JPX Tokyo Stock Exchange Prediction

Capstone Spring 2023

Team 1: Saima Sarowar

Background

- Stock market prediction is the act of forecasting the future performance of stocks and other financial instruments traded on stock exchanges.

- The history of stock exchange prediction dates back to the early 20th century, with the emergence of technical analysis as a means of predicting future stock prices based on past market data.

- In recent years, the rise of big data, artificial intelligence, and machine learning has brought about new methods of stock market prediction, such as algorithmic trading and sentiment analysis.

Background

- The accuracy of stock market prediction has always been a topic of debate, with some experts claiming that it is impossible to predict stock prices with any degree of certainty, while others argue that with the right tools and methods, it is possible to make informed predictions.

- Despite the challenges, stock market prediction remains an important area of research and practice, as it can have a significant impact on investment decisions, portfolio management, and financial stability.

Problem?

- Quantitative trading requires historical and real-time data which is difficult for retail investors to obtain.

- Stock market prediction is complex due to many variables, randomness, limited data, and human behavior.

Importance

- Predicting stock prices can drive financial gains, mitigate risks, stabilize the economy, and advance technological developments.
- Accurate predictions can help investors make better decisions, manage risks, and prevent losses.
- It can help policymakers and economists make better decisions and take actions to prevent economic instability.

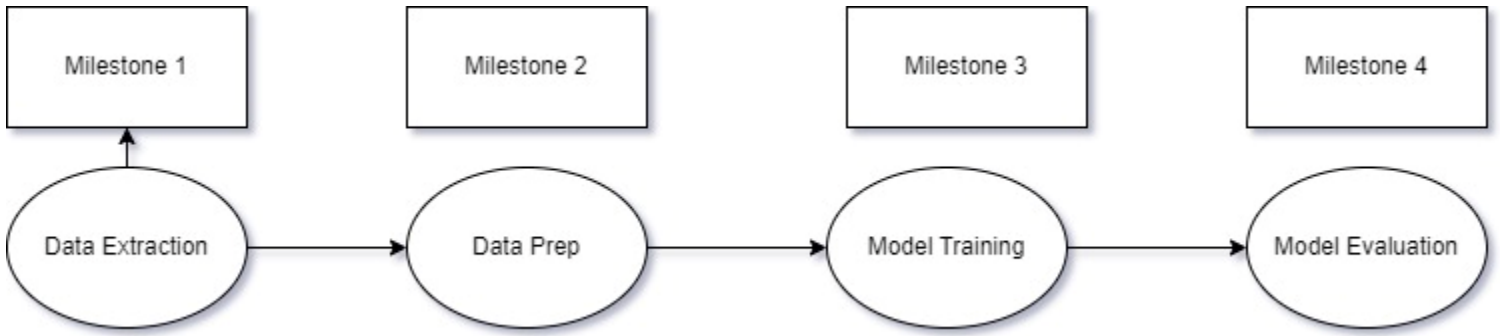
Plan

- Predict the direction of the stock exchange using datasets provided by Kaggle.
- Choose a machine learning algorithm (linear regression, decision trees, random forests, or neural networks) and split data into training and validation sets.
- Train and validate the model, fine-tune it, and test its performance.

Performance Metric

- Mean Squared Error (MSE): measures the average squared difference between predicted and actual values.
- Root Mean Squared Error (RMSE): like MSE, but with the square root taken to make the units of error the same as the original values.
- Mean Absolute Error (MAE): measures the average absolute difference between predicted and actual values.
- Directional Accuracy (DA): measures the percentage of correct predictions in terms of the direction of market movement (up or down).

System Architecture



Milestone 1- Data Extraction & Preparation

- Kaggle account was created, and data was downloaded after agreeing to the competition rules
- Data was extracted from a zip file and was organized into four folders and 1 file.

Milestone 1- Data Extraction & Preparation

- Data Specifications
  - The stock_list.csv file contains over 4500 lists of stocks with information on their products, trade date, sectors, and market capitalization.
  - The data_specifications folder contains 5 csv files that give out the specifications of the data
  - The files include options_spec, stock_fin_spec, stock_list_spec, stock_price_spec, and trades_spec
  - These files provide detailed information on the data and its structure
  - The example_test_files folder contains 6 csv files with examples of how to use the data for testing

Milestone 1- Data Extraction & Preparation

- Data Specifications
  - These files demonstrate how to work with the data and what to expect when using it
  - The supplemental_files folder contains additional data with other variables of the stock list
  - This includes financials, options, stock prices, trades, and secondary_stock_prices
  - Each file has around 100,000-1 million records
  - The train_files folder is the same as the supplemental_files folder but has fewer records

Milestone 1- Data Extraction & Preparation

- Data was cleaned by creating conditions in an excel sheet and removing any records with missing information
- Data was split from supplemental files, with around 70% for training, 15% for testing, and 15% for validation
- The excel files were converted to .csv and stored in Github Gist for easy access

Milestone 2

- Algorithm LSTM, a type of RNN well-suited for processing sequential data.
- Model Architecture is Multiple LSTM layers, fully connected layers for feature extraction, and an output layer
- LSTM Layer contains memory cells that retain information over time and gates that control the flow of information
- Output Layer produces a single value representing the predicted stock price

Milestone 2

- Data Imported into Jupiter Notebook and cleaned to remove missing values
- Data Split: 80% for training and 20% for testing from supplemental files
- Example test files used for validation
- Data Split into Input and Output Variables for model training

Milestone 3

- We scaled and reshaped the stock_prices files for training the LSTM model.
- The dataset was loaded and the lookback parameter was defined.
- Input and output data were extracted and normalized using StandardScaler.
- The input data was then reshaped to a 3D tensor.

Milestone 3

- The LSTM model was built using Keras Sequential API.
- The model consisted of a single LSTM layer with 50 memory cells, followed by a Dense output layer with one unit.
- The model was compiled using the mean squared error loss function and the Adam optimizer.
- The model was trained using the fit() method, with a batch size of 64 and a validation split of 20%.
- After training, the model was used to make predictions on new data using the predict() method.

Milestone 4

- Used Mean Squared Error (MSE) for evaluating the performance of the LSTM model
- MSE measures the average squared difference between the predicted and actual values
- Lower values of MSE indicate better performance of the model
- Visualized the predictions against actual values using a line plot
- The line plot helps in identifying the accuracy of the model by comparing the predicted values with actual values
- The plot shows the trends of predicted and actual stock prices over time.

Conclusion

- Explored the use of Long Short-Term Memory (LSTM) algorithm for stock price prediction in the JPX stock exchange
- Collected and preprocessed the data, followed by splitting it into training and testing sets
- Built an LSTM model using Keras Sequential API and trained it on the training set
- Evaluated the model using the testing set
- Demonstrated the potential of LSTM algorithm in stock price prediction
- Further research and experimentation needed to improve the model's performance and to explore other algorithmic approaches to stock price prediction.

Thank You