**STOCK MARKET PREDICTOR**

In this project I have focused on developing a stock market prediction system using machine learning techniques. This system has a primary objective of forecasting future stock prices based on historical market data. The entire process involves utilizing past stock price movements, along with other relevant factors, to train a predictive model capable of making accurate forecasts. This system aims to assist traders and investors in making informed decisions through the provision of insights into potential future price movements of specific stocks.

**Data Set**

The dataset used in this research is a compilation of historical stock price information from the popular financial data and analytics website Yahoo Finance. When examining the performance of publicly traded companies in the financial markets, this dataset is a useful tool since it offers insights into important variables such as trading volumes and stock price variations over time. The dataset includes a wide range of variables, where each row is the activity of the market for a specific trading day. The following are the primary attributes that are present in the dataset:

**Date:** This column indicates the date of the trading day, providing a chronological timeline of the historical data.

**Open:** The opening price of the stock at the beginning of the trading day. This is the price at which the first trade of the day occurs.

**High:** The highest price reached by the stock during the trading day. This indicates the maximum price level at which the stock was traded during the day.

**Low:** The lowest price reached by the stock during the trading day. This represents the minimum price level at which the stock was traded during the day.

**Close:** The closing price of the stock at the end of the trading day. This is the price at which the last trade of the day occurs.

**Adj Close:** The adjusted closing price, which accounts for any corporate actions such as stock splits or dividends. This adjusted price reflects the true value of the stock, accounting for changes in the stock's capital structure.

**Volume:** The trading volume, representing the total number of shares traded during the day. This metric provides insights into the level of market activity and investor sentiment for the stock.

The Python yfinance package makes the data easier to get by offering a user-friendly interface for getting historical stock price data from Yahoo Finance. Users have the option to designate the preferred timeframe for the data as well as the stock symbol (ticker) of the company they are interested in. This facilitates data access and analysis for a broad spectrum of publicly traded corporations across many sectors and industries.

The dataset is freely available to the public and can be retrieved programmatically through Yahoo Finance's APIs or through third-party libraries like yfinance, or straight from the Yahoo Finance website. It is a well-liked option for study, analysis, and predictive modeling in the fields of finance and machine learning because of its accessibility and comprehensiveness.

Source link: <https://finance.yahoo.com/quote/BTC-USD?.tsrc=fin-srch>

**System Description**

The project's primary process entails utilizing machine learning techniques to develop a stock market forecast system. The main concept is to build a predictive model that can predict future stock values using historical stock price data. Below is a thorough explanation of the system's primary steps:

**Data Retrieval:** The yfinance module in Python is used to retrieve historical stock price data from Yahoo Finance at this time. Users enter the chosen time period and the stock symbol of the company they want to investigate.

**Data Processing:** Preprocessing is done on the obtained data, which involves scaling and dividing it into training and testing sets.

**Model Development:** The Keras library loads a neural network model that has already been trained. Using previous stock price data, the model is trained to identify trends and correlations that may be used to forecast future prices.

**Prediction:** Using the test data, the trained model predicts stock prices; the model's performance is assessed by comparing the anticipated and actual prices.

**Visualization:** To give consumers a clear understanding of the anticipated fluctuations in stock prices, the system creates visualizations such as moving averages, stock price trends, and a comparison of actual and predicted values.

Some of the project's primary libraries include:

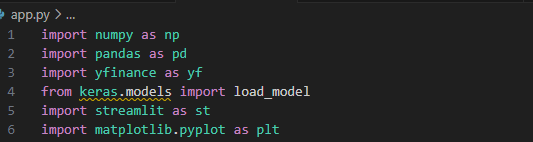
**numpy and pandas:** These are libraries used for preparing and manipulating data. They divide data into training and testing groups and scale the data.

**yfinance:** Yahoo Finance's historical stock price data may be retrieved using this library.

**Keras:** The pre-trained neural network model, which is developed using the Sequential model API of Keras, is loaded using this library.

**Matplotlib:** It is a package used for data visualization, which includes the plotting of moving averages, trends in stock prices, and comparisons of real and expected values.

**Streamlit:** This library is used to build the web application's user interface, which enables users to enter a stock symbol and view the stock market prediction system's output.



**Results and Conclusion**

A number of criteria can be used to assess the stock market prediction system's performance, such as how well the system works overall and how accurate the predictions are in comparison to actual stock prices. When evaluating the system's performance, take into account the following factors:

**Prediction Accuracy:** The system's prediction accuracy serves as a performance indicator. Metrics that measure the variation between the expected and actual stock prices, including mean squared error (MSE) and mean absolute error (MAE), can be used to assess this. Greater forecast accuracy is shown by lower values of these indicators.

**Comparison with Baseline Models:** Comparing the system's performance to other forecasting methods or baseline models is another approach to assess its performance. To gauge the system's efficacy, one way to compare its predictions is to compare them to a basic baseline model, like a moving average or random walk.

**Back testing and Simulation:** These two methods can be used to evaluate how well the system performs in an actual trading environment. This entails simulating trades based on the system's predictions using historical data, then tracking the profitability of these trades over time.



**Conclusion**

The accuracy of the stock market prediction system's forecasts, comparison to baseline models, back testing outcomes, user input, and generalizability can all be used to assess its success. We may have a thorough grasp of the system's efficacy and draw well-informed conclusions regarding its stock price prediction performance by taking these elements into account.