Nguyen Ngoc Gia Thinh

Student ID: 103809954

COS30018

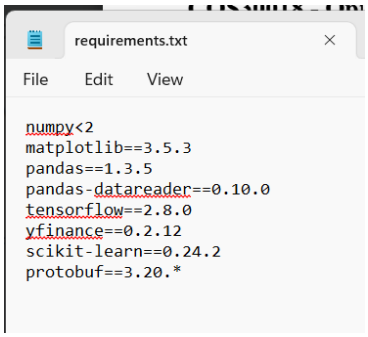
Task 1: Weekly Report

Your Task 1 Report will contain the following details:

• Summaries of your attempt to setup your environment, including details of your requirements file.

Setting up the virtual environment required multiple attempts of installing and uninstalling various packages. After troubleshooting, I was able to establish a stable configuration with the following dependencies:

numpy	< 2
matplotlib	==3.5.3
pandas	== 1.3.5
pandas-datareader	== 0.10.0
tensorflow	==2.8.0
yfinance	==0.2.12
scikit-learn	==0.24.2
protobuf	==3.20.*



This setup ensures compatibility and stability for the required tasks.

• Summaries of your attempts to test the provided code bases (v0.1 and P1) with screenshots.

During testing, I encountered issues due to changes in external data sources. Specifically, since Facebook rebranded to Meta, I had to modify the data loading process from FB to META. After updating the code and resolving errors, I obtained the expected results. Below is the output after fixing the issues:

These results align more closely with my expectations and confirm that the modified code is functioning correctly.

• Summary of your understanding of the initial code base v0.1.

The initial codebase (v0.1) is designed to facilitate stock price prediction using machine learning techniques. It integrates external libraries to collect, preprocess, and analyze financial data. Based on my review of the provided script, key components include:

- **Data Retrieval:** The script utilizes yfinance to fetch historical stock price data, ensuring up-to-date and accurate market trends.
- **Data Preprocessing:** The closing price is normalized using MinMaxScaler, transforming it into a suitable range for training machine learning models.
- **Feature Engineering:** The model looks back at the previous 60 days (PREDICTION_DAYS) to forecast future stock prices, creating structured training data.
- **Machine Learning Model:** The script employs a stacked Long Short-Term Memory (LSTM) neural network within TensorFlow's Sequential model. This architecture consists of multiple LSTM layers with dropout regularization to prevent overfitting.
- **Model Training:** The model is trained using historical stock prices, with adam optimization and mean_squared_error as the loss function to improve accuracy.
- **Predictions & Visualization:** The trained model predicts future stock prices, which are then visualized using matplotlib to compare actual versus predicted values.