Stock Prediction Models Comparison

Abstract

This document compares three stock prediction models—Random Forest Regressor, Gradient Boosting Regressor, and LSTM applied to a historical stock price dataset. The evaluation includes performance metrics, computational resources, interpretability, and generalization. Insights from the comparison aim to guide the selection of the most suitable model for stock prediction.

Dataset Description

A historical stock price dataset was collected for several stocks/companies, including 'PG', 'META', 'AMD', 'NFLX', 'TSM', 'AAPL', 'GOOGL', 'MSFT', 'AMZN', 'TSLA'. The dataset contains various features, including date, close, open, high, low, volume, and adjusted close.

3. Methodology

­­The process for applying the four techniques was identical to be able to evaluate their performance against each other and pick one model for our research.

All four implementations followed this structure:

Data preprocessing:

* Handle Missing Values: Identify and handle any missing values in the dataset.
* Remove Duplicates: Check for and remove any duplicate records in the dataset.
* Feature Engineering: Generate additional features such as lag variables, moving averages, and technical indicators.
* Scaling: Apply Min-Max scaling to normalize numerical features.
* Split the dataset using TimeSeriesSplit for training and testing the models.

Model Implementation

* Initialize and fit the model e.g. RandomForestRegressor to the training data.
* Predictions on the test set
* Evaluate Performance metrics(MAE, MSE, RMSE, R-squared)
* Visualize predicted vs true values.
* All models followed the same steps except LSTM which needed extra training of the model on the training set.

4. Evaluation Metrics

The evaluation metrics for comparing the models will be:

accuracy: the ratio of correctly predicted instances to the total instances.

precision: the ratio of correctly predicted positive instances to the total predicted positive.

recall: the ratio of correctly predicted positives to the total observations in the class.

F1-score: the weighted average of precision and recall.

Confusion Matrix: a table illustrating the counts of true positives, true negatives, false positives, and false negatives.

5. Model Performance

5.1 Gradient Boosting Regressor

A screen shot of a graph

Description automatically generated

5.2 Random Forest Regressor

A screen shot of a graph

Description automatically generated

5.3 Long-Short-Term Memory (LSTM)

A screen shot of a graph

Description automatically generated

6. Analysis and Conclusion

Performance of each model when applied to the full dataset.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Model | MAE | MSE | RMSE | R-Squared |
| Gradient Boosting | 3.483 | 18.46 | 4.296 | 0.615 |
| Random Forest | 3.279 | 17.334 | 4.16 | 0.638 |
| LSTM | 0.309 | 0.1415 | 0.376 | 0.997 |

Factors for Choosing LSTM

The decision to choose LSTM as the preferred model is based on various factors:

* The LSTM model outperforms both Gradient Boosting and Random Forest significantly in terms of MAE, MSE, and RMSE. These metrics show the accuracy of the predictions, and lower values indicate better performance.
* The LSTM model achieves an extremely high R-squared score of 0.997, suggesting that the LSTM model explains almost all the variability in stock prices, making it a highly accurate predictor.
* LSTM is designed to capture long-term dependencies in sequential data, which aligns well with the time series nature of stock prices.
* LSTM is specifically designed for handling sequential data. It considers the historical sequence of stock prices, which is crucial for predicting future values in time series tasks.