Stock Market Analysis

Mrudul Jambhulkar
Department of Electrical Engineering
IIT Bombay
Mumbai, Maharashtra , India
21d070044@iitb.ac.in.com

Siva Madhav Chinta
Department of Electrical Engineering
IIT Bombay
Mumbai, Maharashtra , India
21d070020@iitb.ac.in.com

Aditya Anand
Department of Electrical Engineering
IIT Bombay
Mumbai, Maharashtra , India
21d070007@iitb.ac.in.com

***Abstract*—Stock market is the public market used to buy and sell shares of publicly listed companies . Machine learning and deep learning models can help in predicting the future prices of stocks and estimate the profit or loss of an individual investing in stocks . This paper discusses 3 machine learning techniques - Support Vector Machine (SVM) , Long Short Term Memory Network(LSTM) and Random Forest regression for stock market analysis .**

***Index Terms*—Long Short Term, memory Network(LSTM), random forest, stock market , support vector machine (SVM)**

I. Introduction

In this project, our aim is to predict changes in stock market prices using three machine learning models: Random Forest Regressor, Support Vector Regressor, and LSTM. We kickstart the process by downloading a dataset comprising 5000 five-minute candles, which serves as the raw material for training our models.

The primary objective here is to identify the most effective model for this dataset. By rigorously testing and comparing the performance of each model, our goal is to pinpoint the one that yields the highest profitability when deployed for prediction tasks. This means finding the model that can accurately anticipate price changes and maximize potential profits.

Once we've determined the optimal model, the next step is to leverage it to estimate the expected gain or loss for each of the NIFTY50 stocks. Armed with this predictive insight, investors can make more informed decisions regarding when to buy or sell stocks within the index.

This approach offers a significant advantage over traditional methods, such as manually analyzing price charts for all 50 stocks. By automating the prediction process using machine learning models, we streamline the decision-making process and save valuable time that would otherwise be spent on laborious analysis.

II. LSTM

A. Introduction to LSTM model

LSTM is one of the most suitable model for time series prediction like stock data . LTSMs are a type of Recurrent Neural Network architectures for learning data involving long-term dependencies [1].
There are 3 steps in LSTM :

- First , decide which information to be deleted from the cell in the particular time step. Sigmoid function is used for this step. It looks at the previous state and the current input and computes the function.
- 2 functions are used in the second layer. The first is the sigmoid function, and the second is the tanh function. The sigmoid function decides which values to should pass through (0 or 1). The tanh function assigns the weights to the values passed and determines their level of importance from -1 to 1 .
- The third step is to determine the final output. First, apply a sigmoid layer to determine what parts of the cell state make it to the output. Then, you must put the cell state through the tanh function to scale the values between -1 and 1 and multiply it by the output of the sigmoid gate.

B. Model Architecture

The model consists of three LSTM layers, each consisting of 50 units . To prevent overfitting, dropout layers with a dropout rate of 0.2 are used after each LSTM layer. The 3 LSTM layers are followed by a final Dense layer with one unit is for predicting the target. The model is compiled using the Adam optimizer and Mean Squared Error (MSE) loss function, which are suitable for regression tasks like stock price prediction. During training, a subset of the training data is utilized with a batch size of 32, and the data is used as validation data to test accuracy of the model.

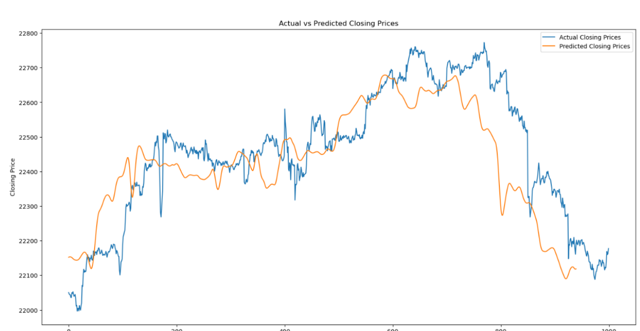

Fig. 1. Predicted closing prices of stock vs actual closing price for NIFTY using LSTM

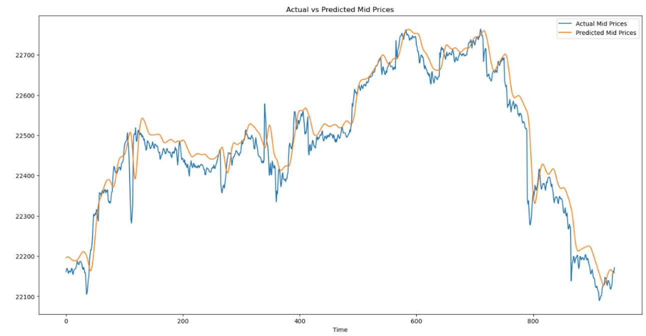

Fig. 2. Predicted average price of stock vs actual average price for NIFTY stock using LSTM

III. SVM

A. Introduction to SVM model

Support Vector Machine (SVM) is a supervised learning algorithm commonly used for classification and regression tasks. In the context of stock market analysis, SVM can be employed to predict the changes in stock prices based on historical data.

[2]The basic idea behind SVM is to find the optimal hyperplane that separates different classes in the feature space while maximizing the margin between the classes. In regression tasks like stock price prediction, SVM aims to find the hyperplane that best fits the training data points with the maximum margin, while still allowing some points to violate the margin.

B. Model Training and Hyperparameter Tuning

In this project, we used the Support Vector Regressor (SVR) variant of SVM for predicting stock price changes. The SVR model was trained using historical stock data, with features extracted from the price and volume information.

The hyperparameters of the SVR model, such as the regularization parameter C and the choice of kernel function, were optimized using grid search with cross-validation. This approach helps in finding the optimal combination of hyperparameters that minimizes the prediction error on the validation set.

C. Prediction and Trading Strategy

Once the SVR model is trained and optimized, it can be used to predict the future price changes of stocks. In our trading strategy, we iterated over a range of days and made predictions for each day based on the SVR model.

Using the predicted price changes, we implemented a trading strategy to identify the top stocks with the highest predicted gains. We considered factors such as the predicted gain percentage, actual gain percentage, and current balance to determine the quantity of stocks to buy or sell.

D. Results and Performance

The performance of the SVM model was evaluated based on the profitability of the trading strategy. By trading the top stocks with the highest predicted gains, we calculated the profit obtained for each iteration.

The cumulative profit obtained over multiple iterations provides insights into the effectiveness of the SVM model for stock market analysis and trading.

The best parameters obtained using grid search for predicting NIFTY stock prices were C= 10, gamma = 1.973e-07 .

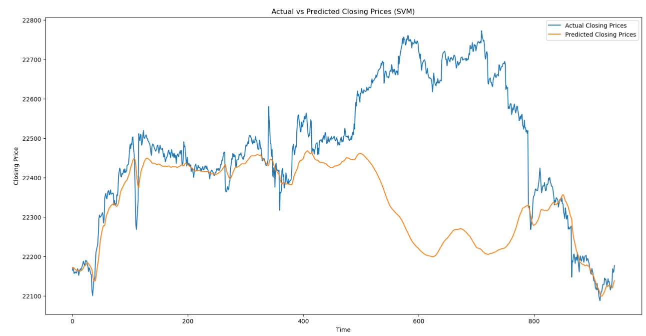

Fig. 3. Actual vs Predicted Closing Prices (SVM) for NIFTY stock

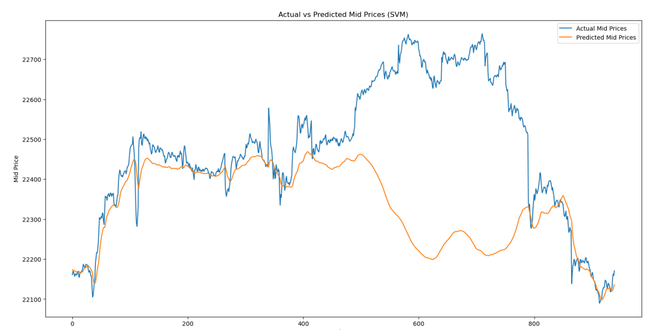

Fig. 4. Actual vs Predicted Mid Prices (SVM) for NIFTY stock

IV. Random Forest

A. Introduction to Random Forest model

Random Forest is an ensemble learning technique based on decision trees, commonly used for classification and regression tasks. In the context of stock market analysis, Random Forest can be employed to predict the changes in stock prices based on historical data.

[2]The Random Forest algorithm builds multiple decision trees during training and combines their predictions to improve accuracy and reduce overfitting. Each decision tree in the forest is trained on a random subset of the training data and features, and the final prediction is determined by aggregating the predictions of individual trees.

B. Model Training and Hyperparameter Tuning

In this project, we utilized the Random Forest Regressor to predict stock price changes. The Random Forest model was trained using historical stock data, with features extracted from the price and volume information.

Hyperparameters such as the number of trees in the forest (nestimators) and the maximum depth of the trees were tuned to optimize the model's performance. In addition, feature shuffling was performed to enhance the generalization capability of the model.

C. Prediction and Trading Strategy

Once the Random Forest model is trained and optimized, it can be used to predict the future price changes of stocks. Similar to the SVM approach, we implemented a trading strategy based on the predicted price changes.

By analyzing the predicted gains and actual gains, along with other factors such as current balance and stock prices, we devised a trading strategy to identify the top stocks with the highest predicted gains.

D. Results and Performance

The performance of the Random Forest model was evaluated based on the profitability of the trading strategy. By trading the top stocks with the highest predicted gains, we calculated the profit obtained for each iteration.

The cumulative profit obtained over multiple iterations provides insights into the effectiveness of the Random Forest model for stock market analysis and trading.

The best parameters obtained using grid search for predicting NIFTY stock prices were max depth = 10, number of estimators= 100 for predicting closing prices . Best parameters for mid prices were max depth = 100 and number of estimators=100.

Fig. 5. Actual vs Predicted Closing Prices (Random Forest) for NIFTY stock

V. Data and Methodology

A. Trading idea

The stock prices change significantly between the close and open of the market the next day. So the main idea is to exploit this and use a ML model to look at the previous day's movement and then predict how much the stock price might move. We will get the gain predictions of all the NIFTY 50 stocks and trade the 10 stocks with the best predicted gains.

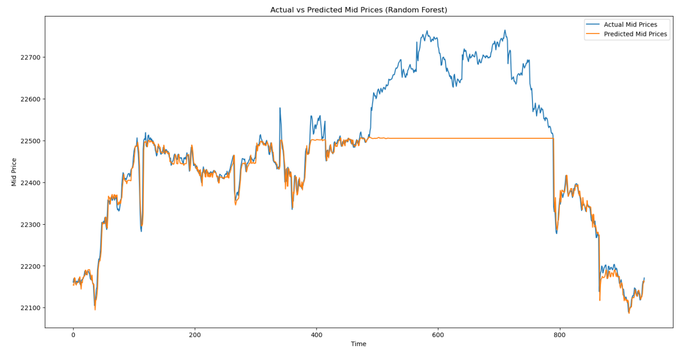

Fig. 6. Actual vs Predicted Mid Prices (Random Forest) for NIFTY stock

B. Data preprocessing

- Got data of the Nifty 50 stocks from Trading view. The data consists of the 5 min chart data for 5000 candles.
- The data needs to be converted to X, y for training and testing. This requires a lot of preprocessing. We added new columns for year, hour and minute using the datetime index of the dataframes.
- Then we found the start and end time indices in the dataframes for each day. The market starts at 9:15 Am and ends at 3:30 Pm. So we will consider this as 1 trading day.
- We consider that we will look at the data from 9:25Am to 3:!5 Pm of that day and predict the percentage change from 3:15 PM of that day to 9:45Am of the next day, i.e. our X will be the data from 9:25AM to 3:15 Pm and our y will be the percentage change in price from 3:15Pm to 9:45 Am of the next day.
- Then we used a MinMaxScaler to normalise the data.

C. Prediction using Model

Now that we have converted the data, its time to predict using the models. We are gonna take the data and apply the models to get the predicted gains for all the NIFTY 50 stocks. Then the gains are sorted in descending order and the top 10 stocks are bought. The actual gains are seen and the profits we get are calculated. For this we considered our capital to be 10000 rupeees.

VI. Observations

Upon analyzing the performance of three models trained on Nifty Fifty data, a few key observations come to light. Firstly, it's clear that the LSTM model comes out on top. It consistently generates the highest profit and demonstrates the most accurate predictions among the trio. Secondly, while not as strong as the LSTM, the Random Forest Regressor still holds its ground, securing the second spot with decent predictive abilities. On the other hand, the Support Vector Regressor disappoints, showing the weakest performance across the board.

Looking at the price prediction plots, it's evident that the LSTM model's predictions closely match the actual trends.

This indicates a solid fit to the data. Conversely, the Random Forest Regressor's predictions show a bit more variability, capturing both signal and noise.

In summary, the LSTM model shines brightest in this analysis, showcasing its prowess in capturing intricate patterns and making reliable predictions. While the Random Forest Regressor holds its own, the Support Vector Regressor falls short. These findings emphasize the importance of choosing the right model for financial forecasting tasks, considering factors such as dataset characteristics and model performance.

REFERENCES

[1] Avijeet Biswal, "Stock Price Prediction using Machine Learning," Simplilearn, [Online]. Available: https://www.simplilearn.com/tutorials/machine-learning-tutorial/stock-price-prediction-using-machine-learning#understanding_long_short_term_memory_network

[2] Chatgpt