**BÁO CÁO PROJECT III**

**Dự đoán giá chứng khoán bằng mô hình học sâu**

**PHẠM MINH KHÔI – 20183566 – IT1-02-K63**

Khoi.pm183566@sis.hust.edu.vn

**Ngành Khoa học Máy tính**

|  |  |
| --- | --- |
| **Giảng viên hướng dẫn:** | TS. Nguyễn Tuấn Dũng |
| **Bộ môn :** | Project 3 |
| **Viện :** | Công nghệ thông tin và Truyền thông |

**HÀ NỘI, 12/2021**

**Lời cảm ơn**

Trong thời gian làm bài tập lớn môn Project III, em đã nhận được rất nhiều sự giúp đỡ, đóng góp ý kiến của thầy và các bạn.

Em xin gửi lời cảm ơn chân thành đến TS. Nguyễn Tuấn Dũng, giảng viên Viện Công nghệ thông tin và truyền thông, trường Đại học Bách khoa Hà Nội, người trực tiếp hướng dẫn em hoàn thành bài tập lớn môn Project III.

Em cũng xin cảm ơn các thầy cô, bạn bè đã luôn tạo điều kiện quan tâm, giúp đỡ để em hoàn thành học phần này.

Em rất vinh dự nếu nhận được sự góp ý của thầy cô và các bạn để hoàn thiện đề tài này hơn.

Em xin chân thành cảm ơn!

**Tóm tắt nội dung**

* Vấn đề cần thực hiện: xây dựng mô hình học sâu để có thể dự đoán được giá trị của dữ liệu dạng time series forecast, cụ thể là giá trị chứng khoán
* Phương pháp thực hiện:
  + Tìm hiểu tổng quan về các thư viện python phổ biến và các thư viện hỗ trợ: investpy, yfinace, pandas\_datareader, keras, …
  + Tiền xử lý dữ liệu.
  + Huấn luyện mạng học sâu như CNN, LSTM.
  + Đề xuất hàm phạt trong việc huấn luyện.
  + Biểu diễn dữ liệu bằng matplotlib.
* Kết quả: Dự đoán được các giá trị trong một khoảng thời gian của mã chứng khoán .
* Tính thực tế: Chương trình giúp đỡ và hỗ trợ trong việc ra quyết định mua bán, trading của nhà đầu tư ngắn hạn và dài hạn.
* Định hướng phát triển: Tìm cách cải tiến mới tối ưu tốc độ và bộ nhớ, đồng thời kết hợp với các phương pháp khác trong việc huấn luyện mô hình để cải thiện độ chính xác…

Sinh viên thực hiện

Ký và ghi rõ họ tên

Phạm Minh Khôi

**MỤC LỤC**

[CHƯƠNG 1. PHÁT BIỂU ĐỀ TÀI 4](#_Toc60267668)

[1.1 Hiện trạng và tính thực tế 4](#_Toc60267669)

[1.2 Phát biểu bài toán 4](#_Toc60267669)

[1.3 Mô tả dữ liệu 4](#_Toc60267669)

[1.4 Đề xuất ý tưởng giải quyết 4](#_Toc60267669)

[CHƯƠNG 2. GIẢI QUYẾT BÀI TOÁN 4](#_Toc60267668)

[2.1 Công cụ sử dụng 4](#_Toc60267670)

[2.1.1 Thư viện hỗ trợ 4](#_Toc60267672)

[2.1.2 Mạng LSTM 4](#_Toc60267672)

[2.1.3 Mạng CNN 4](#_Toc60267672)

[2.2 Tiền xử lý dữ liệu 4](#_Toc60267670)

[2.3 Đề xuất hàm mất mát 4](#_Toc60267670)

[2.4 Huấn luyện mô hình 4](#_Toc60267670)

[CHƯƠNG 3. KẾT QUẢ THỬ NGHIỆM VÀ ĐÁNH GIÁ 7](#_Toc60267674)

[CHƯƠNG 4. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN 8](#_Toc60267674)

[TÀI LIỆU THAM KHẢO 8](#_Toc60267677)

**CHƯƠNG 1: PHÁT BIẾU BÀI TOÁN**

* 1. **Hiện trạng và tính thực tế**

Ngày nay, trong cuộc cách mạng khoa học công nghệ 4.0, trí tuệ nhân tạo và học máy trở thành những ngành đi đầu, tiên phong trong việc ứng dụng vào các vấn đề thiết thực của đời sống; vì vậy trí tuệ nhân tạo đã có những bước tiến vô cùng mạnh mẽ và vẫn đang trên đà phát triển.

Không nằm ngoài xu thể đó, việc áp dụng trí tuệ nhân tạo vào các ngành nghề có nhiều dữ liệu nhưng lại không ổn định như tài chính thì vẫn còn là một thách thức lớn. Cụ thể trong việc dự đoán biến động lên xuống của giá chứng khoán cũng đã là một bài toán khó khăn nhưng đã được phần nào hỗ trợ bởi các mô hình học máy như LSTM, RNN.

**1.2 Phát biểu bài toán**

Bài toán cần được giải quyết là: xây dựng được mô hình học sâu xử lý đầu vào là dữ liệu dạng thời gian của một mã chứng khoán, từ đó cho ra các giá trị trong khoảng gian kế tiếp sau đó.

* 1. **Mô tả dữ liệu**

Tập dữ liệu trong bài toán là giá trị giao dịch của mã cổ phiếu 0005.HK trong giai đoạn 2000-2020.

* 1. **Đề xuất ý tưởng giải quyết**

Có thể dễ dàng nhận thấy đây là một bài toán kiểu time-series-forecasting/ dự đoán dữ liệu thời gian tương tự như các bài toán như dự đoán thời tiết, dự đoán doanh số bán hàng, tất cả đều có điểm chung đó là “quan sát các biến động của dữ liệu trong một khoảng thời gian để từ đó đưa ra quan điểm và dự báo trong khoảng thời gian tiếp theo”. Đây là một nhận định cơ bản, chưa thể bao hàm hết được sự phức tạp của vấn đề trong việc dự đoán, bởi vì dữ liệu lại nhận ảnh hưởng ngược lại từ quyết định mua/bán của nhà đầu tư. Tuy nhiên trong giới hạn đề tài, ta giả định dữ liệu đã đủ chính xác để có thể đưa ra nhận định trong tương lai gần.

Từ đó, ta có thể giải quyết bài toán như sau: sử dụng mô hình LSTM để xử lý dữ liệu đầu vào dạng time-series.

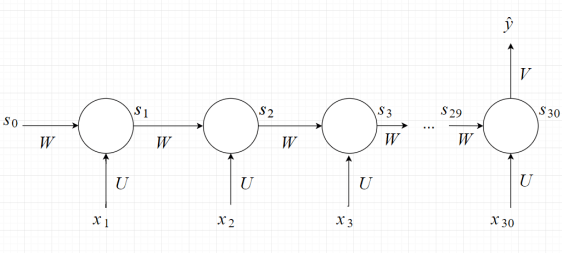
**CHƯƠNG 2: GIẢI QUYẾT BÀI TOÁN**

**2.1 Công cụ sử dụng**

**2.1.1 Thư viện hỗ trợ**

* Thư viện nhập xuất và các hàm tính toán cơ bản: pandas + numpy
* Thư viện hỗ trợ lấy dữ liệu từ web: yfinance + investpy
* Thư viện xây dựng model deep learning: scikit-learn + keras
* Thư viện biểu diễn hình ảnh: matplotlib

**2.1.2 Mô hình mạng LSTM**



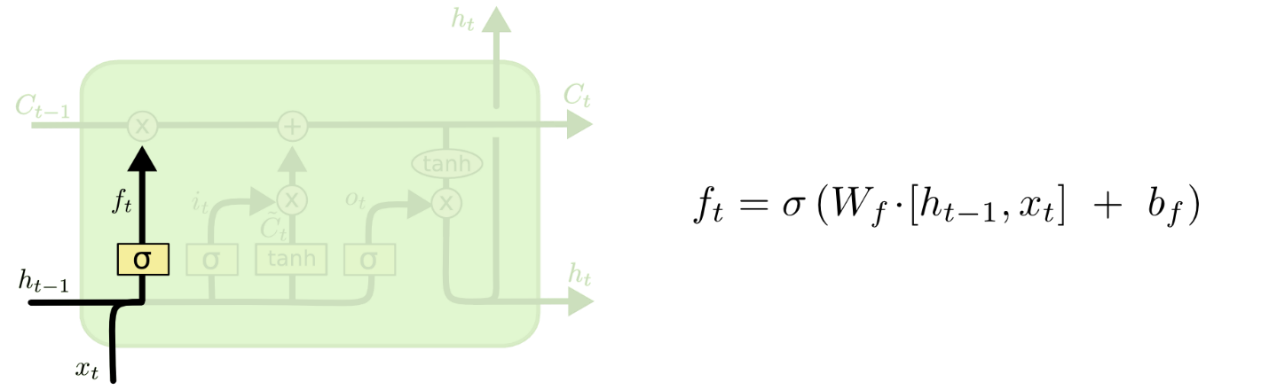
Cấu trúc mạng LSTM được cải tiến dựa trên việc khắc phục các nhược điểm của mạng RNN đó chính là vấn đề RNN chỉ nhớ được những thông tin ở gần vị trí hiện tại của nó, hay còn gọi là short term memory, vì vậy ý tưởng về Long short-term memory ra đời.

LSTM được giới thiệu bởi Hochreiter & Schmidhuber(1997) , khắc phục được vấn đề phụ thuộc xa nên đã dần trở nên phổ biến và thay thế RNN trong các bài toán xử lý dữ liệu chuỗi.

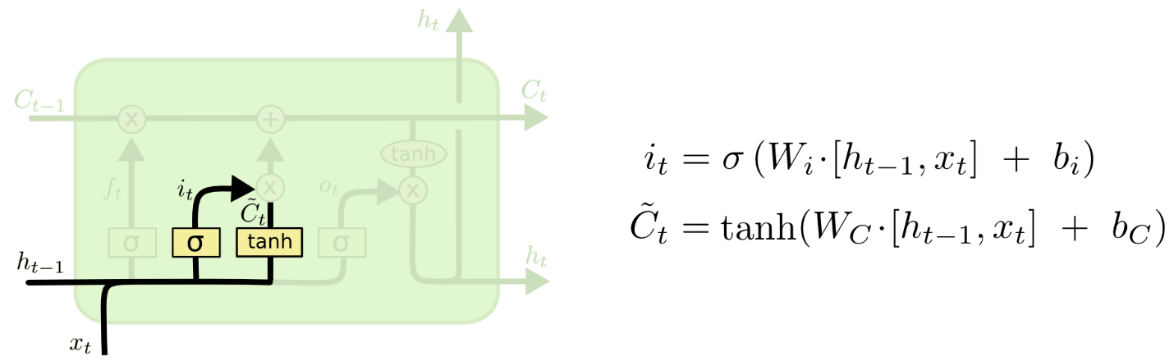


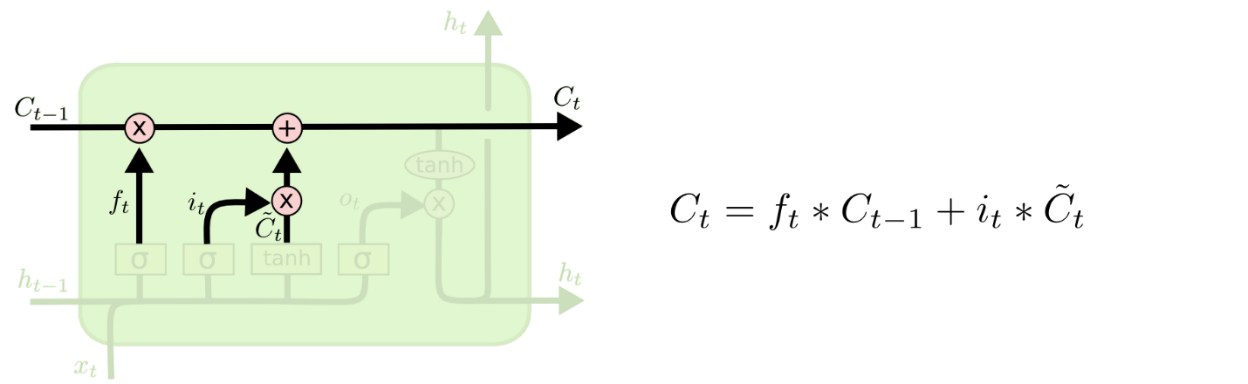
Cấu trúc của LSTM bao gồm các cell như trên, mỗi cell lại có thêm các cổng (gate) bao gồm một mạng sigmoid và một phép nhân. Trong mỗi bước lặp của LSTM thì thông tin sẽ được xử lý theo các bước:

1. Quyết định thông tin nào cần bỏ đi thông qua forget gate, đầu ra sigmoid là giá trị nhớ = 1 và quên = 0.

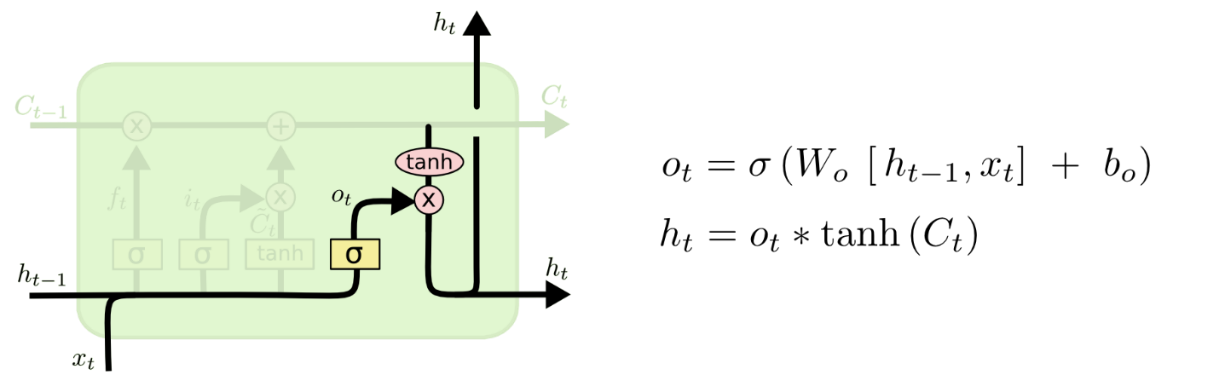


1. Lựa chọn thông tin mới sẽ được lưu trong cell, phần này bao gồm: lựa chọn thông tin để thêm vào, từ đó tính toán thông tin sẽ thêm và cập nhật thông tin đó.



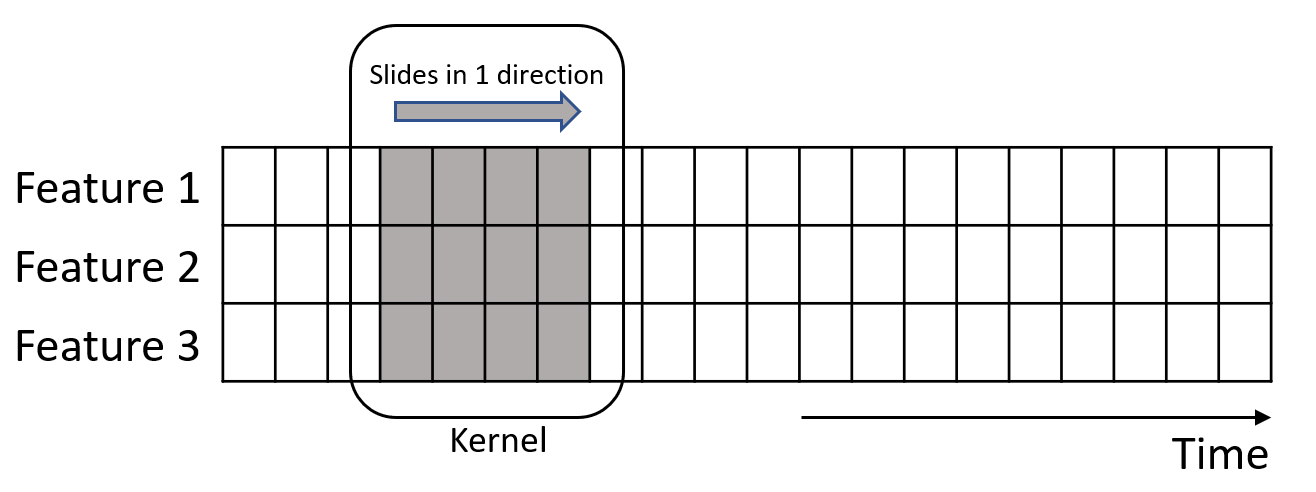


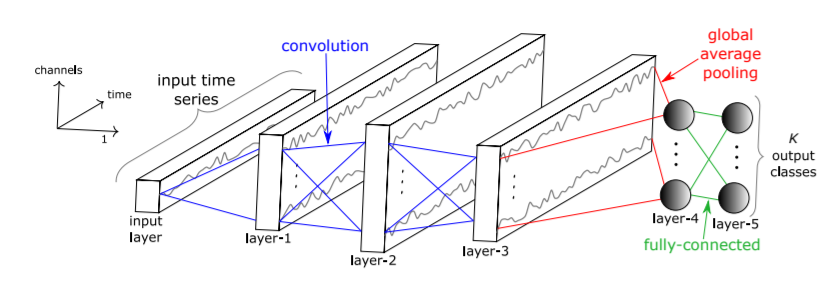
1. Tính toán output để gửi đến lần lặp tiếp theo, giá trị đầu ra sẽ dựa vào trạng thái tế bào, nhưng sẽ được tiếp tục sàng lọc. Đầu tiên ta chạy một tầng sigmoid để quyết định phần nào trong trạng thái ta muốn xuất, sau đó đưa qua 1 hàm tanh để co về giá trị [-1,1] và nhân nó với đầu ra ở cổng sigmoid để được output mong muốn.



**2.1.3 Mạng CNN**

CNN là mạng neurol được cải tiến từ perceptron nhiều lớp, bao gồm các filter để lọc dữ liệu đầu vào, thường được áp dụng trong các bài toán phân giải ảnh, quy đổi ảnh thành các dữ liệu nhỏ hơn mà không mất đi thông tin cần thiết. Về cơ bản, CNN có thể khử nhiễu đầu vào, trích xuất được đặc trưng của dữ liệu, nó thể hiện được sự hiệu quả trong việc ứng dụng vào các bài toán như xử lý ngôn ngữ, thay vì chỉ cho những bài toán thị giác máy tính như trước đây. Vì vậy, trong bài toán này, chúng ta hãy cùng thử sử dụng cả CNN để phân tích dữ liệu chuỗi thời gian để so sánh hiệu quả với LSTM.





Một số ưu điểm của CNN khi so sánh với RNN/LSTM:

* CNN tốn ít chi phí hơn về mặt tính toán, bởi vì CNN học theo lô trong khi RNN lại học tuần tự. Do đó, RNN không thể thực hiện các tác vụ song song vì phải đợi các tính toán trước đó.
* CNN không cần giả định là chuỗi thời gian đã hoàn chỉnh , vì vậy nếu dữ liệu có bị thiếu hay sai sót, CNN vẫn có thể hoạt động tốt.
* CNN có thể có những đánh giá rộng hơn, vì RNN bắt buộc phải sử dụng những dữ liệu ở trước còn CNN thì không.

**2.2 Tiền xử lý dữ liệu**

Trước khi đưa dữ liệu vào huấn luyện mô hình, chúng ta cần phải chuyển các dữ liệu thô về đúng dạng và giảm thiểu những sai sót không đáng có thông qua bước tiền xử lý dữ liệu. Việc tiền xử lý dữ liệu sẽ giúp tính toán các giá trị còn thiếu, loại bỏ ngoại lai và giảm chiều dữ liệu cần thiết. Tuy nhiên, bài toán lại có các thư viện hỗ trợ lấy dữ liệu một cách trực quan và đơn giản nhất nên dữ liệu sẵn có đã không có giá trị thiếu sót hay ngoại lai nào cả. Vì vậy, ta chỉ cần xử lý và biến đổi đơn giản để có thể một đầu vào hoàn chỉnh:

* Sử dụng MinMaxScaler để điều chỉnh dữ liệu về khoảng (0,1]
* Sử dụng cửa sổ trượt để chuyển một chuỗi dữ liệu liên tục thành các chuỗi rời rạc.
* Gán nhãn cho các chuỗi rời rạc: nhãn của mỗi chuỗi là giá trị cổ phiếu của ngày kế tiếp trong tập dữ liệu.

**2.3 Đề xuất hàm mất mát**

**2.3.1 Đặt vấn đề**

Hàm mất mát được sử dụng mặc định trong LSTM là hàm lỗi bình phương trung bình (Mean Squared Error -MSE). Tuy nhiên việc tính toán sai khác giữa giá trị dự đoán và giá trị thực tế thôi là chưa đủ, vì bên cạnh sai số đó vẫn còn một yếu tố nữa quyết định độ hiệu quả trong phán đoán giá của thị trường, đó chính là độ lên xuống, hay còn gọi là chiều biến đổi của giá cổ phiếu theo thời gian. Ví dụ ta có 2 mô hình dự đoán là model\_1 và model\_2, giá hiện tại là 100.000, giá hôm sau là 90.000, 2 model đưa ra dự đoán là 110.000 và 50.000; nếu sử dụng MSE thì rõ ràng là giá 110.000 sẽ đúng hơn giá 50.000 một cách đáng kể. Nhưng trong thực tế, mô hình đưa ra được dự đoán giá đi xuống lại còn quan trọng hơn cả việc nó đi xuống bao nhiêu, bởi vì nhà đầu tư sẽ quyết định là cắt lỗ ngay khi có dấu hiệu giảm, và đây mới là điều cần hướng đến của bài toán này trong việc hỗ trợ ra quyết định của nhà đầu tư.

**2.3.2 Đề xuất phương pháp**

Sau khi xem xét vấn đề trên, ta cần thiết kế một hàm mất mát phức tạp hơn MSE, ta cần một hàm mất mát phạt nặng hơn cho những dự đoán sai về chiều thay đổi của giá trị, đặc biệt là khi dữ liệu thực tế là giảm mà mô hình lại dự đoán nó đi lên. Từ đó ta có các bước để tính hàm mất mát như sau:

B1: Xác định các dự đoán sai về mặt xu hướng biến đổi giá cổ phiếu.

B2: Tính giá trị chênh lệch giữa dự đoán và thực tế.

B3: Nhân giá trị chênh lệch với một trọng số alpha để tăng mức độ phạt với dự đoán xu hướng sai.

**2.4 Huấn luyện mô hình**

Lần lượt thử nghiệm với các mô hình LSTM và CNN như đã đề xuất, kết hợp với hàm mất mát bổ trợ để so sánh với nhau. Huấn luyện với các tập siêu tham số khác nhau để chọn ra tập tốt nhất.

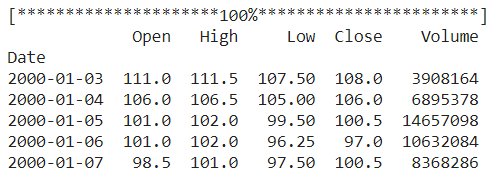
**CHƯƠNG 3: KẾT QUẢ THỰC NGHIỆM VÀ ĐÁNH GIÁ**

**3.1 Bộ dữ liệu**

Tập dữ liệu là tập các giao dịch của mã cổ phiếu 0005.HK được lấy từ thư viện yfinance, nguồn dữ liệu thư viện kéo về từ trang finance.yahoo.com.

Dataset được lấy trong giai đoạn 2000 – 2020, có tổng cộng 4922 bản ghi, chia thành 3 phần (train, valid, test) = (80,10,10)

Minh họa cho mẫu dữ liệu:





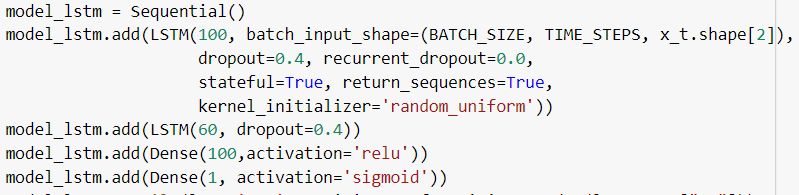
**3.2 Tiền xử lý dữ liệu**

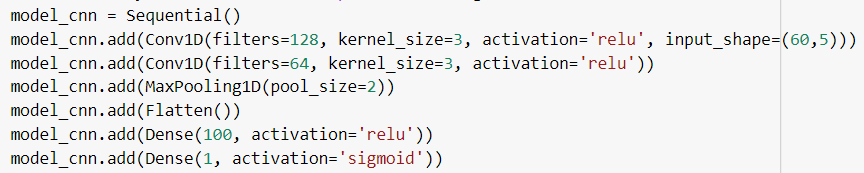
Do bộ dữ liệu được thu thập một cách tự động nên không dữ liệu không bị thiếu hay bị sai sót. Vì vậy chỉ cần tiền xử lý qua những bước đơn giản như sau:

* Đưa dữ liệu qua hàm MinMaxScaler để điều chỉnh, giảm tải tính toán khi huấn luyện.
* Đưa dữ liệu về kiểu time-series để chuẩn hóa: gộp 60 bản ghi thành 1 mẫu, tức là data sẽ mang thông tin của 60 ngày liên tiếp và có nhãn là giá trị close của ngày cuối cùng.

**3.3 Mô hình huấn luyện**

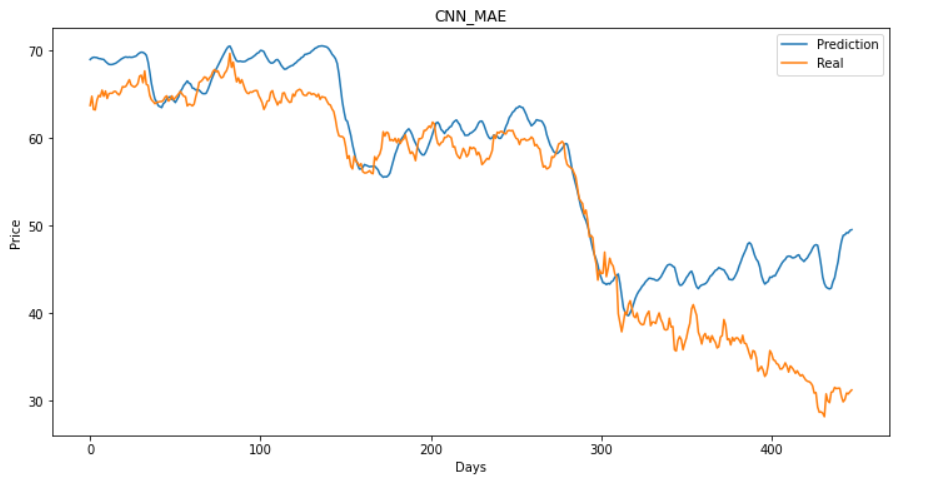
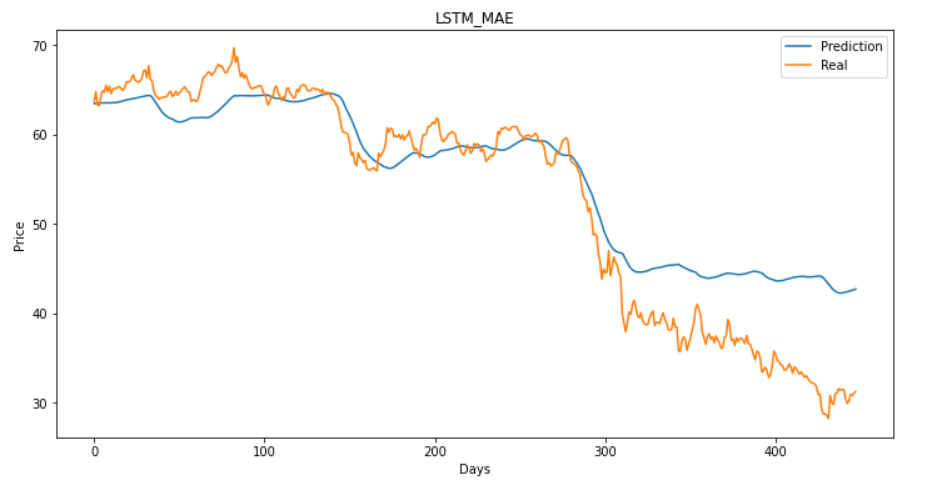
Sử dụng 2 mô hình để so sánh kết quả với nhau là LSTM và CNN:





**3.4 Kết quả nhận được**

Chương trình được huấn luyện bằng Colab: 



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | LSTM\_MAE | CNN\_MAE | LSTM\_Custom | CNN\_Custom |
| Train\_loss | 0.0402 | 0.0298 | 1.6506 | 0.9178 |
| Val\_loss | 0.0324 | 0.0196 | 0.4939 | 0.9247 |
| Test\_loss\_mae | 0.501118 | 0.494407 | 0.501118 | 0.498881 |

**CHƯƠNG 4: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN**

* Về cơ bản đã xây dựng được một chương trình giải quyết được đầy đủ các yêu cầu mà bài toán đề ra.
* Từ bảng kết quả nhận được, ta có thể thấy rằng mô hình CNN luôn nhận được kết quả chính xác tốt hơn LSTM, mặc dù chênh lệch nhau là không đáng kể. Qua nhiều lần huấn luyện, em nhận thấy kết quả của khi train cho LSTM kết hợp custom\_loss sẽ tốt hơn một chút; còn CNN thì lại không được như vậy, kết quả cuả hàm MAE vẫn ổn định hơn. Đấy là do hàm custom\_loss được thiết kế chỉ để dành cho mạng LSTM có tính chất sử dụng kết quả của các node trước làm đầu vào của nốt phía sau, cho nên custom\_loss không phù hợp với CNN.
* Từ hình ảnh visualize nhận được, hàm custom\_loss có xu hướng làm mịn các đoạn thay đổi xu hướng của giá trị cổ phiếu hơn là hàm mae; và càng về sau giá trị dự đoán càng lệch so với giá trị thực tế. Điều này chứng tỏ rằng phải cập nhật giá trị gốc thường xuyên vì mô hình không thể dự đoán được tương lai xa. Đồng thời tất cả mô hình đều có dự đoán đúng về thời điểm gái trị cổ phiếu giảm xuống, giảm sâu; điều này sẽ giúp ích rất nhiều cho các nhà đầu tư khi ra quyết định bán, giữ hay mua vào.
* Các hướng phát triển tiếp theo có thể hướng đến như là :
  + Sử dụng thử các mô hình huấn luyện chuyên biệt dành cho time series data nói chung và stock predict nói riêng như: ARIMA-LSTM, DP-LSTM, HATS,…
  + Kết hợp với một số phương pháp như nhận diện cảm xúc khi phân tích các thông tin mới về thị trường chứng khoán, ví dụ như phân tích tweet và comment của những người như Elon Musk.
  + Phân tích thử biểu đồ hình nến.

Source code: <https://github.com/ghuioio/project3>

**Tài liệu tham khảo**

[1]. <https://www.macnica.co.jp/business/ai_iot/columns/135112/>

[2]. <https://www.analyticsvidhya.com/blog/2021/08/hands-on-stock-price-time-series-forecasting-using-deep-convolutional-networks/>

[3]. <https://towardsdatascience.com/customize-loss-function-to-make-lstm-model-more-applicable-in-stock-price-prediction-b1c50e50b16c>

[4]. <https://machinelearningmastery.com/how-to-scale-data-for-long-short-term-memory-networks-in-python/?fbclid=IwAR1Imj62iSw9Uphkt39PI4kLvXobIl-brTpBJ8_EtQaa1mZAofcWtX-mtLk>

[5]. <https://machinelearningmastery.com/using-cnn-for-financial-time-series-prediction/?utm_source=drip&utm_medium=email&utm_campaign=Using+CNN+for+financial+time+series+prediction&utm_content=Using+CNN+for+financial+time+series+prediction>