Deep Learning for Artificial Intelligence

FINAL REPORT

Time-series Data And Application To Stock Market

Pham Hoang Lan UG190042

Instructor: Prof. Dang Huynh

Fulbright University Vietnam

May 26th, 2023

TABLE OF CONTENTS

ABBREVIATION

- RNN: Recurrent Neural Network
- LSTM: Long Short-Term Memory
- GRU: Gated Recurrent Unit
- mse: Mean Squared Error
- SMA: Simple Moving Average
- EMA: Exponential Moving Average

I. NASDAQ STOCK PRICE PREDICTION

1. Choice of stock price data

In order to focus on the task of building and comparing the performance of various RNN models on the dataset, the decision was made to utilize only the stock price data of Microsoft Corporation Common Stock (MSFT).

By doing so, the efforts were concentrated on analyzing the stock performance of a single company. However, the inclusion of multiple stock price datasets from different companies will be reserved for task 4, where the aim is to develop an optimized investment strategy for a portfolio.

2. Choice of window size

Two-time window sizes were selected for comparison: 30 days, approximately 1 month of trading days, and 252 days, corresponding to 1 year of trading days. These window sizes will be assessed in terms of model accuracy, with each model trained using an equal number of epochs, batch sizes, and folds in cross-validation. Various aspects of performance will be analyzed and compared across these window sizes.

3. Choice of RNN models

As discussed in class, three RNN models have been specifically designed to handle sequential data or time-series: 1D Convolution, LSTM, and GRU. For this task, the models used are Bidirectional LSTM, GRU, and 1D Convolution.

While Bidirectional LSTM is commonly employed for grammatical data, its performance on historical stock price data presents an intriguing testing opportunity. The multi-layer LSTM model will be utilized in a subsequent stage to process historical stock price data from different companies for portfolio construction.

To ensure a fair comparison, the parameters used in constructing the same type of model for each window size will remain unchanged. This includes parameters such as learning rate, loss function, activation functions, and so on.

GRU has been selected as the model of choice for performing cross-validation on the dataset due to its excellent performance observed during the experiments conducted with the training-validation-test split.

4. Main hyper-parameters

- Number of epochs = 200
- Size of batch = 4096
- Number of folds = 10
- Loss function: mse

5. Results and Discussion

i. Training – Validation – Testing split

- **Bi-LSTM model**

  MSE on the test set

  - Window size in 30 days: 0.034
  - Window size in 252 days: 0.0028

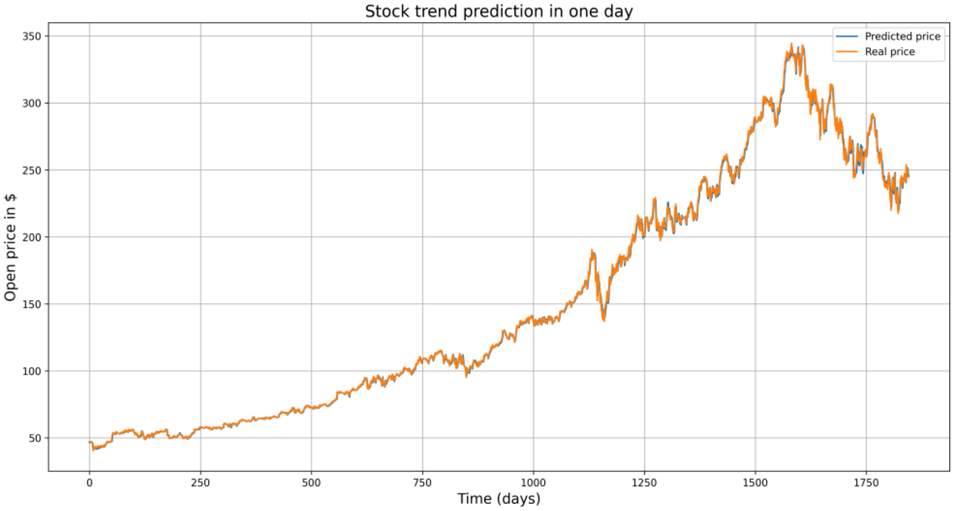


Figure 1. Predicted price from Bi-LSTM model compared with real price with in 30 days

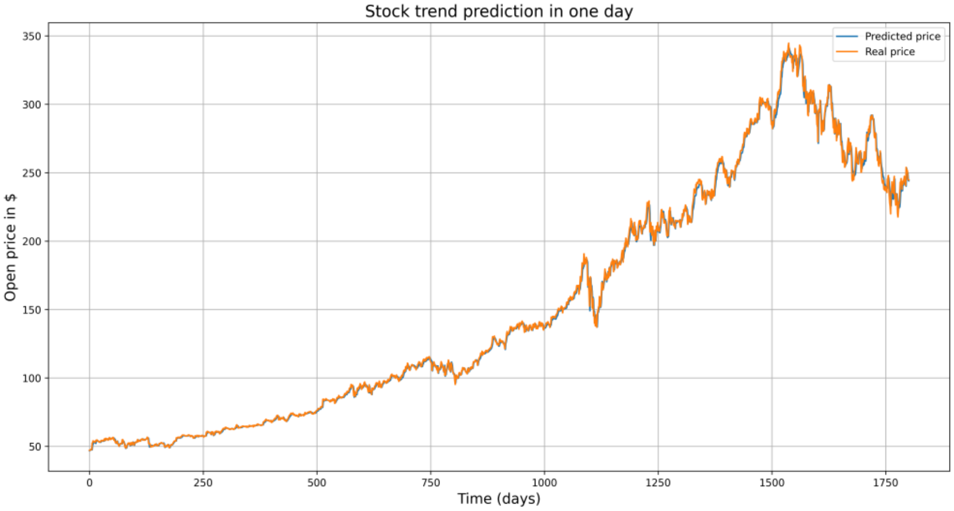


Figure 2. Predicted price from Bi-LSTM model compared with real price with in 252 days

- **GRU model**

MSE on the test set

- Window size in 30 days: 0.0337
- Window size in 252 days: 0.0028

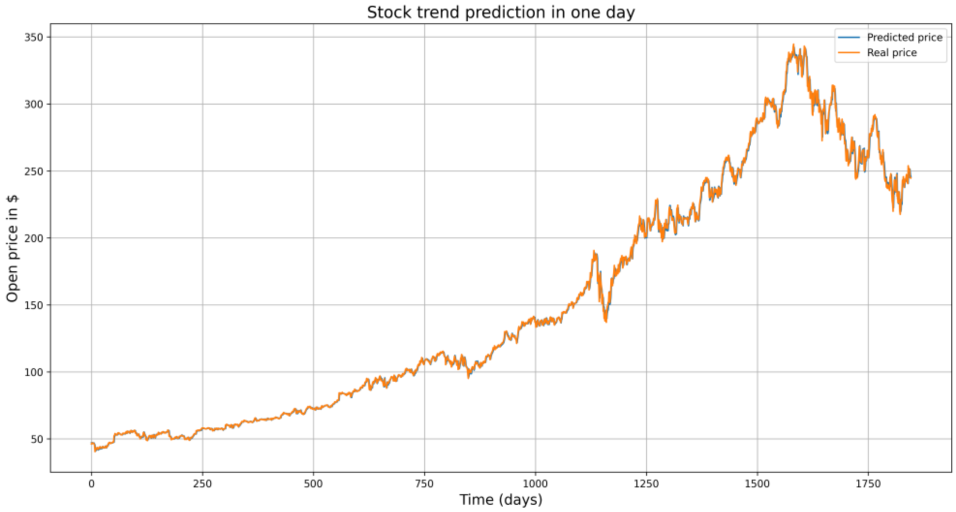


Figure 3. Predicted price from GRU model compared with real price with in 30 days

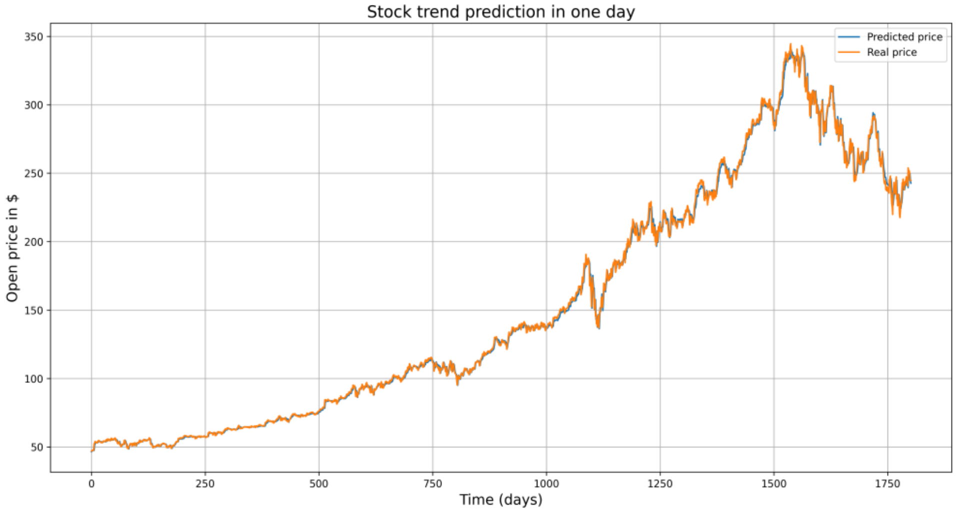


Figure 4. Predicted price from GRU model compared with real price with in 252 days

- **Conv1D model**

  MSE on the test set

  - Window size in 30 days: 0.06
  - Window size in 252 days: 0.035

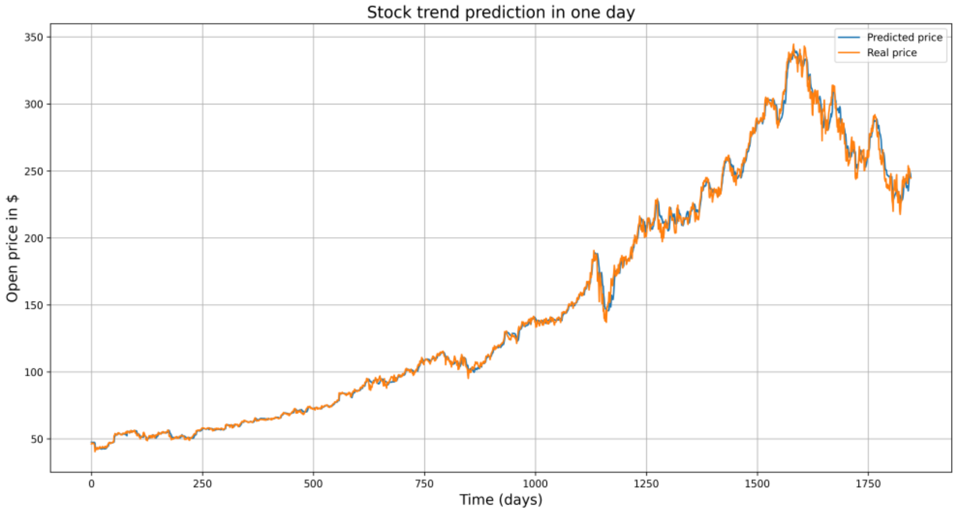


Figure 5. Predicted price from Conv1D model compared with real price with in 30 days

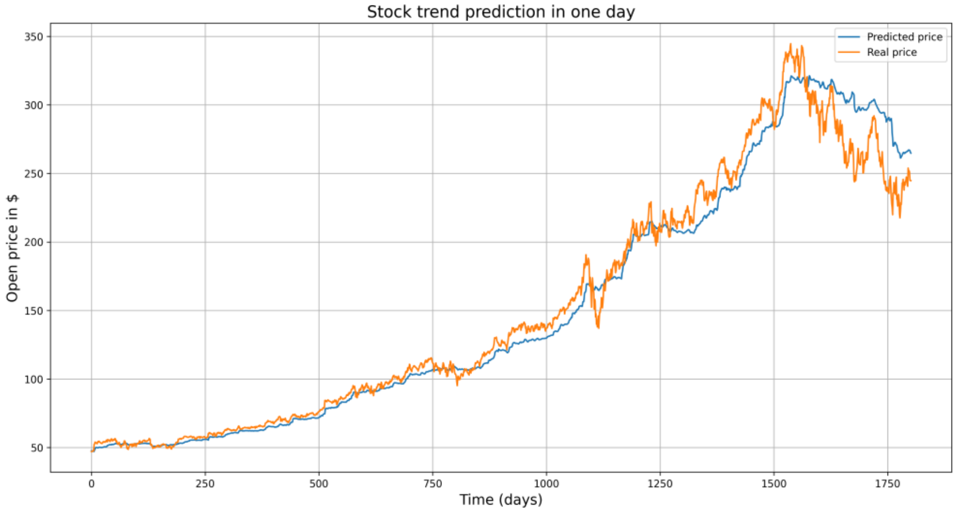


Figure 6. Predicted price from Conv1D model compared with real price with in 252 days

➔ Drawing on the predicted price from RNN models compared with real price and the MSE in two defined window sizes, there are several key findings:

- It is possible to confirm that both experiment window sizes generate good fit with the real price of the company's historical data in Bi-LSTM and GRU models
- Both window sizes deliver good performance and look very fit to the historical data.

ii. Cross validation

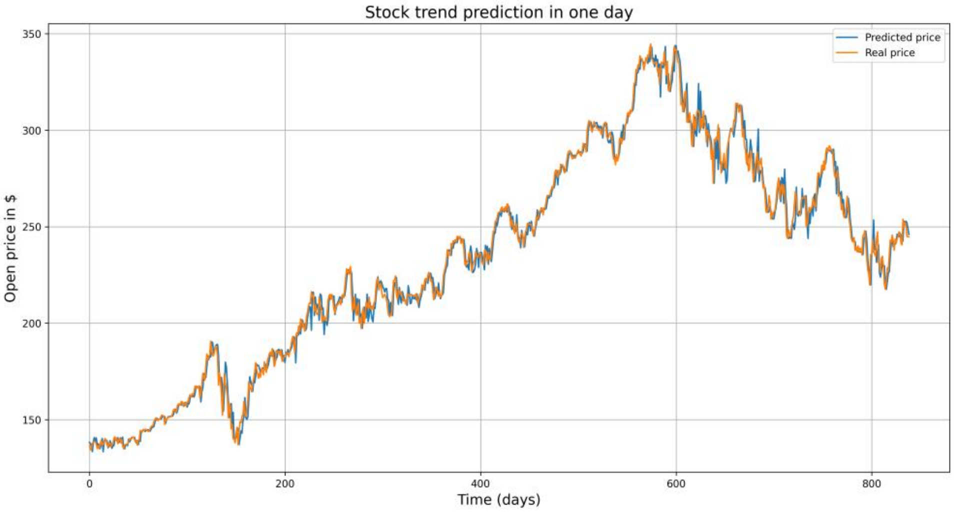


Figure 7. Predicted price from cross validation compared with real price with in 30 days

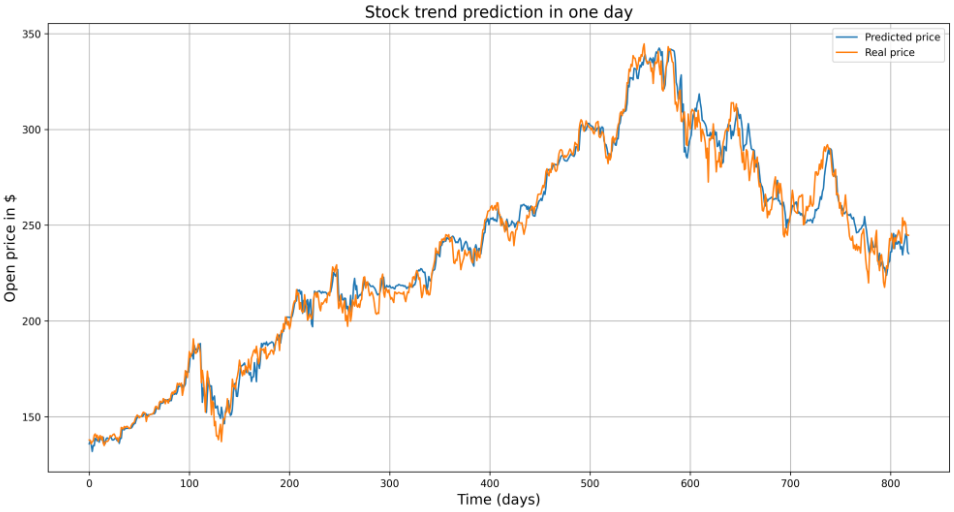


Figure 8. Predicted price from cross validation compared with real price with 252 days

➔ Cross validation works better for small window size

II. VIETNAM STOCK PRICE PREDICTION

1. Choice of stock price data

In order to focus on the task of building and comparing the performance of various RNN models on the dataset, the decision was made to utilize only the stock price data of Vietnam Airlines (VNA).

2. Choice of window size

Two-time window sizes were selected for comparison: 50 days, approximately 1 month of trading days, and 252 days, corresponding to 1 year of trading days. These window sizes will be assessed in terms of model accuracy, with each model trained using an equal number of epochs, batch sizes, and folds in cross-validation. Various aspects of performance will be analyzed and compared across these window sizes.

3. Choice of RNN models

Similar to task 1.

4. Main hyper-parameters

- Number of epochs = 150
- Size of batch = 2048
- Number of folds = 10
- Loss function: mse

5. Results and Discussion

iii. Training – Validation – Testing split

Bi-LSTM model

MSE on the test set

- Window size in 30 days: 0.015

- Window size in 252 days: 0.0038

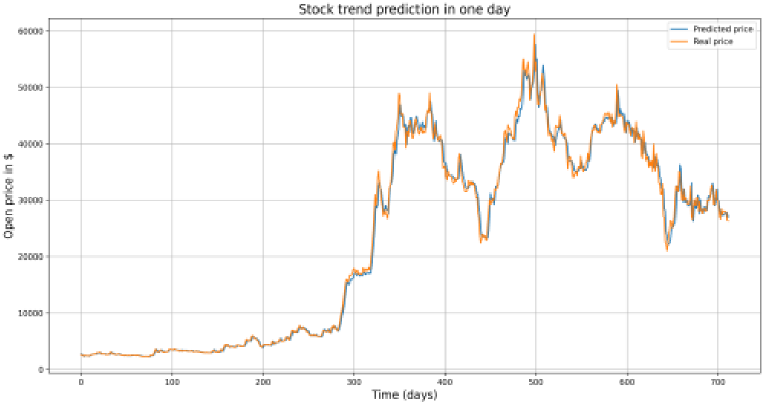


Figure 9. Predicted price from Bi-LSTM model compared with real price with in 30 days

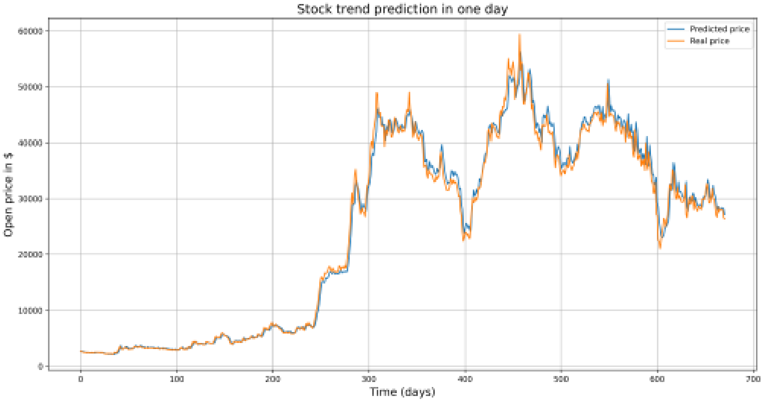


Figure 10. Predicted price from Bi-LSTM model compared with real price with in 252 days

GRU model

MSE on the test set

- Window size in 30 days: 0.015

- Window size in 252 days: 0.0038

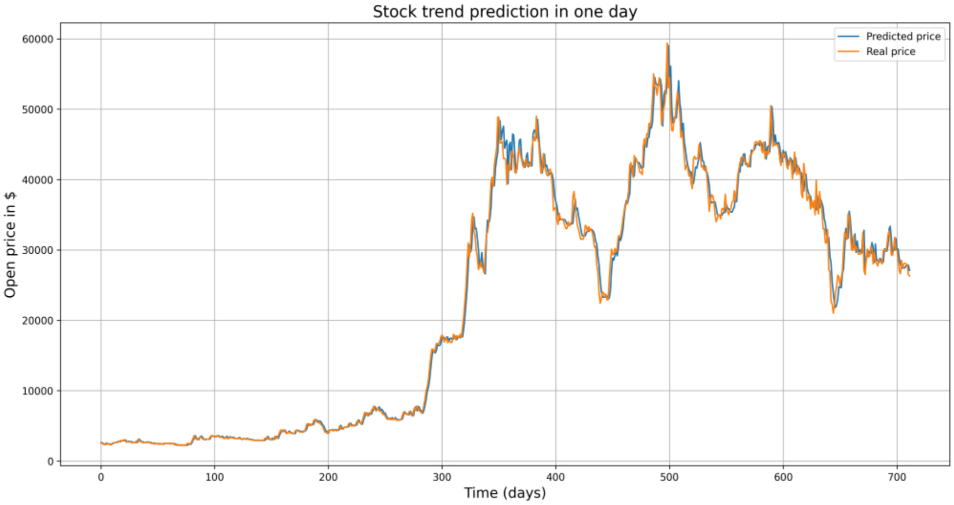


Figure 11. Predicted price from GRU model compared with real price with in 30 days

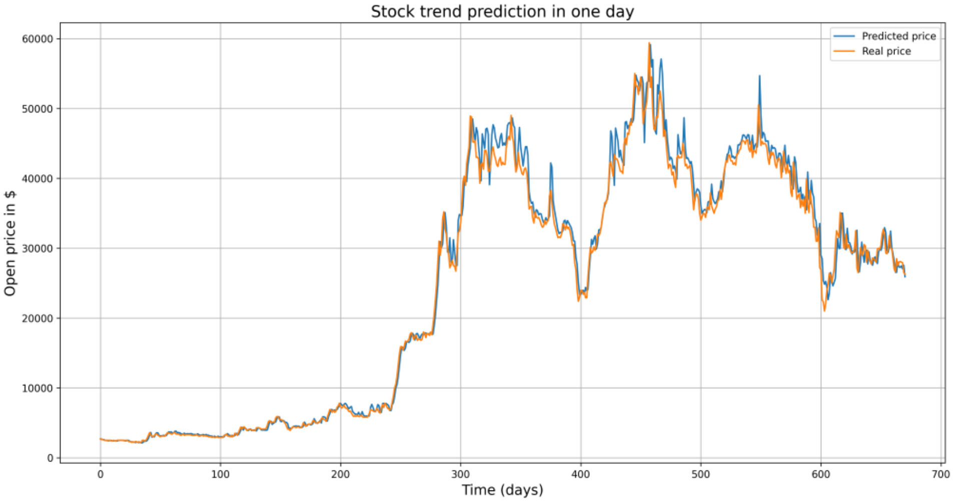


Figure 12. Predicted price from GRU model compared with real price with in 252 days

- **Conv1D model**

MSE on the test set

- Window size in 30 days: 0.069
- Window size in 252 days: 0.18

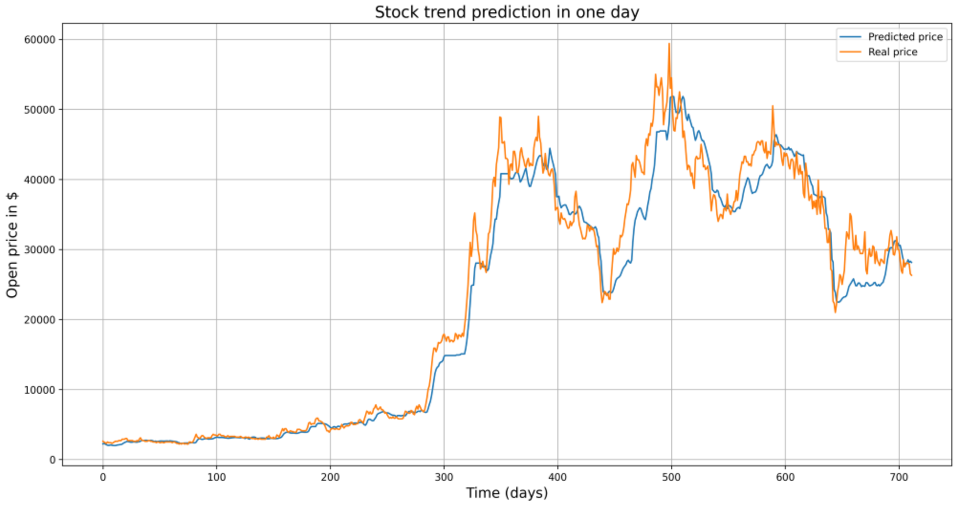


Figure 13. Predicted price from Conv1D model compared with real price with in 30 days

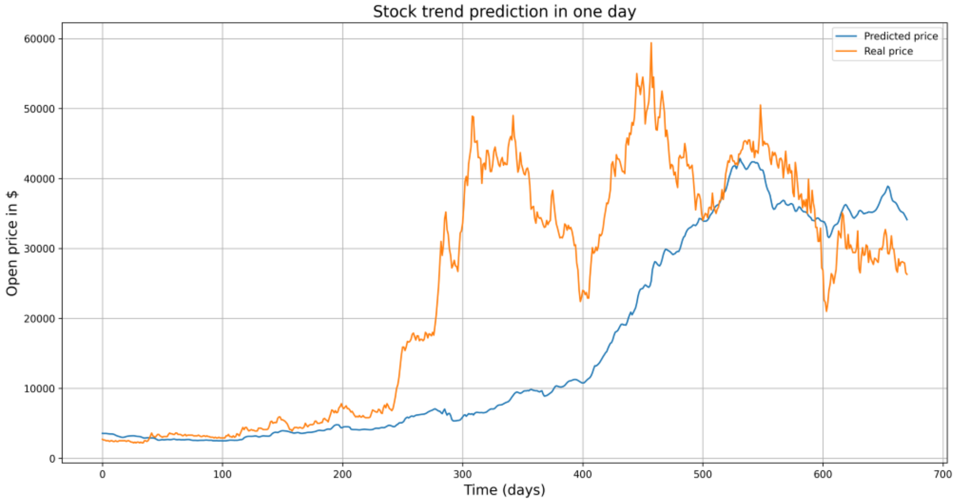


Figure 14. Predicted price from Conv1D model compared with real price with in 252 days

➔ Drawing on the predicted price from RNN models compared with real price and the MSE in two defined window sizes, there are several key findings:

- It is possible to confirm that both experiment window sizes generate good fit with the real price of the company's historical data in Bi-LSTM and GRU models
- The Conv1D model is not appropriate for historical stock price data due to its high loss, though it can still work with window size = 30 better than that with window size = 252.
- However, Conv1D still shows a somewhat predicted line corresponding to the trends from the real price line.

- Additional Vietnam's data such as dividend history, industry analysis, and financial ratio are not preferred to employ since they are only available for some companies and insufficient in terms of data samples.

iv. Cross validation

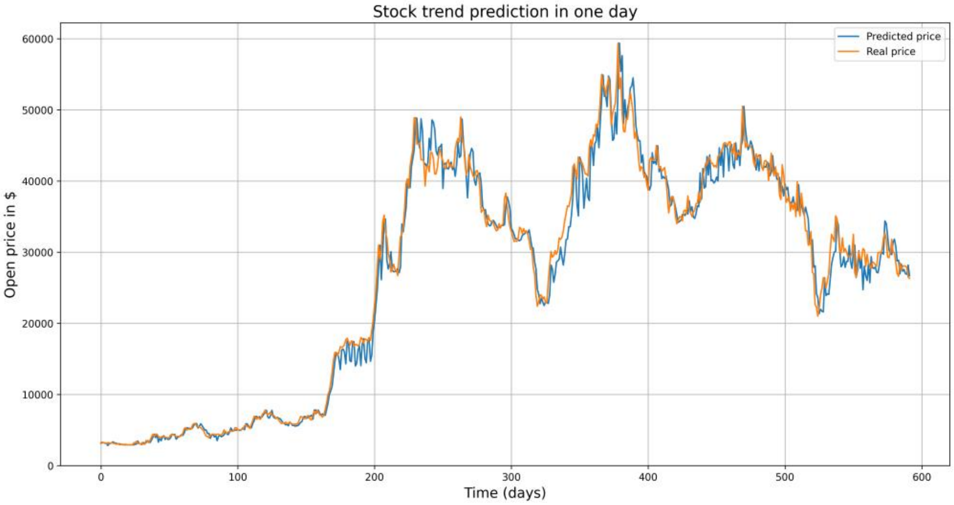


Figure 15. Predicted price from cross validation compared with real price with in 30 days

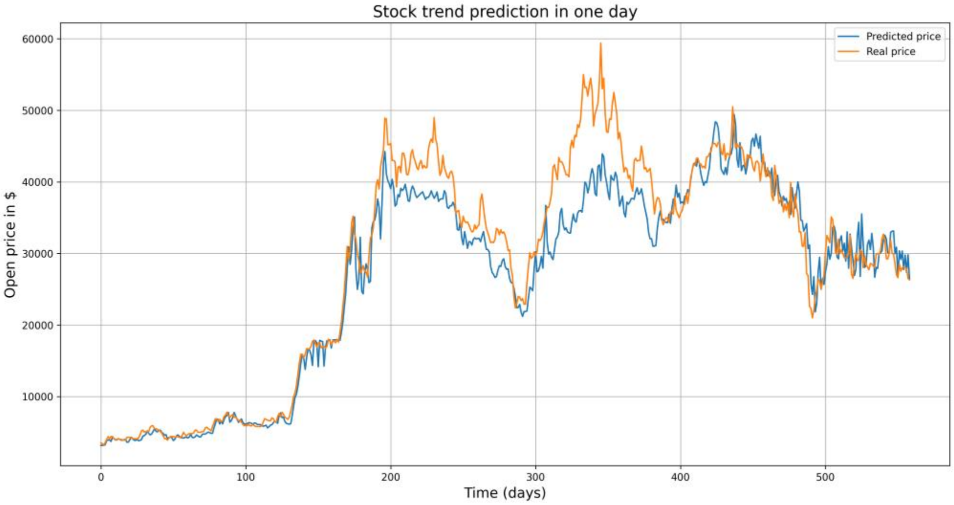


Figure 16. Predicted price from cross validation compared with real price with in 252 days

➔ The cross validation with GRU model prefers small window sizes over larger ones. However, the predicted line with the large window size still, to some extent, matches with all of the ups and downs of the real price.

III. VIETNAM TRADING POINT IDENTIFICATION

1. Choice of stock price data

The stock price data in task 2 is reused in this task 3, which is the historical stock price data of Vietnam Airlines (VNA).

The stock price data is filtered to only remain the data in over the last 5 years (from May 30th, 2018, to Feb 28th, 2023. The data can be illustrated as follows:

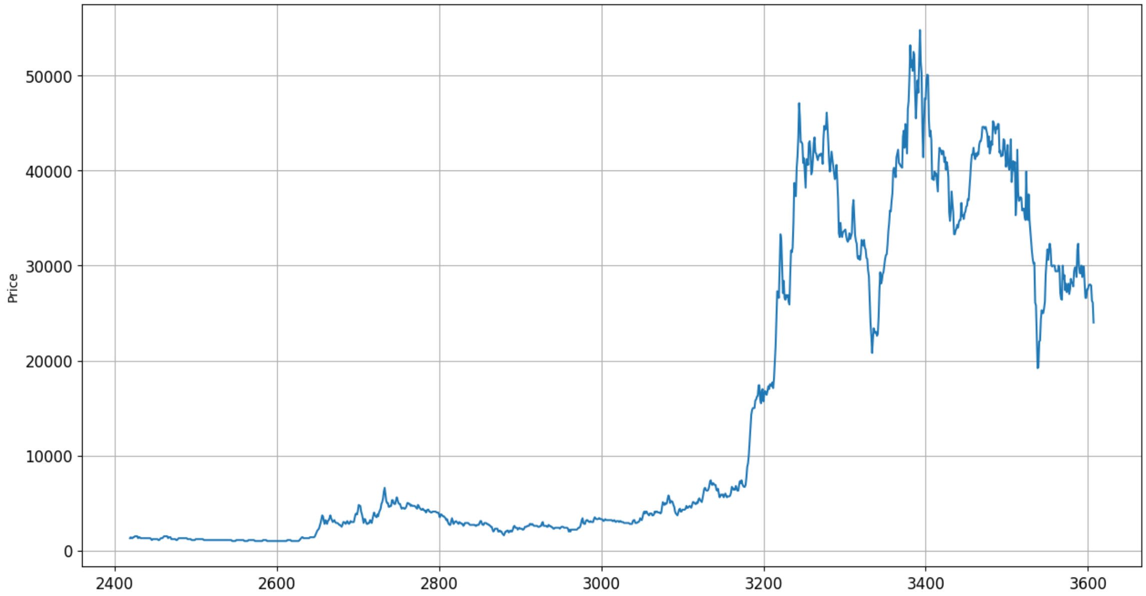


Figure 17. Stock price data of VNA in the last 5 years

2. Choice of window size

- Window size 1 = 50 days
- Window size 2 = 100 days

3. Simple Moving Average

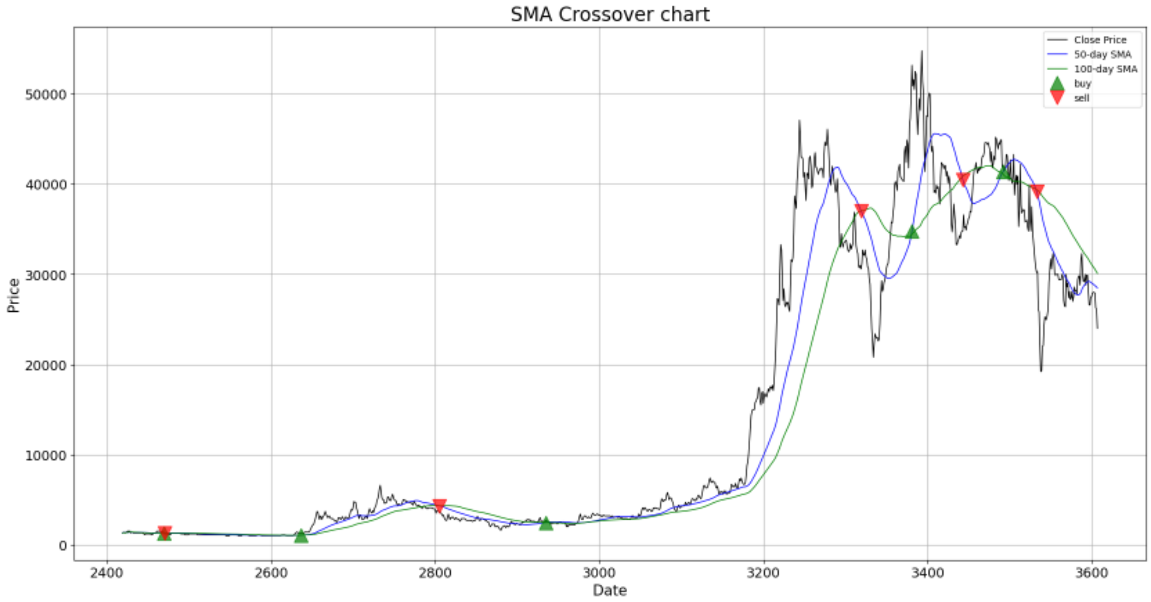


Figure 18. SMA crossover chart

- ➔ It can be seen that the smaller window size has a curve that matches the original data in terms of magnitude and shape.

4. Exponential moving average

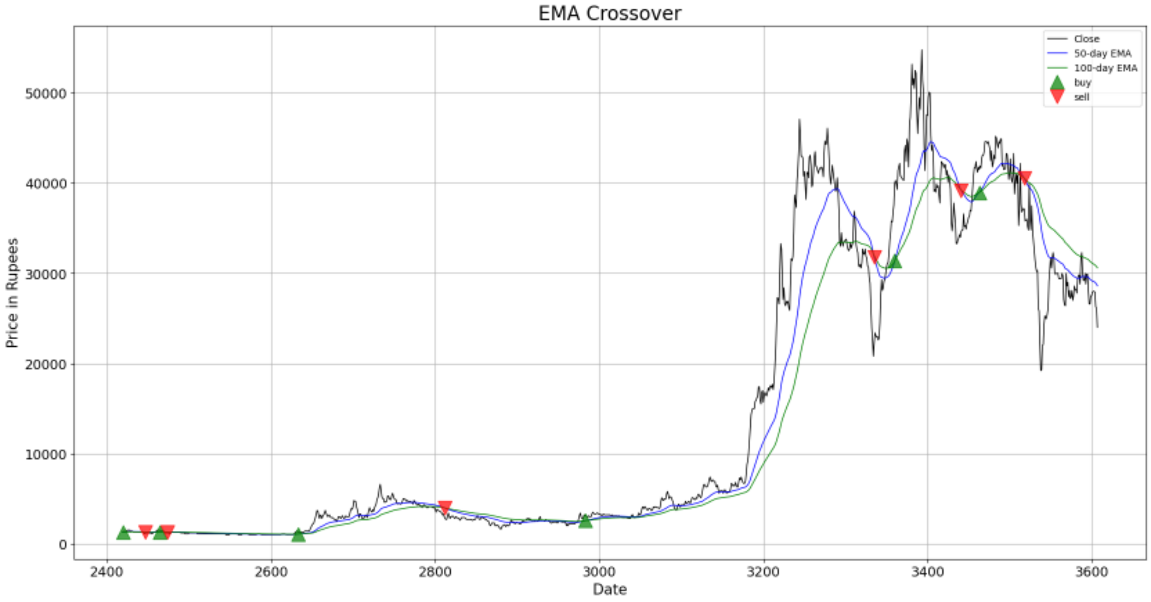


Figure 19. EMA crossover chart

- ➔ Similar to the SMA crossover chart, a smaller window size demonstrates a more effective response when it comes to buying and selling decisions. This phenomenon can be attributed to the fact that the closer the training data is to the current data point, the higher the likelihood of accurately predicting future outcomes over the next few days. By utilizing a smaller window size, the model can capture more recent trends and patterns, enhancing its predictive capabilities and improving the accuracy of buy and sell signals.

IV. NASDAQ PORTFOLIO

1. Choice of stock price data

The selection process begins by choosing the top 100 companies listed on Nasdaq, as they have demonstrated longevity in the market and provide ample data and learning resources. However, these companies undergo further refinement through several filters, including the grouping of companies into their respective industries. Consequently, only a specific set of companies belonging to a particular industry proceed to the preprocessing stage. This approach ensures that the analysis and modeling focus on a targeted group of companies within a clearly identified industry.

2. Hyper-parameters

- Window size (or time steps): 30
- Number of epochs: 200
- Size of the batch: 4096
- Train ratio = 0.8 → to define ratio of data used for training and testing
- Number of companies in the portfolio: 10
- Industry: Industrials (7 companies in top 100 Nasdaq; companies in Industrials usually have a strong core in production, thus they are expected to be quite stable)

3. Choice of RNN model

The multilayer LSTM model is now used to train filtered data. The model is then trained to:

- o Define buy/sell trading points
- o Calculate the profit in a certain period
- o Combine potential scores and risk scores into a portfolio
- o Decide which list of companies are to hold/get rid of

4. Results and Discussion

- MSE on the test set: 0.0023

- Once the model has been trained, a trade table (Figure 21) will be extracted for each company. This table serves as a valuable resource for investors, providing them with clear indications of when to execute buying and selling decisions. By referencing the trade table, investors can effectively navigate the market and make informed choices regarding their investment activities.

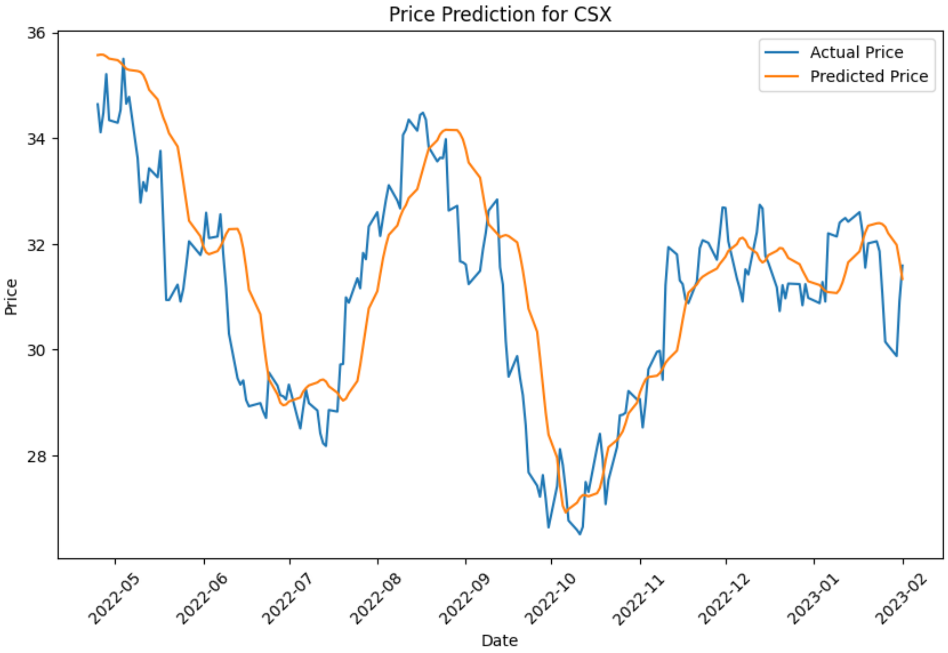
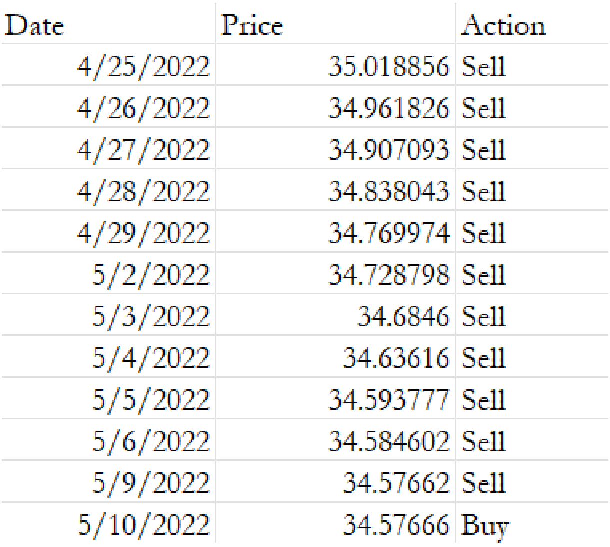


Figure 20. Example of price prediction for company CSX on Nasdaq



Date	Price	Action
4/25/2022	35.018856	Sell
4/26/2022	34.961826	Sell
4/27/2022	34.907093	Sell
4/28/2022	34.838043	Sell
4/29/2022	34.769974	Sell
5/2/2022	34.728798	Sell
5/3/2022	34.6846	Sell
5/4/2022	34.63616	Sell
5/5/2022	34.593777	Sell
5/6/2022	34.584602	Sell
5/9/2022	34.57662	Sell
5/10/2022	34.57666	Buy

Figure 21. Example of trade table of company CSX

- The next step involves calculating a profit list, which is derived from the recommendations for buying and selling. This list holds considerable importance as it aids in determining which companies to retain and which ones to divest from.

- Additionally, potential scores and risk scores will be computed for each company within the industry, providing investors with additional tools to define their investment strategies.

- Finally, the resulting portfolio table is saved as a .csv file for convenient access and reference.

- Potential scores and risk scores will also be calculated for each company in the industry to offer the investors more tools to define their investment strategies

- The final table of portfolio is saved as a .csv file

Company	Profit	Potential Score	Risk Score	Action
CSX	5.1867437	8.641383	5.181173914	Hold
VRSK	2.8607554	4.7661667	34.32608966	Hold
PCAR	-0.60022146	-1	8.136286116	Get Rid
CTAS	2.0698147	3.4484184	92.31865361	Hold
FAST	2.186661	3.6430905	11.06193433	Hold
VRSN	2.325254	3.8739934	31.45205657	Hold
CPRT	0.10251981	0.17080331	15.98313394	Hold

-

- Figure 22. Table of portfolio for companies in Industrials

- ➔ As anticipated, the majority of stocks in the Industrials sector prove to be profitable due to their inherent stability. However, I have encountered numerous challenges in preprocessing the data for model training, making it currently impractical to conduct multiple tests with other RNN models and incorporate cross-validation.

V. HOW TO RUN THE NOTEBOOKS

1. Upload the folder to Google Colab
2. Save the path as path = path = '/content/drive/MyDrive/Final Project DL4AI/*your file here…*'
3. Modify every path in the notebooks
4. The notebooks should run properly by now

→ I completed each task separately from task 1 to task 4

→ Task 4 includes a bit of missing in task 3 (i.e., in task 3, I did not add the training model. There are only SMA and EMA in task 3)