CHAPTER 1 – DOCUMENTATION

1.1 Overview

This section of the dissertation provides a step-by-step guide for using the stock price prediction system developed as part of the Master's dissertation. The project is divided into two key stages: training models on selected stock symbols and forecasting stock prices using the trained models. The user will work with two Jupyter notebooks— training_stocks.ipynb and forecasting_stocks.ipynb—to complete these tasks.

1.2 Prerequisites

Before you begin, ensure that you have the following installed:

- Python (version 3.6 or above)
- Google Colab Notebook
- Required Python libraries: pandas, numpy, scikit- learn, tensorflow, keras, yfinance, and others as specified in the notebooks.

1.3 Step 1: Training the models for each stock

1.3.1 Open “training_stocks.ipynb”

Start by launching Jupyter Notebook and opening the training_stocks.ipynb file. **Refer Figure 33**.

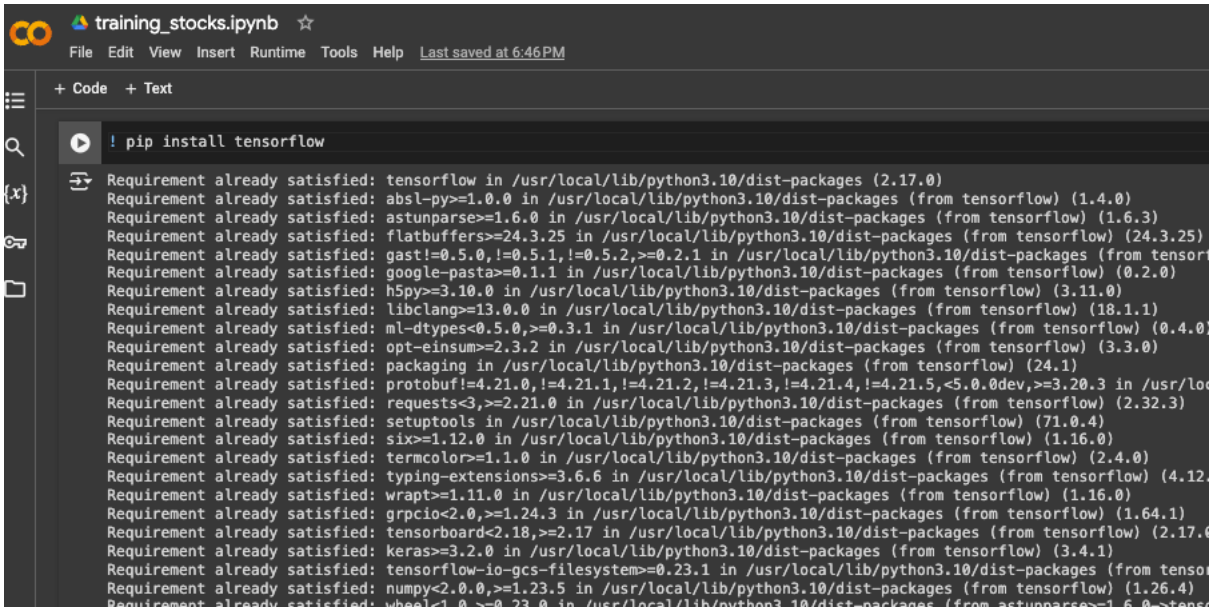

Figure 1 - training_stocks.ipynb

1.3.2 Specify stock symbols

Locate the cell containing the variable “stock_symbols”. This is a Python list where you will input the stock symbols of the companies you wish to train models for.

Example: stock_symbols = ["AAPL", "MSFT", "NVDA"]

Modify this list to include the stock symbols of your choice. Refer **Figure 34.**

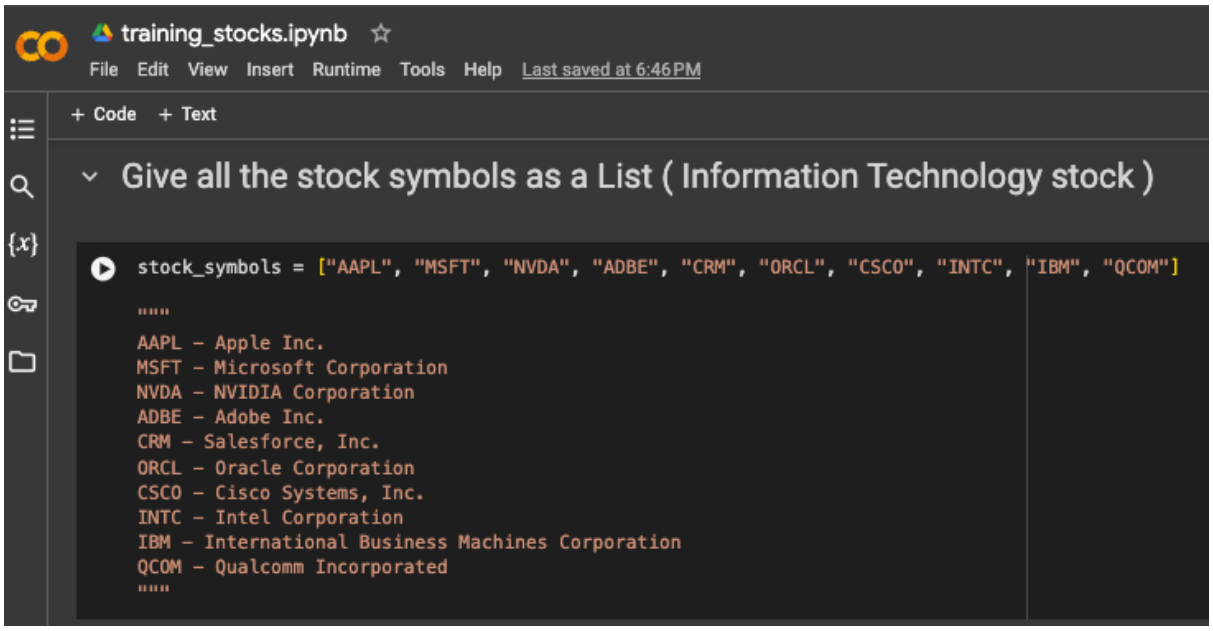

Figure 2 - stock symbols as list of strings

1.3.3 Run the code

After specifying the stock symbols, run all cells in the notebook. You can do this by selecting Cell > Run All from the menu of Google Colab. The notebook will fetch historical stock data, preprocess it, and train the Long Short Term Neural Network model (LSTM) for each stock symbol provided. This process may take time depending on the number of stocks you wish to train. The train time also depends on your system configuration. Use of Google Colab T4 GPU is recommended. **Refer Figure 35.**

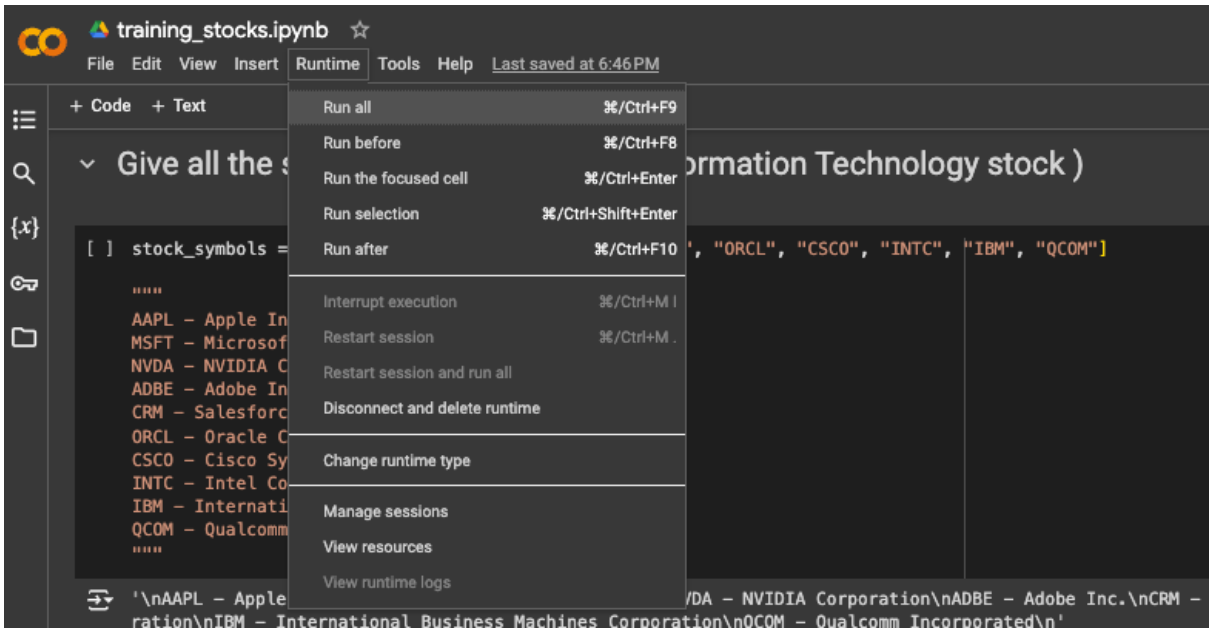

Figure 3- Running the code

1.3.4 Check for trained models

Upon successful completion, a directory named “trained_models” will be created in your working directory. Inside this directory, you will find the trained models saved with filenames corresponding to the stock symbols.

1.4 Step 2: Forecasting stock prices

1.4.1 Open “forecasting.ipynb”

Once the models are trained, open the forecasting_stocks.ipynb file in Google Colab Notebook. Refer Figure , to look at the notebook. Refer Figure 36.

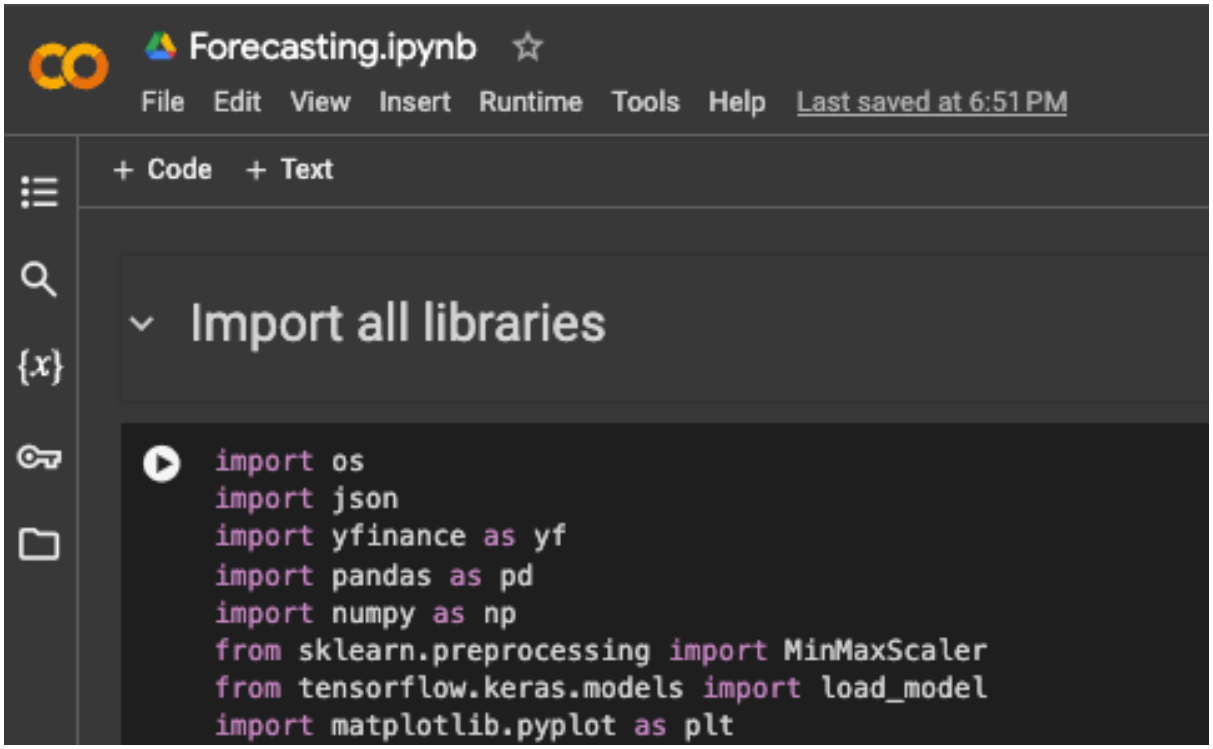

Figure 4 – open” forecasting.ipynb” notebook

1.4.2 Specify the Trained Models Directory

In this notebook, you will find a cell where you need to input the path to the directory containing the trained models. You can find this in the last cell.

Example: models_directory = "trained_models/"

Ensure that the path correctly points to the directory where the trained models are stored.

Refer Figure 37, for the model_directory variable

Refer Figure 38, for the uploading of trained models in a directory.

```
        json.dump(mae_results, json_file, indent=4)

# Directory where models are saved
models_directory = '/content/trained_models'  # Update with the correct path

# List of forecast durations to be performed
forecast_days_list = [30, 60, 90, 120]  # Update with the desired forecast durations

# Output JSON file to store MAE results
output_json_file = 'mae_results.json'

# Process all models in the directory and save MAE results
```

Figure 5 - model_directory variable

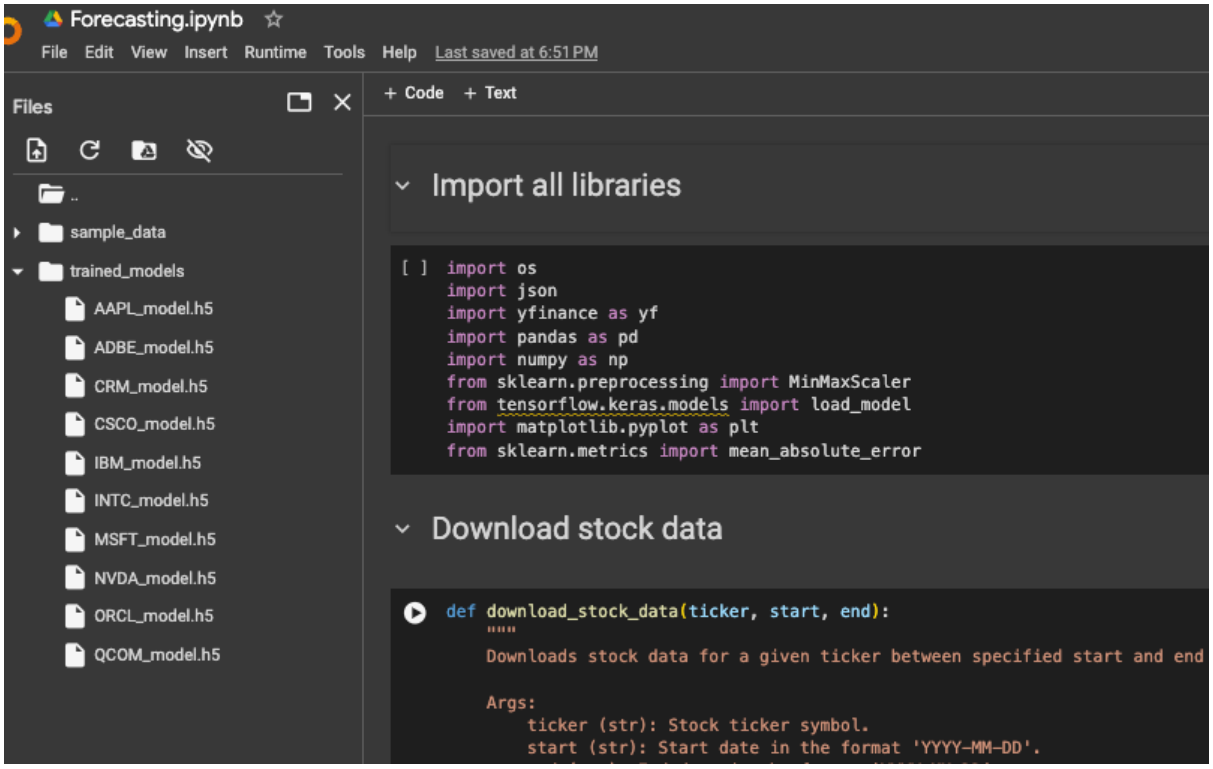

Figure 6 - Upload of trained models in a directory

1.4.3 Run the code

Run all cells in the notebook. This will load the trained models and use them to predict future stock prices based on the latest available data. The output will include predicted prices for each stock symbol that you trained earlier.

1.5 Outputs

- **Training Phase:** The output is a set of trained models stored in the "trained_models" directory. The stock data, hyperparameter tuning results and test set evaluation metrics will also be found.
- **Forecasting Phase:** The output includes a chart comparing the predicted stock prices vs Actual stock prices. The time periods chosen are 30 days, 60 days, 90 days, and 120 days. These predictions are generated based on the trained models and the stock data provided during training. Mean absolute error will also be computed for each time period and displayed on top of the price chart.

1.6 Troubleshooting

- **Error in Fetching Stock Data:** Ensure that the stock symbols are valid and correctly formatted. The symbols should match those used by Yahoo Finance (e.g., "AAPL" for Apple Inc.).
- **Model Not Found:** Verify that the path provided points to the directory containing the trained models in the file “forecasting_stocks.ipynb. “
- **Library Issues:** Ensure all required Python libraries are installed and up to date. If you encounter errors related to missing libraries, use “pip install library_name” to install the required packages.

1.7 Conclusion

This system allows for efficient training and forecasting of stock prices based on selected stock symbols. By following the steps outlined in this documentation, users can train models on their desired stocks and generate forecasts to aid in investment decisions.

Investing in the stock market inherently involves market risk. This project leverages historical closing stock prices to predict future trends. However, it is crucial that you conduct thorough independent research and due diligence before making any investment decisions.

For further assistance or questions regarding this project, please refer to the contact information provided in the dissertation.

1.8 Contact

Name: Midhun Lakshmanasamy Nirmala

Student ID : 201711362

Email : sgmlaksh@liverpool.ac.uk