Predicting Stock Market prices from Yahoo Finance using LSTM and Random Forest

ABSTRACT

Yahoo Finance data contains voluminous historical information about the daily stock prices, trading volume, market capitalization and other metrics. Stock prediction is a challenging task due to the complexity, volatility and inaccurate incomplete data quality of stock market. The stock market is subjected to a lot of noise which makes it even harder to do an analysis of stock prices. This paper proposes an innovative approach on how to use two machine learning algorithms, recurrent neural network (RNN) using LSTM and random forest (RF) to predict stock prices and comparing the performance of these two models using data from the last fifteen years. The combination of RNN and LSTM offers a good solution for processing sequential data, provides improved long-term memory which selectively stores or discards information. Random Forest builds multiple decision trees and combines their output to make predictions. We use multiple parameters to train the model. Mean Absolute Error, Root Mean Squared Error and Mean Square Error are used for evaluating the performance of the model. Graphs are used to visualize and get a better idea of our results by using training and testing sets.

I. INTRODUCTION

Web services are applications that facilitate communication between devices over the internet, using a standardised XML format for information exchange. However, as web services involve exchanging data over the internet, they are vulnerable to various security risks, with identity verification being a major challenge. Authentication is a common method for verifying identity, but SQL injection attacks can bypass authentication by injecting malicious code into input fields. This can grant attackers illegitimate access to the application. This project aims to address these security issues.

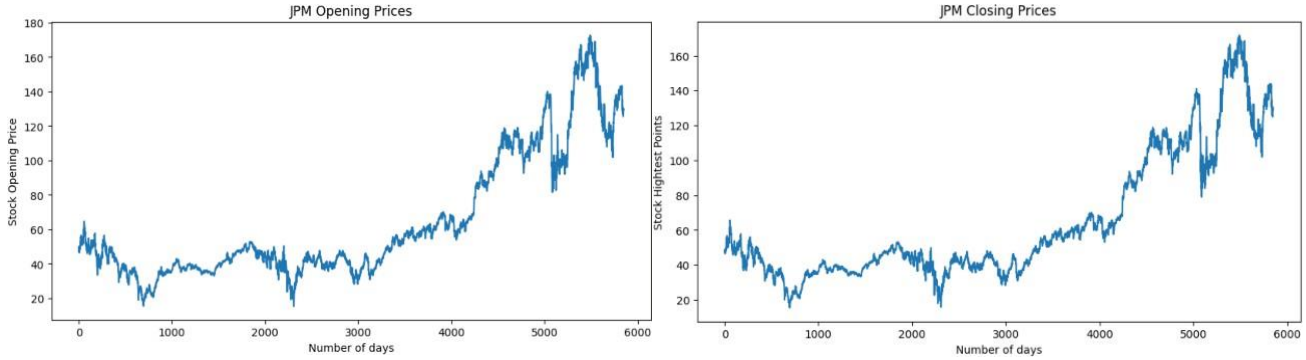
There have been some remarkable advancements in the field of science and technology in recent years. This has led to significant transformations in the business landscape, and the financial markets are crucial in all this. As of 2021, the total market capitalization of all US listed stocks was around $50 trillion. The stock market, also known as the equity market, is a financial market where publicly traded companies' issues and trade stocks to raise capital from investors. The stock market allows investors to buy and sell ownership in these companies and profit from their success. Using historical data to predict stock values through a model can be highly profitable, providing valuable insights and informing investment decisions.

With the help of this study, we want to predict the future trends of stock market prices using two different approaches, namely RNN and Random Forest. LSTM (Long Short-Term Memory) is a type of Recurrent Neural Network architecture that is trained to deal with sequential data, such as time-series data. In RNNs, each input is processed in conjunction with information from previous input, and the weights and parameters are adjusted during the training process in order to improve accuracy of the model's prediction. On the other hand, Random Forest is a machine learning algorithm that uses several decision trees to make predictions. This approach involves training these trees on a random subset of input data. These individual predictions are then combined to generate the final prediction.

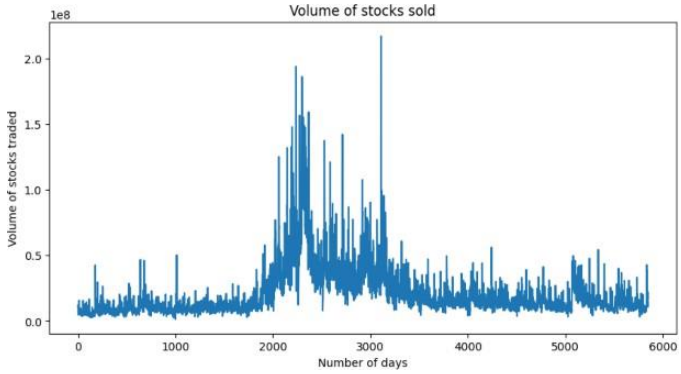
Our model is constructed using python and pyspark to make predictions efficiently. This research will contribute to the stock market's ability to predict stock prices more accurately. We have also given the results of our study and our different approaches. This will also be helpful in our goal. In further sections we provide existing literature on the stock market and prior methods used for predicting stock prices. In Section III, we discuss the machine learning algorithms, LSTM with RNN, Time Series Analysis, Random Forest and graph-based approach. Section IV contains information about our dataset and preprocessing strategies, while Section V discusses our Results and Analysis, where we talk about the results of our experiments. Finally, Section VI is Conclusion and future, which concludes our research and talks about future scope.

II. DATASET AND PREPROCESSING

The dataset taken from Yahoo Finance API contains stock market data for JPMorgan Chase & Co. (JPM) for the period of January 2000 to present. The dataset includes several features such as the opening, closing, highest, and lowest prices of the stock, as well as the volume of shares traded for each day. We can observe the trends across these features through the plots below:



Our final goal is to predict the closing price, so we are only using the 'Close' column from the dataset and creating a new dataframe. Additionally, the dataframe may contain missing/null values, for which we performed check to conclude that there were no null values in the dataset. We used the Pandas isNull() function to check for null values. Now, in order to use the data for LSTM, we perform normalization for which we use the vectorAssembler and MinMaxScalar and return a normalized 'Close' column after passing it though a User-Defined Function (UDF) to convert from vector to scalar value.



Next, we need the data to be in a specific sequence as LSTMs are sensitive to the order of input sequence. We create the input sequence of length 20 and output sequence of length 1 using rdd.flatMap(). Now, as part of the last step we have defined a function that will split the input data into training and testing sets.

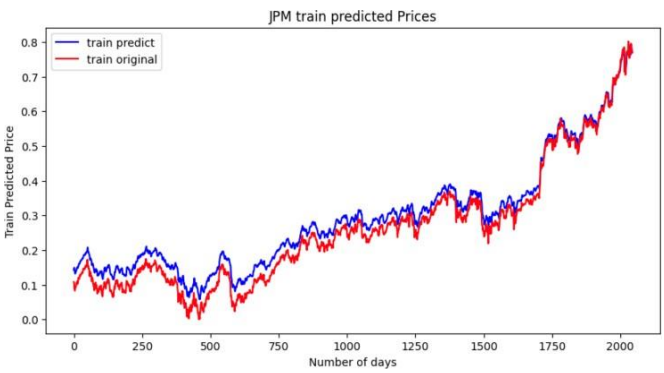
III. RESULT ANALYSIS

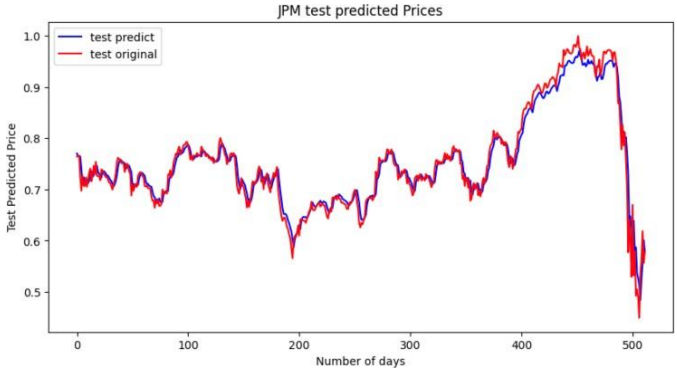
A. *Error Functions:* We are evaluating our model's performance by calculating the Root Mean Square Error (RMSE), Mean Squared Error (MSE), and the Mean Absolute Error (MAE).

B. *Number of Epochs:* The number of epochs determines the number of times the entire dataset is processed during the training of LSTM model. Although increasing the epoch size helps the model learn better and improves the accuracy, it may also lead to overfitting and impact the performance which may result in longer training time and increase computational costs.

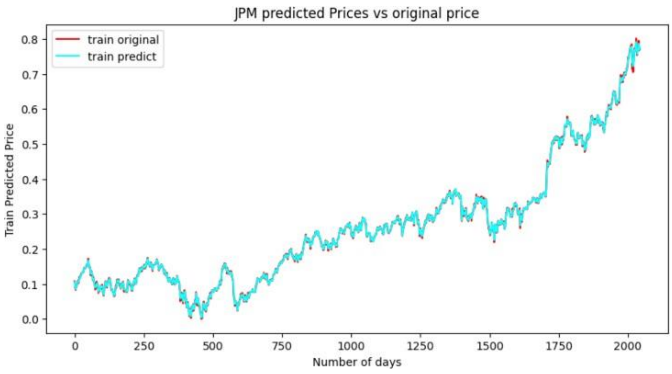
C. *Input Dimensions:* This determines the shape of the input for LSTM and the number of features used in the model's input sequence.

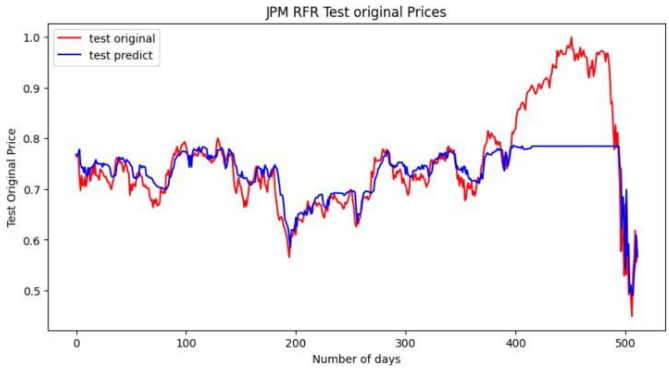
D. *Hidden Dimensions:* Hidden dimension determines the size of internal memory cell and number of neurons in an LSTM cell. Increasing this hyper parameter improves the model's ability to learn complex patterns in the data but also has the risk of overfitting when we are using a small dataset to train the model.





We have used RandomForestRegressor from sklearn to compare our results obtained using our LSTM approach. We passed the parameters n_estimators = 100 and random state as 0 to the RandomForestRegressor. Based on these, the following results obtained were:





IV. Conclusion and Directions

Through this project, we were able to successfully predict the stock market prices for the selected stock across its various trends throughout the years. As we have selected a sufficiently bigger time period, since there is a large amount of data available, we are able to predict the data accurately. The data availability helps in navigating through the erratic fluctuations in stock prices. While comparing our results both algorithms are very close in prediction. With minor details considered LSTM performed better compared to Random Forest Regressor for the mentioned parameters and dataset. However, for time-series models, we cannot always have a model that can perform with the best accuracy; it all depends on the parameters (learning rate, no. of epochs, etc.). One possible future scope could be exploring the effectiveness of ensemble methods, such as combining the predictions of multiple models, to improve the accuracy of the stock market price predictions.