UNIVERSITY OF EDINBURGH
Business School

2022-23

CMSE11475 FINANCIAL MACHINE LEARNING I

FORECASTING STOCK VOLATILITY

B213323

Word count:

1,542

Abstract

This project investigates the effectiveness of LSTM and Autoencoder models in forecasting stock return volatility and compares the forecasting performance of different stocks. The study uses daily adjusted prices of Amazon (AMZN) stock from January 2008 to December 2022, with seven features, including open, high, low, close, adjusted close, return, and volatility. The project aims to predict stock return volatility for the next 7, 14, 21, and 28 days.

The LSTM model with sliding windows of 7, 14, 21, and 28 days is used to predict stock return volatility. The study applies 10-fold Timeseries Split cross-validation to evaluate the LSTM model's performance, and the Root Mean Squared Error (RMSE) is used as the evaluation metric. In addition, the project integrates autoencoders to extract latent features from the data to improve the LSTM model's performance.

The results show that the combinations of a window of 14 days in the past and seven days into the future yield the lowest RMSE scores. Moreover, the study finds that an increase in the amount of historical data used for predictions leads to improved RMSE.

Incorporating autoencoder into the framework yields promising results, with the best performance observed when using five and six latent dimensions. The study also shows that concatenating the original and extracted features as input to the LSTM model improves forecasting prediction.

Additionally, the cross-validation results show that the LSTM model performs well on AMZN, AAPL, MSFT, GOOGL, and NVDA stocks, with low RMSE scores observed in all folds. The study provides insights into the effectiveness of LSTM in forecasting stock return volatility and using autoencoders to extract latent features to improve model performance.

Finally, compared with other models, especially ensemble methods like XGBoost and Random Forests Regressor, LSTM yields better RMSE scores than those methods.

Introduction

This project investigates the feasibility of using LSTM and Autoencoder to forecast stock return volatility and compares the forecasting performance of different stocks. The models' input data is the selected stocks downloaded from Yahoo Finance. The output is the stock return volatility, calculated as the past 30-day arithmetic return standard deviation.

The LSTM model will be explored in this project. The models will be trained on historical data and evaluated based on their ability to predict the volatility of future returns. The performance of the models will be compared using root means squared error (RMSE).

The potential outcomes of this project can provide insights into the effectiveness of LSTM in forecasting stock return volatility. The project can also reveal which window and horizon are the best regarding forecasting accuracy.

Data

The data used in this project was downloaded from Yahoo Finance for AMZN from January 2008 to December 2022. The data consists of daily adjusted prices of the stocks and has seven columns/features, namely Open, High, Low, Close, Adjusted Close, Return, and Volatility. The Return column represents the percentage change mean for the last 30 days, and the Volatility column represents the standard deviation of the last 30 days of arithmetic return.

The data was split into three sets for training, validation, and testing, with a split ratio of 60%, 20%, and 20%, respectively. The training data fit the LSTM and Autoencoders models, while the validation data was used for hyperparameter tuning and early stopping. The testing data was used to evaluate the performance of the models.

The training data format is an array, with each row representing the features of a single day and the columns representing the different features. The forecasting horizon used in this project is seven days, 14 days, 21 days, and 28 days, which means that the models are trained to predict the volatility of the stock returns for these different time horizons. Given the past data, the forecasting horizon determines how many future days the model should predict the volatility.

Model

In this project, we have selected a Long Short-Term Memory (LSTM) model for forecasting stock return volatility. We expect the LSTM model to perform well in capturing the patterns and trends in the historical data and use them to make accurate predictions of future volatility. We will use the LSTM model with different sliding windows of 7, 14, 21, and 28 days to predict the volatility of stock returns for different forecasting horizons. The sliding windows allow us to capture the temporal dependencies in the data and use them to make predictions.

To evaluate the performance of the LSTM model, we will use 10-fold Timeseries Split cross-validation (TSCV). Finally, we will assess the performance of the LSTM model on the test set using the root mean squared error (RMSE) metric.

In addition to the LSTM model, we incorporate autoencoders into our framework to extract latent features from the data. We will experiment with different numbers of latent features extracted, from 1 to 6, to see how they affect the performance of the LSTM model.

To incorporate the extracted features into the LSTM model, we will concatenate them with the original input features and use them as input to the LSTM model. This will enable the model to learn from the original and extracted features.

Result Analysis

Table 1 RMSE of different window and horizon

No	Look Back Window	Horizon	RMSE
1	7	7	0.000721
2	7	14	0.000723
3	7	21	0.001686
4	7	28	0.001296
5	14	7	0.000552
6	14	14	0.000766
7	14	21	0.001516
8	14	28	0.001017
9	21	7	0.000527
10	21	14	0.000682
11	21	21	0.002791
12	21	28	0.001076
13	28	7	0.000705
14	28	14	0.000848
15	28	21	0.001008
16	28	28	0.001036

Based on the results presented in the table, it can be observed that the combinations involving a window of 14 days in the past and seven days into the future yield the lowest RMSE scores. Notably, most iterations with a look-back window smaller than the forecast horizon exhibit high RMSE values. This implies that an increase in the amount of historical data used for predictions leads to improved RMSE.

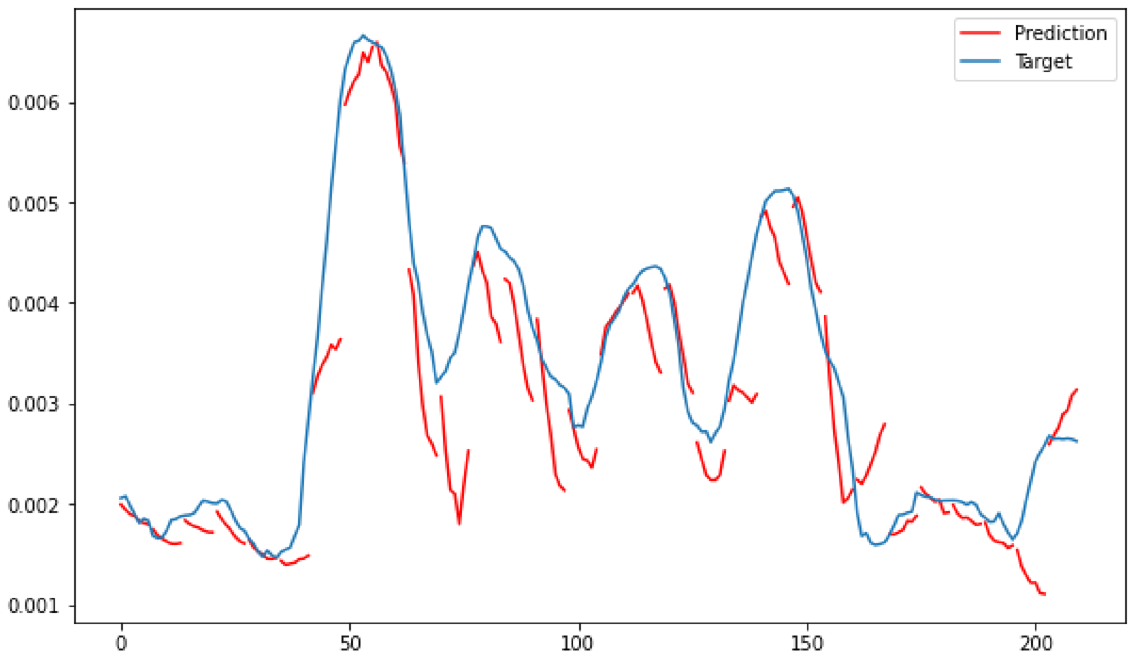

Figure 1 Prediction for window=14, horizon=7, horizon period=30

The objective of extracting features from LSTM is to determine whether using different latent dimensions, ranging from 1 to 6, in a comparable LSTM model will result in an improved prediction outcome, as evaluated by RMSE.

Table 2 Latent dimension size from LSTM Autoencoders and RMSE for prediction using LSTM.

No	Number of Latent Dimension	RMSE
1	1	5.97044
2	2	5.96455
3	3	5.77288
4	4	5.68045
5	5	6.05777
6	6	11.3637

Upon careful examination of the findings in Table 2, we have determined that the optimal number of latent dimensions for our LSTM Autoencoder is three or four latent dimensions. These latent dimensions serve as a compressed representation of the original data, reducing data complexity while minimising the loss of critical information. However, our analyses have indicated that incorporating these latent dimensions from the LSTM Autoencoder failed to yield a significant improvement in the RMSE value of the LSTM Model for this dataset. Figure 2 shows the forecasting result of latent dimension = 3.

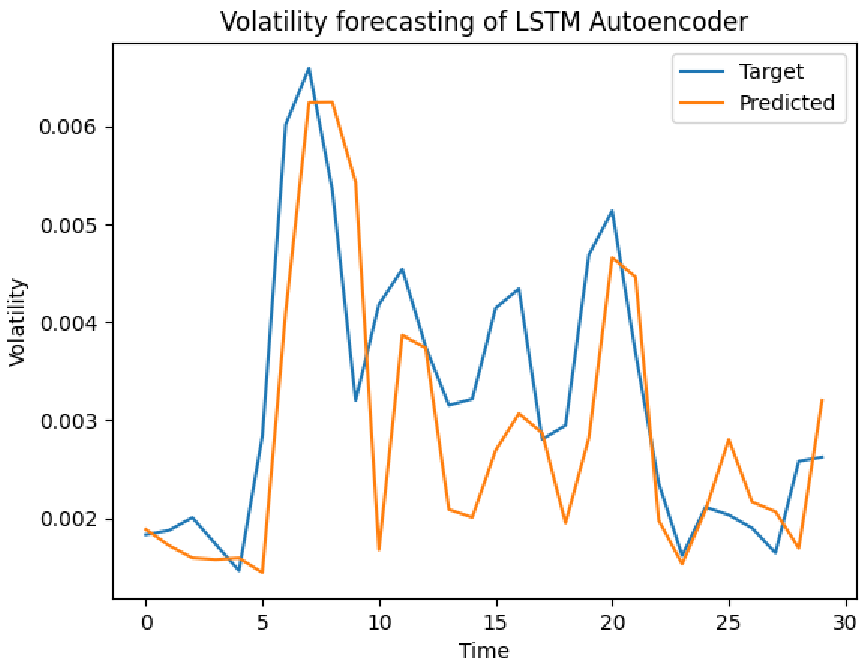

Figure 2 Forecasting result for latent dimension = 3 from LSTM Autoencoder

We also tested the reliability and versatility of our LSTM model by applying it to forecast the volatility of various stocks, namely AAPL, MSFT, GOOGL, and NVDA. The results of our experiments, as presented in Table 3, reveal that the RMSE values for each stock are consistent, indicating that our LSTM model performs consistently well across different stocks.

Table 3 RMSE for other stocks with 10-Folds Cross-Validation Results

No	Fold	AMZN	AAPL	MSFT	GOOGL	NVDA
1	1st	0.002190	0.001078	0.001283	0.001411	0.002536
2	2nd	0.001966	0.001160	0.001297	0.001606	0.003286
3	3rd	0.001213	0.001514	0.000983	0.001197	0.000976
4	4th	0.001478	0.000942	0.000947	0.000993	0.001041
5	5th	0.002326	0.001374	0.001620	0.001560	0.001985
6	6th	0.000904	0.001319	0.000831	0.000909	0.002486
7	7th	0.001337	0.001322	0.000744	0.000902	0.001948
8	8th	0.001538	0.001796	0.001381	0.001427	0.002498
9	9th	0.001463	0.001725	0.001208	0.001472	0.001855
10	10th	0.002650	0.001696	0.001670	0.001424	0.003735
Mean		0.001706	0.001392	0.001196	0.001290	0.002235

We represent the comparison of our LSTM model for five different stocks in Figure 3.

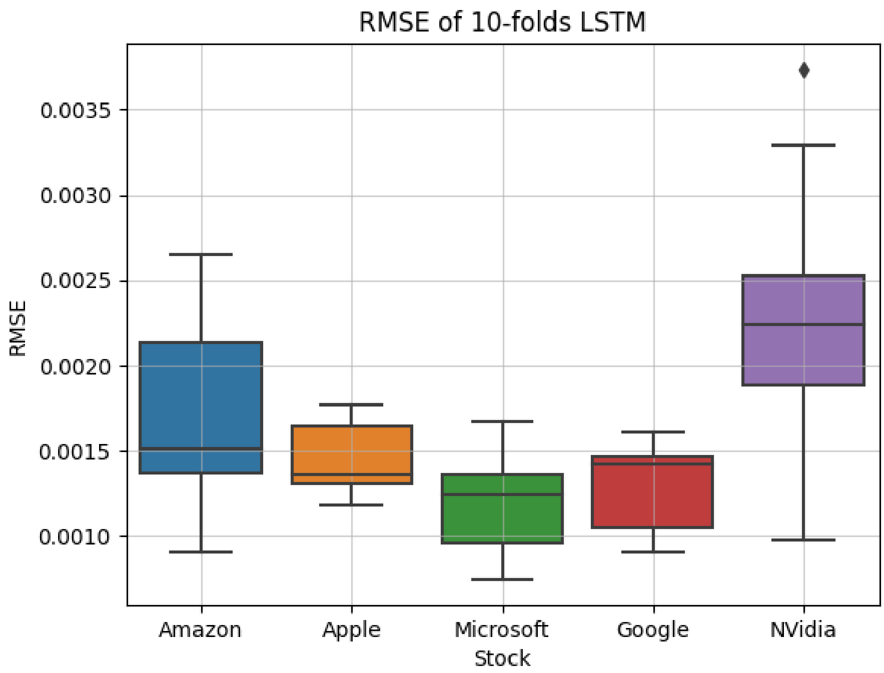

Figure 3 RMSE values for five different stocks

Furthermore, we conducted a thorough analysis comparing the performance of our LSTM Model with various regression models, particularly ensemble models. As illustrated in Figure 4, we compared three distinct models. We observed that the LSTM Model outperformed XGBoost and Random Forests Regressors regarding the stock forecasting task, with a lower median of RMSE.

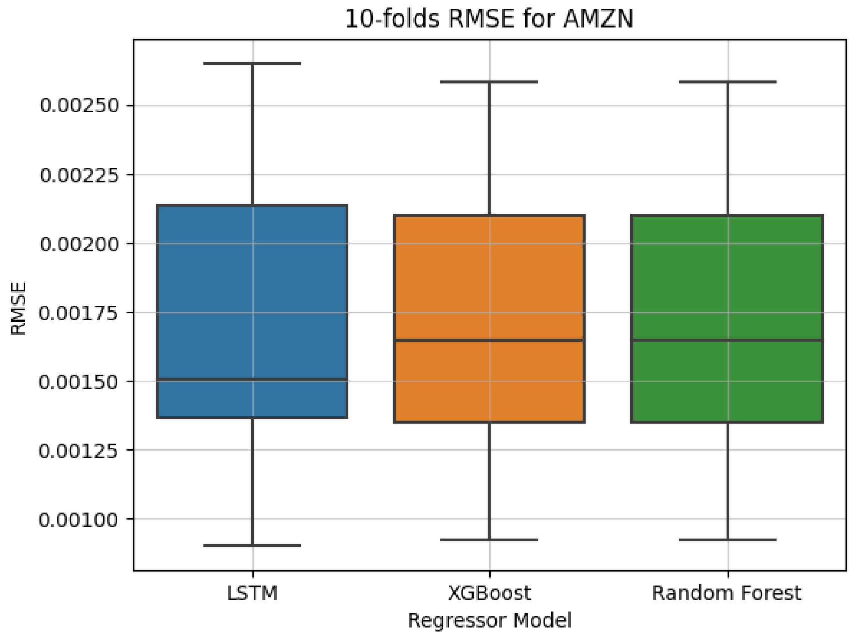

Figure 4 Comparison with other regressors

It is also important to consider the context in which these results were obtained. The models' training and testing period may not represent future market conditions. As such, the models may need to be generalised better to new data. Additionally, other external factors could impact the accuracy of the models, such as changes in market regulations, political events, or global economic conditions.

Overall, the time series LSTM model provided a helpful starting point for evaluating the performance of predictive models for stock price forecasting and outperformed XGBoost and Random Forests Regressors. While the results may not be perfect, they can still provide valuable insights into the underlying patterns and trends in the stock data.