection: Ensure compliance with data privacy regulations, such as GDPR or CCPA, when collecting and processing data.

Implementation and Deployment: Ensure compliance with intellectual property laws and licensing agreements when deploying the system.

Operational Side: Data Collection: Plan for efficient and reliable data collection to ensure the availability of data for analysis.

Implementation and Deployment: Plan for user support and maintenance of the deployed system to ensure smooth operation. Overall, the feasibility of stock price prediction using machine learning will depend on the availability and quality of data, the suitability of machine learning models for the problem, and the cost-effectiveness of the project. Conducting a feasibility study can help identify potential risks, challenges, and opportunities associated with the project and enable informed decision-making.

3. Methodology

In this project, we will implement a model using long short-term memory (LSTM) and compare and evaluate its results with those of other machine learning models that predict stock price with a lower percentage of error. Here, we will be using the Tata Global Dataset, and with its detailed design and assessment, this work adds to the stock analysis research community in both the financial and technical domains. Stock market prediction or analysis is one of the most difficult jobs to complete. Some of them might be market volatility and various dependent and independent variables that influence stock value. As a result, it is difficult for any stock market expert to accurately predict the rise or fall of a specific stock. So here we are building the stock price prediction model for TATA Beverages Limited. The stock price data will be provided as a comma-separated file (.csv) that may be opened and analysed in Excel or a spreadsheet.

In this project, we have trained all the algorithms individually and evaluated their performance using RMSE for each dataset. After the comparison with previous models, LSTM outperformed them. The main finding in this model is that this method had the lowest RMSE value compared to other models.

3.1 Fundamental vs. technical analysis of stocks

Fundamental and technical stock analysis are at the opposite ends of the detailed competitive spectrum.

3.1.1 Fundamental analysis

Explores the inherent worth of a company's value, including but not limited to tangible assets, balance sheets, management effectiveness, development strategies, and consumption patterns; in short, all of a company's fundamentals. The chart patterns, as a good indicator for long-term investment, quantify revenues, securities, charges, borrowings, and so on, also using historical and current data. In general, fundamental analysis results don't vary about short-term news.

3.1.2 Technical analysis

Stock market operations provide visible data, such as stock prices, past and current returns, and the volume of previous trades; quantitative data that may be utilized to detect technical indicators and capture stock market movement patterns. Technical analysis, like fundamental analysis, focuses on past and present data, although it is mostly used for short-term trading. Technical analysis results are often impacted by news due to their short-term nature. The use of moving averages (MA), assistance and return to the beginning position, trend lines and channels, and other basic and technical approaches are widespread.

We'll solely look at technical indicators for our proposal, with an emphasis on the simple MA and exponential MA approaches for forecasting stock values. In addition, we will use LSTM (Long Short-Term Memory), a supervised neural structure for time series, to create prediction models and compare their performance to our technical analysis.

3.2 Stock prices as a time series

Stock prices aren't merely numbers generated at random, despite their volatility. As a result, they may be examined as a set of discrete-time data or observations collected at different times in the past (usually daily). Time series forecasting can help with stock predictions (predicting future values based on historical values).

We need a way to gather time-series data because it is sequential in nature. The most intuitive approach available is MA, which is capable of smoothing out short-term variations. The next section will provide further details.

3.3 Dataset Collection and Analysis

We will use the TATA Beverages stock prices for this project.

Stock prices for TATA Beverages Data can be downloaded from the Nasdaq data link by logging into your Nasdaq account and saving it in a suitable format, such as CSV.

We'll train the model with the oldest 80% of the data and keep the most recent 20% as a holdout testing set. We will use the root-mean-square error (RMSE) and average absolute proportion of total error (MAPE) metrics to assess the efficacy of our methods. The lower the value for both metrics, the better the prediction.

We will be implementing some machine learning models such as linear regression and extreme gradient boosting (XG Boost) and then using the advanced machine learning technique LSTM (Long Short-Term Memory).

3.4 Evaluation Metrics

As the stock market is essentially a predictive model, our current model evaluation metrics will be the RMSE (Root Mean Square Error) and MAPE (Mean Absolute Percentage Error). Both are reliable predictors of forecast accuracy.

$$\text{MAPE}=\frac{1}{N}\times\sum_{t=1}^{N}\left|\frac{At-Ft}{At}\right|$$

$$\text{RMSE}=\sqrt{\frac{1}{N}\times\sum_{t=1}^{N}(At-Ft)^{2}}$$

where N denotes the number of time points, At denotes the current or true stock price, and Ft denotes the predicted or forecast value.

3.5 Types of Metrics Used in Our Project

RMSE (Root Mean Squared Error) The root-mean-square error (RMSE) is a non-linear evaluation rule that takes the average phase error into account. It's calculated by dividing the squared difference between prediction and inference by the number of variables. (Sci-kit learn). Here, in our project, we are comparing the LSTM algorithm to the other existing algorithms, and RMSE is used for evaluating the algorithms.

Mean Absolute Error (MAE) MAE calculates the average magnitude of mistakes in a series of predictions without accounting for their direction. It's the average of the absolute differences between prediction and process indicators across the test sample, weighted equally for each unique difference. The average model prediction error measurements MAE and RMSE are both stated in units of a variable of interest. Both measures have a 0–100 range and are indifferent about the direction of inaccuracy. Because these are bogus scores, the lower the value, the better.

3.6 Coefficient Matrix

Instead of estimating the relationship for each group of characteristics, we can create a correlation matrix if we have a large set of variables. All diagonal entries will be 1 in this case, and the correlation will indeed be present for all statistically possible combinations. The two techniques for measuring the correlation as well as its matrix are Pearson and Spearman.

Using Pearson

The Pearson correlation measures the strength of a two-variable, linear relationship. It has a value between -1 and 1, with -1 going to represent total negative linear association, 0 going to represent no correlation, and +1 representing total positive correlation.

Using Spearman

The direction of affiliation between X (the variable) and Y is shown by the sign of the Pearson correlations (the dependent variable). The Spearman's correlation value is positive if Y tends to climb as X rises. A "nonparametric" coefficient is what Spearman's correlation coefficient is called.

4 A Stock Price Prediction Using LSTM

Long Short-Term Memory (LSTM) is a perceptron that can learn how to organize items in a series. Rather than having this framework pre-specified and fixed, LSTMs may learn the context needed to make decisions in time series circumstances. Despite its potential, there is significant debate over whether LSTMs are suitable for time series prediction.

4.1 Using an LSTM model to forecast stock prices

To address the issue of The LSTM deep learning technology was created to do gradient descent in extended sequences. An update gate, a forget gate, and an output gate are the three gates of the LSTM. Whether each cell state element is updated is determined by the update and forget gates. The output gate tells the following layer how much data to extract as activation functions. Below is the lengthy short-term memory framework that we are willing to employ. We'll utilize two layers of LSTM components with a single hidden layer in between to avoid over-fitting. (Kang, 2017).

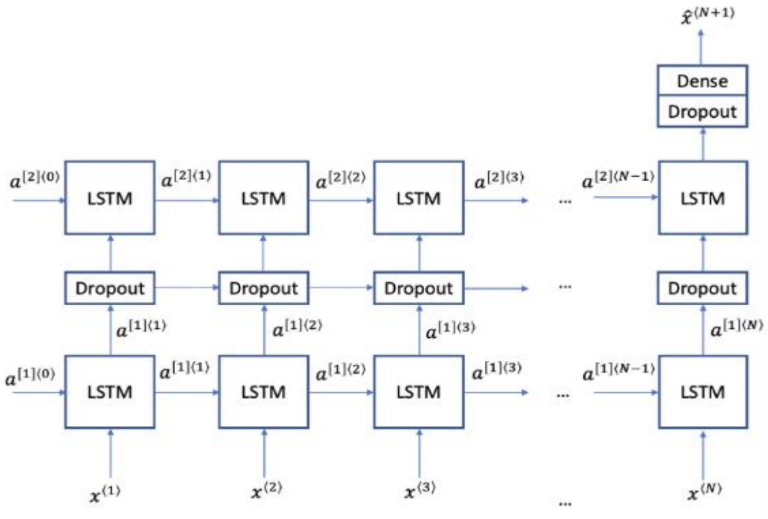

$\hat{x}^{\langle N+1\rangle}$
Dense
Dropout
$a^{[2]\langle 0\rangle}$
LSTM
$a^{[2]\langle 1\rangle}$
LSTM
$a^{[2]\langle 2\rangle}$
LSTM
$a^{[2]\langle 3\rangle}$
...
$a^{[2]\langle N-1\rangle}$
LSTM
Dropout
Dropout
Dropout
...
Dropout
$a^{[1]\langle 1\rangle}$
$a^{[1]\langle 2\rangle}$
$a^{[1]\langle 3\rangle}$
$a^{[1]\langle N\rangle}$
$a^{[1]\langle 0\rangle}$
LSTM
$a^{[1]\langle 1\rangle}$
LSTM
$a^{[1]\langle 2\rangle}$
LSTM
$a^{[1]\langle 3\rangle}$
...
$a^{[1]\langle N-1\rangle}$
LSTM
$x^{\langle 1\rangle}$
$x^{\langle 2\rangle}$
$x^{\langle 3\rangle}$
...
$x^{\langle N\rangle}$

Figure 1: Architecture of LSTM

Forget Gate

The forget gate, as its name implies, decides which information to discard from the current cell state. Mathematically, it employs a sigmoid function to output or return a value between [0, 1] for each value from the previous cell state (Ct-1); '1' denotes "completely passing through," whereas '0' denotes "completely filtering out."

Input Gate

It determines which new information is added to and stored in the current cell state. A sigmoid function is used in this layer to reduce the values in the input vector (it), and then a tanh function squashes each value between [-1, 1]. (Ct). The multiplication of it and Ct on an element-by-element basis represents new information that must be added to the current cell state.

Output Gate

The output gate is used to control the flow of output to the next cell state. An output gate, like an input gate, employs a sigmoid function followed by a tanh function to sieve out undesired information and keep only what we've decided to allow through. Even though training LSTMs eradicates the "vanishing gradient" problem (weights become far too small, underfitting the model), it retains the "exploding gradient" problem (weights become too large, overfitting the model). Training LSTMs is simple using scripting language frameworks such as TensorFlow, PyTorch, Theano, and others, but as with RNNs, a GPU is required for training deeper

LSTM networks. LSTMs are widely used in works such as voice recognition, language generation, image OCR, etc. because they handle long-term dependencies. This technique is also gathering steam in object detection (particularly scene text detection).

4.2 Dissecting the LSTM Architecture

1. **Learn Gate** accepts events (Et) and previous short-term memories (STMt-1) as input and stores only relevant information for prediction. (Srivastava, 2017)

Calculation

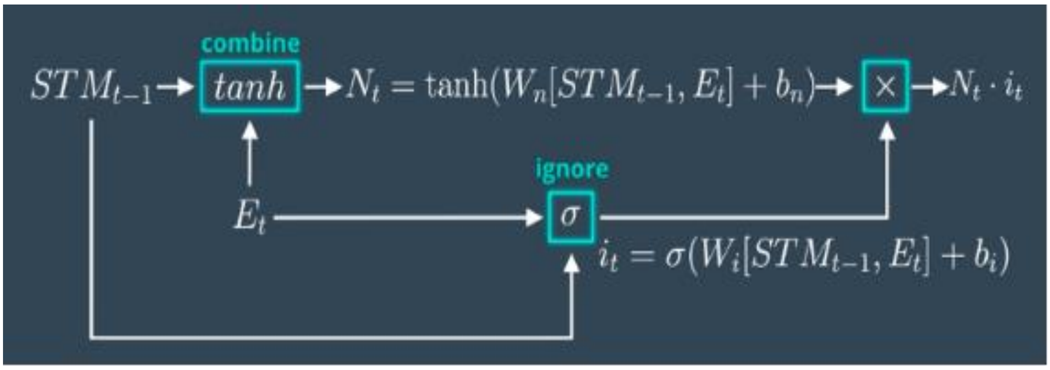

combine
STM_{t-1}
tanh
$N_t = \tanh(W_n[STM_{t-1}, E_t] + b_n)$
×
$N_t \cdot i_t$
ignore
E_t
σ
$i_t = \sigma(W_i[STM_{t-1}, E_t] + b_i)$

Figure 2: Learn Gate Architecture

The previous short-term memory STMt-1 and the present occasion variable Et are combined and multiplied with weight matrices Wn with a bias, which is then transferred to the tanh (hyperbolic tangent) feature to integrate quasi, resulting in matrix Nt. We calculate one ignore factor by linking STMt-1 and the current event tensor Et, multiplying with the weight matrix Wi, and sending it through the Sigmoid activation function with some bias. To acquire the learn gate result, multiply the learn matrix Nt by the ignore factor. (Udacity).

2. **The Forget Gate** This gate takes as input Prior Long-Term Memory (LTM-1) and gets to decide which data to maintain and which to discard.

Calculation

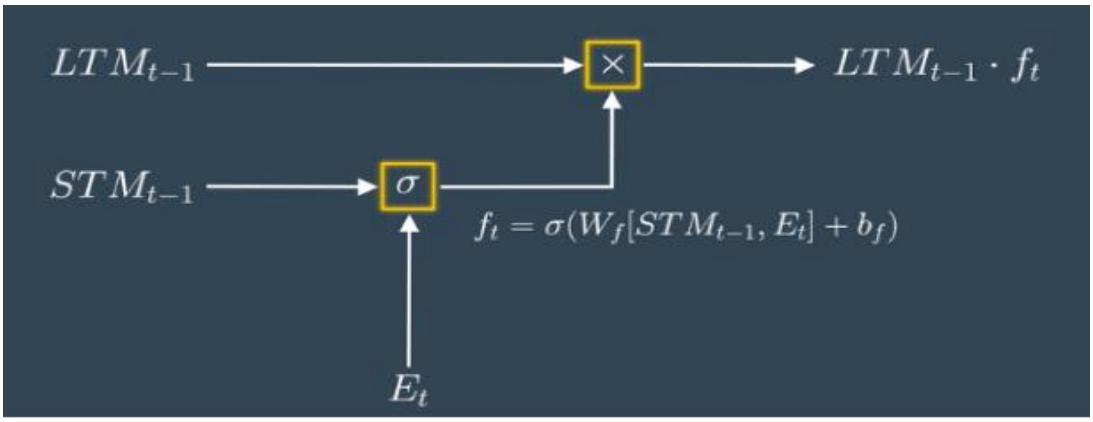


Figure 3: Forget Gate Architecture

3. **The Remember Gate** To generate output, combine previous short-term memory (STMt-1) and current events (Et).

Calculation

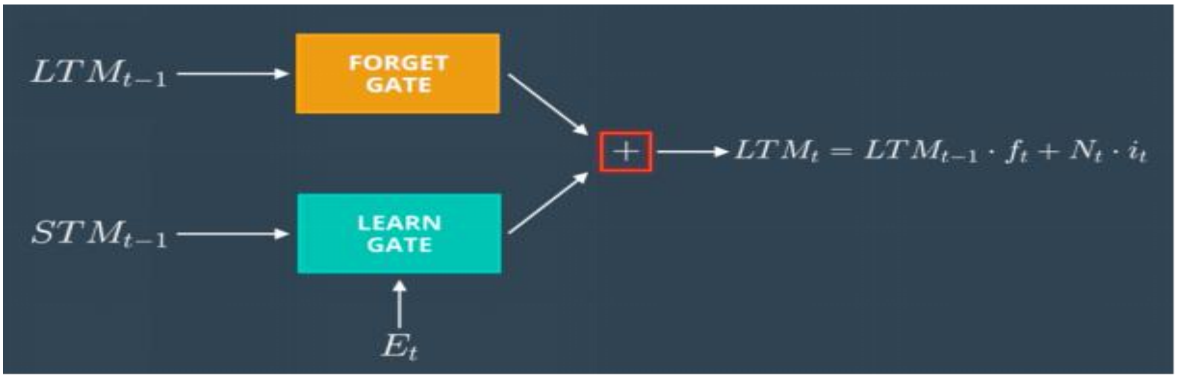


Figure 4:Remember Gate Architecture

4. **The Use Gate** combine important information from the previous long-term memory and the previous short-term memory to create STM for the next cell and to produce output for the current event.

Calculation

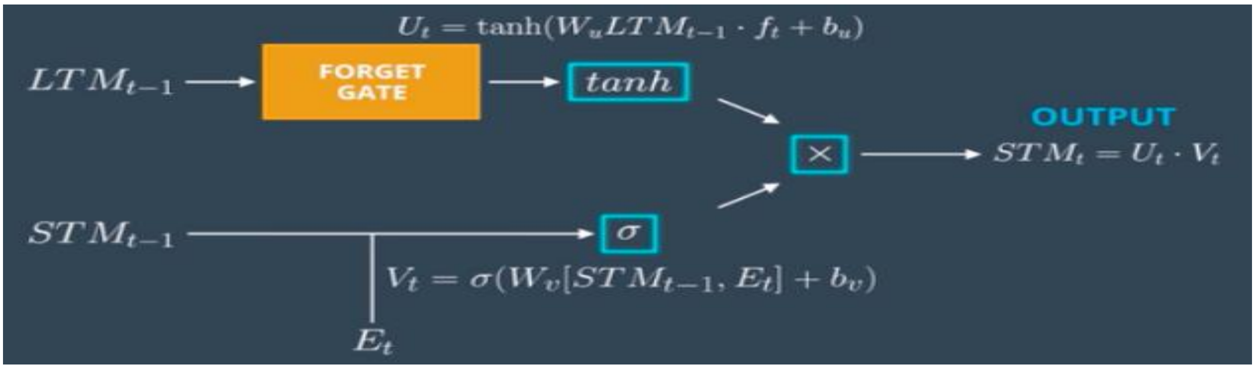

$U_t = \tanh(W_u LTM_{t-1} \cdot f_t + b_u)$
LTM_{t-1}
FORGET GATE
$tanh$
OUTPUT
$STM_t = U_t \cdot V_t$
STM_{t-1}
σ
$V_t = \sigma(W_v[STM_{t-1}, E_t] + b_v)$
E_t

Figure 5: Use Gate Architecture

Previous long-term memory (LTM-1) is fed through the Tangent activation function with a bias to yield Ut.

To generate Vt, previous short-term memory (STMt-1) and current events (Et) are merged and passed through a sigmoid activation function with a bias.

The outputs of the use gate, Ut and Vt, are multiplied together to generate the use gate's output, which also serves as STM for the following cell. (Udacity)

4.3 Working of LSTM

LSTM is a special network structure with three "gate" structures. Three gates are placed in an LSTM unit: the input gate, the forgetting gate, and the output gate. While information enters the LSTM's network, it can be selected by rules. Only the information that conforms to the algorithm will be left, and the information that does not conform will be forgotten through the forgetting gate.

The experimental data in this paper are actual historical data downloaded from the Internet. Three data sets were used in the experiments. It is needed to find an optimization algorithm that requires fewer resources and has a faster convergence speed.

- Used Long Short-term Memory (LSTM) with an embedded layer and the LSTM neural network with an automatic encoder.
- LSTM is used instead of RNN to avoid exploding and vanishing.
- In this project, Python is used to train the model and MATLAB is used to reduce the dimensions of the MySQL database, which is used as a dataset to store and retrieve data.
- The historical stock data table contains information on the opening price, the highest price, the lowest price, the closing price, the transaction date, the volume, and so on.

5. Results

In this section, we go over the forecasting techniques we used and the results we obtained with them in detail. We start with regression techniques and then move on to advanced machine learning techniques. We calculated the prediction RMSE of the models. The metrics are defined further below.

5.1 Comparison of all applied models

Model Name	RMSE
LSTM	0.126800
Ridge	0.395366
Linear Regression	0.395432
Lasso	0.460331
Gradient Boosting	8.528331
Extra Trees	8.595852
Random Forest	8.674361
XGBoost	8.700223
Bagging	8.784813
Decision Tree	9.210396
LGBM	9.720238
Cat Boost	10.81173
AdaBoost	18.03755
SVM	41.13916

Table 1: Comparison table

In linear regression, we have achieved a 0.395 RMSE, which is high compared to LSTM due to the fact that by its nature it looks only at linear relationships and is even very sensitive to outliers. It can't handle a complete description of relationships among labels. Next coming to KNN As with large datasets, the cost of calculating the distance between the new point and each existing point is prohibitively expensive, degrading the algorithm's performance. It is mandatory to do feature scaling before applying KNN models; otherwise, there are huge chances of producing wrong predictions. One of the most significant benefits of using LSTM networks is that they address the vanishing gradient problem, which makes network training difficult for long sequences of words or integers. Gradients are used to update RNN parameters, and for a long sequence of words or integers, these gradients become smaller and smaller until no network training is possible. LSTM networks help to solve this problem by capturing long-term dependencies between keywords or integers in sequences separated by a large distance. They are the best fit for time series analysis as they just remember the important layers or networks that are required for model building, and even if we can observe, we got a lower RMSE score, i.e., 4, as compared to previous machine learning models. Here are all the model testing results that have been performed and their RMSE values. We can observe that the LSTM model has the lowest RMSE compared to any other model.

5.2 Parameters Used

The following are the parameters and their analysis/Meaning which have been used in our dataset

Column Name	**Column Definition**
Column: Date	Stock Price on This Date
Column: Open	A share's opening price
Column: Close	A share's Closing price
Column: Volume/Trade Quantity	The total number of shares exchanged
Column: High	The day's highest share value
Column: Low	The day's lowest share value
Column: Turnover	The day's Total Turnover share value

Table 2: Parameters table

LSTM Algorithm Analysis

Step 1 Begin.

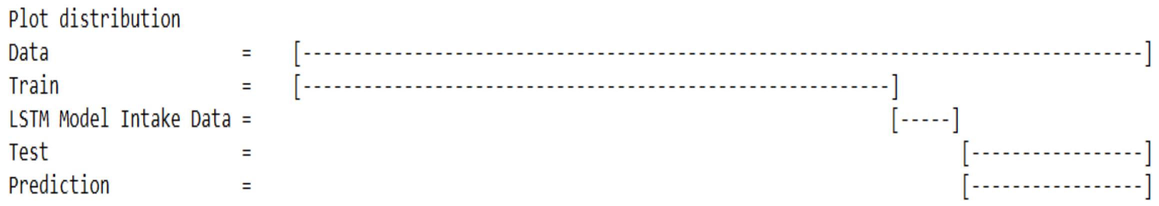
Step 2 Read the data price and import the dataset into the data structure.

Step 3 Carry out a feature scale.

Step 4 Create a training model with epoch 1 and compare outcomes after shifting epochs.

Step 5 Using plotting tools, make predictions and see the outcomes.

The suggested LSTM mode (Pothuganti, 2019, 92) is implemented in machine learning and forecasts the future price of TATAMOTORS shares based on past data. The visualisation of the TATASHARE prediction is shown in the image below. We created an algorithm in our project that forecasts the stock price of a share over a particular time period; the graph below displays the expected price of TATA MOTORS. The graph below shows the plotted form of our algorithm's result using LSTM units for obtaining RMSE.



```
Plot distribution
Data                   =    [----------------------------------------------------------------------------------]
Train                  =    [---------------------------------------------------------]
LSTM Model Intake Data =                                                              [-----]
Test                   =                                                                     [-----------------]
Prediction             =                                                                     [-----------------]
```

Figure 6: LSTM Intake

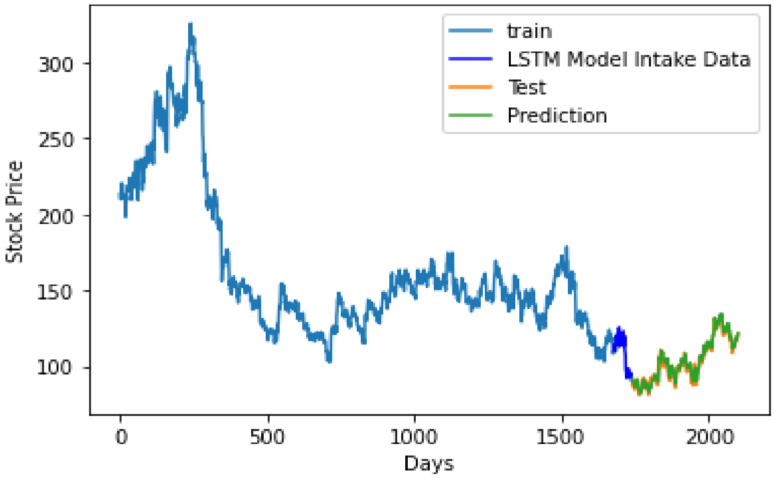


Figure 7: Prediction vs Actual graph

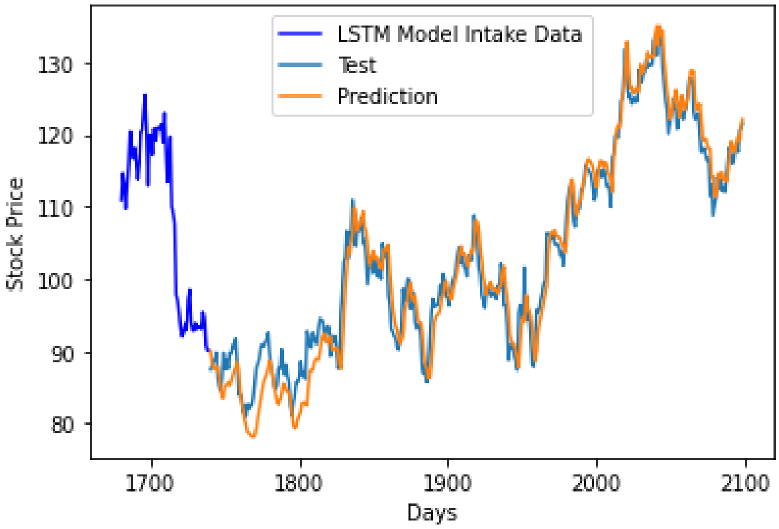


Figure 8: Prediction vs Actual graph (closer Look)

6. Conclusion

LSTM has high capability in understanding the complex hidden patterns irrespective of seasonality, stationarity, and sometimes the noise in the data compared to other models such as linear regression, KNN, and random forest, and LSTM can generalize the data we are providing and can perform well on unknown data as well. When we compare the results, for almost all the products, the RMSE scores are lower in LSTM models. This research project provided a holistic analysis of various algorithms available for demand forecasting of products in the retail industry.

Recommendations

The plan is to train a few neural networks at different time intervals. Train the model with different inputs like open price, volume, or any other variables that impact the stock price; this may result in a more accurate model. Researchers might get better results.

References

1. Li- "A novel ensemble deep learning model for stock prediction based on stock prices and news", https://doi.org/10.1007/s41060-021-00279-9 , 17 Sept 2021.
2. Razavi- "Forecasting stock price using integrated artificial neural network and metaheuristic algorithms compared to series models", https://doi.org/10.1007/s00500-021-05775-5, 8483-8513, 25 April 2021.
3. Vegan- "Stock Price Prediction using Hidden Markov Models and understanding the nature of underlying Hidden States" May 2021.
4. Hachcham, A., "The KNN Algorithm – Explanation, Opportunities, Limitations" http://matlab.izmiran.ru/help/toolbox/nnet/backpr25.html, 2021.
5. DEAGON, B. "Amazon Vs. Walmart: The Epic Battle of Retail Kings Gets Hot", (2021).
6. Polamuri, "A Survey on Stock Market Prediction Using Machine Learning Techniques", 10.1007/978-981-15-1420-3_101, 924- 931, May 2020
7. Strader, "Machine Learning Stock Market Prediction Studies: Review and Research Directions", https://scholarworks.lib.csusb.edu/jitim/vol28/iss4/3 ,63-83,2020.
8. Mehar Vijh, "Stock Closing Price Prediction using Machine Learning Techniques", 599-606,2020.
9. Xuan Ji, "A stock price prediction method based on deep learning technology", https://www.emerald.com/insight/2398-7294.htm. ,55-72, Sept 2020.
10. Andrewngai, ". Using AWS Sage maker and Lambda to Build a Serverless ML Platform. Build a serverless ML application to predict flight delays on AWS",2020.
11. Parmar, "Stock Market Prediction Using Machine Learning", 574-576,2018
12. Xu, "Stock Movement Prediction from Tweets and Historical Prices", 1970-1979, July 2018.
13. Nayak, "Prediction Models for Indian Stock Market",441-449,2016.
14. Pagolu, Venkata Sasank, "Sentiment Analysis of Twitter Data for Predicting Stock Market Movements",2016.