ĐẠI HỌC BÁCH KHOA HÀ NỘI

TRƯỜNG CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG

BÁO CÁO TUẦN

Môn học: Project II

Chủ đề: Xây dựng mô hình dự đoán giá cổ phiếu bằng mạng Transformer

Giáo viên hướng dẫn: **Đỗ Tuấn Anh**

Sinh viên thực hiện: ***Lê Minh Triết***

Mã số sinh viên: **20220045**

Hà Nội - 2025

MỤC LỤC

Contents

1. Công việc đã hoàn thành trong tuần

1.1. Xác định rõ phạm vi của project

Đề tài của project đã được xác định từ trước, là **xây dựng mô hình dự đoán giá cổ phiếu**.

Để làm rõ phạm vi của đề tài, em đã tìm hiểu các NoteBook đã có và phát hiện thường họ chỉ huấn luyện trên 1 cổ phiếu cụ thể. Do vậy em quyết định lựa chọn phạm vi của đề tài là **xây dựng mạng Transformer có khả năng dự đoán giá đóng cửa của cổ phiếu trong trung hạn (5 ngày) và có khả năng tổng quát hóa cho nhiều cổ phiếu**. Lý do tại sao là trung hạn (5 ngày – số ngày hoạt động của cổ phiếu trong tuần) chứ không phải chỉ là dự đoán 1 ngày ngay sau em sẽ nhắc tới trong phần *1.6.*

Em cũng dự định sẽ thêm các tùy chọn Fine-tuned mô hình tổng quát này nếu cần thiết để dự đoán chính xác hơn 1 cổ phiếu nào đó.

Chính vì vậy, em quyết định thu thập dữ liệu từ nhiều cổ phiếu để phân tích và huấn luyện

1.2. Thu thập dữ liệu cổ phiếu từ Yahoo Finance

Dữ liệu thô: Dữ liệu thô được thu về qua lời gọi hàm từ thư viện yfinance của python

```
yf.download(tickers=tickers, start=start, end=end, group_by='ticker')
```

Với tickers là một mảng gồm 47 cổ phiếu em dự định thu về.

Dữ liệu thu về được lưu ở dạng một cột đa thuộc tính, sau khi qua các bước xử lý đơn giản, em đã đưa dữ liệu về dạng bảng như sau:

	Open	High	Low	Close	Volume	Ticker	Collect Date
0	16.100000	16.396667	15.942000	16.312668	92439000.0	TSLA	2019-10-01
1	16.219334	16.309999	15.962000	16.208668	84471000.0	TSLA	2019-10-02
2	15.457333	15.632000	14.952000	15.535333	226267504.0	TSLA	2019-10-03
3	15.440667	15.652000	15.204667	15.428667	119925000.0	TSLA	2019-10-04
4	15.320000	15.904000	15.236667	15.848000	120963000.0	TSLA	2019-10-07
...	...	...	...	...	...	...	...
65560	2894.000000	2931.500000	2863.000000	2882.500000	24980800.0	7203.T	2025-01-24
65561	2913.500000	2941.500000	2910.500000	2922.000000	18257200.0	7203.T	2025-01-27
65562	2900.000000	2936.000000	2889.500000	2889.500000	18314000.0	7203.T	2025-01-28
65563	2917.000000	2936.500000	2898.500000	2930.000000	17997800.0	7203.T	2025-01-29
65564	2927.000000	2963.500000	2922.500000	2949.000000	18783700.0	7203.T	2025-01-30

65565 rows × 7 columns

Hình 1. Dữ liệu sau khi tái cấu trúc

Trong đó, Open – High – Low – Close lần lượt là giá cổ phiếu lúc mở cửa, lúc cao nhất, lúc thấp nhất, lúc đóng cửa trong ngày, trong khi Volume là khối lượng giao dịch cổ phiếu trong ngày.

Dữ liệu được dùng để huấn luyện sẽ là dữ liệu của các cổ phiếu **từ ngày 01-01-2020 tới 31-12-2024**. Trong dữ liệu thô có mở rộng ở 2 đầu 2 mốc thời gian này, do em đã lên kế hoạch

sử dụng sliding window cho dữ liệu thời gian để tạo thành chuỗi thời gian, nên cần thêm dữ liệu 2 đầu để các dữ liệu chính không bị thiếu.

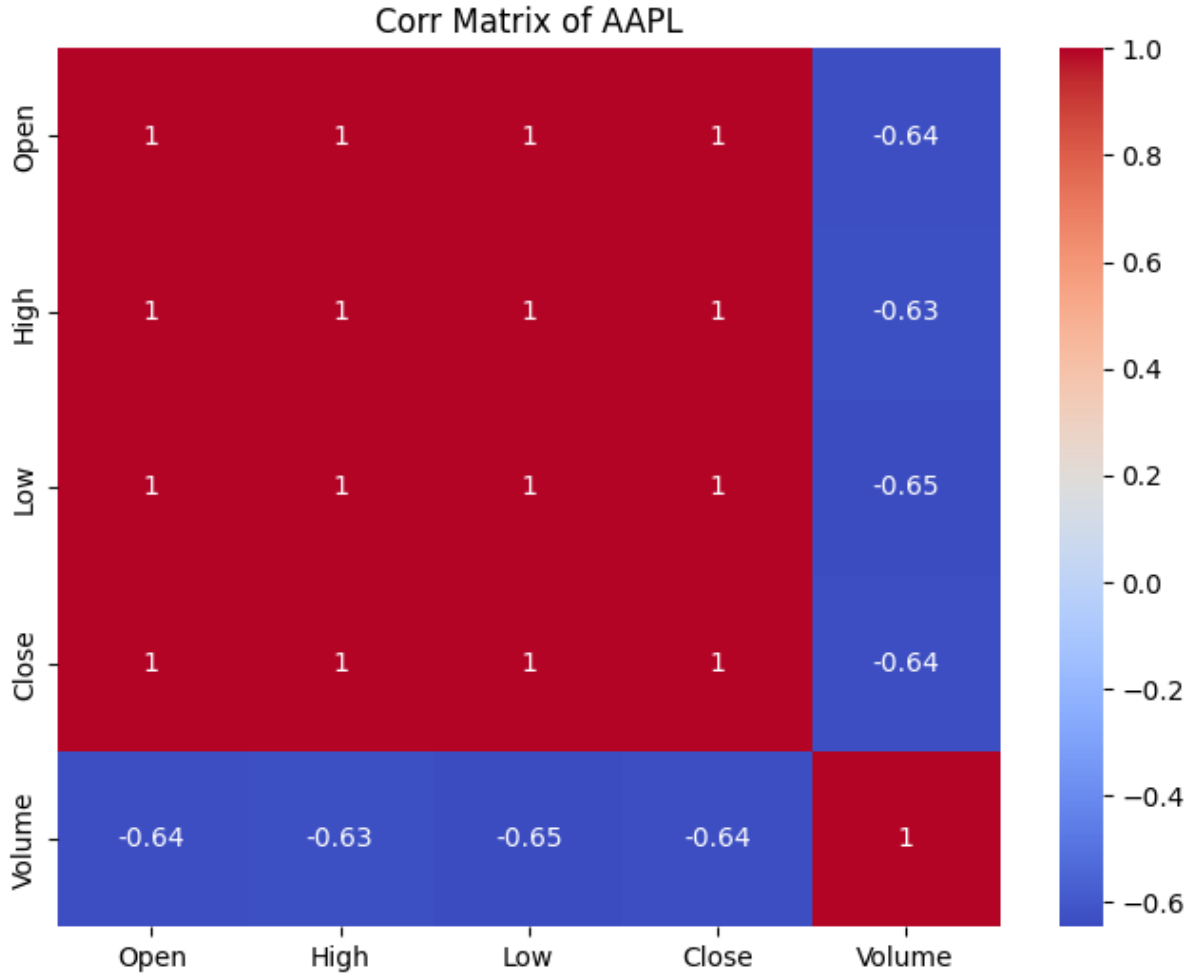
1.3. Phân tích dữ liệu cổ phiếu để lựa chọn đặc trưng huấn luyện

Do phạm vi của dự án là xây dựng mô hình, nên phần phân tích dữ liệu của em sẽ là khá đơn giản và chỉ tập trung phân tích để hiểu rõ 1 vài xu hướng giá, và quan trọng nhất, là lựa chọn đặc trưng nào để huấn luyện mô hình.

Các thống kê, biểu đồ, nhận xét chi tiết em đã để ở

https://github.com/trietp1253201581/StockPrediction/blob/main/data/EDA.ipynb

Sau đây em chỉ điểm qua phần quan trọng nhất, đó là sử dụng tương quan để lựa chọn đặc trưng. Em lựa chọn 2 cổ phiếu để phân tích là AAPL và NVDA, sau khi vẽ ma trận tương quan, em thu được



Hình 2. Ma trận tương quan của các đặc trưng trong cổ phiếu AAPL

Do các đặc trưng về giá bằng 1 (hoặc gần bằng) nên em đánh giá có khả năng chúng sẽ không hỗ trợ nhau trong quá trình huấn luyện, điều này cũng xảy ra với cổ phiếu NVDA. Chính vì vậy, qua tìm hiểu và phân tích, em chỉ lựa chọn 2 đặc trưng cho việc huấn luyện là Close và Volume. Đặc trưng Close được lựa chọn vì qua tìm hiểu, em biết rằng giá đóng cửa (Close) phản ánh rõ nhất giá cổ phiếu trong 1 ngày. Đồng thời, **Close cũng là target trong model.**

1.4. Tiền xử lý dữ liệu cho huấn luyện

Em đã xây dựng module tiền xử lý dữ liệu với class StockDataProcessor, có nhiệm vụ load lại dữ liệu thô vào DataFrame và tái cấu trúc; điền dữ liệu thiếu (đơn giản bằng ffill và bfill), scale lại dữ liệu để phù hợp cho huấn luyện (có lưu trữ để sau này tập test cần inverse scale sẽ sử dụng); cuối cùng là tạo sliding window (sliding window là L nghĩa là dùng dữ liệu lịch sử của L ngày để dự báo cho 5 ngày kế tiếp) rồi chia dữ liệu thành 3 tập train, validation, test.

Các dữ liệu train, validation, test đều cho dưới dạng các numpy array cho dễ xử lý và đưa vào mô hình.

Chi tiết em đã để ở file

https://github.com/trietp1253201581/StockPrediction/blob/main/data/process.py

1.5. Xây dựng kiến trúc model tổng quát

Để sau này dễ dàng hơn trong việc xây dựng mô hình, em đã xây dựng một số mô hình model trừu tượng, bao gồm

1. BaseModel: Chứa các phương thức train, predict không được cài đặt (phương thức trừu tượng).

2. BasePytorchModel: Cài đặt các phương thức train, predict chung cho tất cả các model học sâu dùng PyTorch.

Chi tiết em để ở file

https://github.com/trietp1253201581/StockPrediction/blob/main/data/common.py

1.6. Xây dựng baseline

Em chọn 2 baseline đơn giản nhưng rất mạnh trong các bài toán chuỗi thời gian, đó là

1. LastDayModel: Lấy dữ liệu giá đóng cửa của ngày gần nhất biết được để làm dự đoán cho toàn bộ 5 ngày sau. Nghĩa là hiện tại đang là ngày 8/3/2025 và biết giá đóng cửa là 123.5 thì dự đoán cho 5 ngày sau đó là 123.5, 123.5, 123.5, 123.5.

   Đây là mô hình đơn giản nhất, nhưng cũng rất mạnh mẽ, đặc biệt khó bị vượt qua khi chỉ dự đoán 1 ngày duy nhất, vì giá Close của 2 ngày liên tiếp cũng có tương quan rất cao. Đây cũng là lý do em lựa chọn dự báo trung hạn thay vì ngắn hạn, vì các mô hình Deep Learning thường sẽ phù hợp với trung hạn và dài hạn hơn là ngắn hạn.

2. MAModel (Moving Average): Lấy dữ liệu n ngày trước đó và tính trung bình, dùng làm dự báo cho ngày thứ n+1. Baseline này không mạnh bằng baseline trên, nhưng nó tốt hơn trong quá trình dự báo xu hướng.

Hai base line trên đều không cần huấn luyện mà dự đoán trực tiếp trên đầu vào. Chi tiết ở file

https://github.com/trietp1253201581/StockPrediction/blob/main/data/baseline.py

1.7. Đọc Notebook trên Kaggle và triển khai lại LSTM điển hình.

Em đã đọc 1 vài Notebook trên Kaggle và lựa chọn mô hình LSTM điển hình nhất để triển khai. Cấu trúc chính của mô hình này là

- 1 hoặc nhiều lớp LSTM được xếp chồng.
- Thành phần Fully Connected với 2 lớp Linear và 1 hàm kích hoạt ReLU.

Chi tiết xem ở file

https://github.com/trietp1253201581/StockPrediction/blob/main/data/lstm.py

Em cũng đã chạy thử mô hình LSTM này (chỉ dùng 1 lớp LSTM) trên bộ dữ liệu đầu vào với sliding window bằng 40 và dự báo 5 ngày. Kết quả so sánh với 2 baseline thì LSTM có lỗi trung bình MSE trong 5 ngày thấp hơn so với 2 baseline. Qua đó cũng khẳng định sức mạnh của các mô hình Deep Learning trong dự báo trung và dài hạn. Chi tiết em đã để ở

https://github.com/trietp1253201581/StockPrediction/blob/main/data/Test.ipynb

2. Dự kiến các công việc tuần tới

Tuần này em đã cố gắng làm xong các công việc mình đã biết, vì vậy, kế hoạch cho tuần tới của em là.

Tìm hiểu và nắm rõ lý thuyết về mạng Transformer, cách nó hoạt động và các cách triển khai Transformer, đặc biệt phải phù hợp với dữ liệu chuỗi thời gian.

Do em chưa có bất kỳ kiến thức nào về mạng Transformer nên việc tìm hiểu lý thuyết theo em đánh giá là quan trọng và sẽ có thể tốn nhiều thời gian hơn 1 tuần.