**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TRƯỜNG ĐẠI HỌC CMC**

**BÁO CÁO BÀI TẬP LỚN**

**HỌC PHẦN: HỌC MÁY VÀ KHAI PHÁ DỮ LIỆU**

**Phát triển ứng dụng dự báo giá cổ phiếu dựa trên các mô hình học máy**

**Nhóm sinh viên:**

Nguyễn Phương Anh - BCS230007 - Lớp: 23CS1

Chu Thùy Dương - BCS230032 - Lớp: 23CS1

Nguyễn Thùy Dương - BCS230033 - Lớp: 23CS1

**HÀ NỘI – 2025**

**Hà Nội, tháng - 2025**



**MỤC LỤC**

[MỞ ĐẦU 2](#_Toc15050)

[PHẦN I. TỔNG QUAN 3](#_Toc31846)

[1.1. Giới thiệu bài toán 3](#_Toc7724)

[1.2. Một số thông tin liên quan 3](#_Toc1206)

[1.3. Sơ đồ tổng thể giải quyết bài toán 4](#_Toc11894)

[1.4. Kế hoạch thực hiện 4](#_Toc5709)

*[Bảng 1. Kế hoạch thực hiện](#_Toc30707)* [4](#_Toc30707)

[1.5. Kết luận 5](#_Toc28210)

[PHẦN II. PHƯƠNG PHÁP THỰC HIỆN 6](#_Toc17870)

[2.1. Đặt vấn đề 6](#_Toc11296)

[2.2. Các mô hình được sử dụng để giải quyết bài toán 6](#_Toc7469)

*[2.2.1. Hồi quy tuyến tính (Linear Regression)](#_Toc23465)* [6](#_Toc23465)

*[2.2.2. Random Forest](#_Toc12855)* [7](#_Toc12855)

*[2.2.3. Support Vector Regression (SVR)](#_Toc25529)* [7](#_Toc25529)

*[2.2.4. XGBoost (Extreme Gradient Boosting)](#_Toc11805)* [7](#_Toc11805)

*[2.2.5. LightGBM Regression](#_Toc30438)* [8](#_Toc30438)

*[2.2.6. LSTM (Long Short-Term Memory)](#_Toc5792)* [8](#_Toc5792)

[PHẦN III. THỰC NGHIỆM VÀ ĐÁNH GIÁ KẾT QUẢ 9](#_Toc13156)

[3.1. Dữ liệu thực nghiệm 9](#_Toc27314)

[3.2. Triển khai thực nghiệm 9](#_Toc22439)

[3.3. Đánh giá kết quả 20](#_Toc25884)

[PHẦN IV. KẾT LUẬN 28](#_Toc22767)

[4.1. Kết quả đạt được 28](#_Toc7504)

[4.2. Hướng phát triển tiếp theo 28](#_Toc21329)

[PHỤ LỤC 30](#_Toc16336)

# **MỞ ĐẦU**

Thị trường chứng khoán là một trong những lĩnh vực tài chính quan trọng, đóng vai trò như một kênh huy động vốn và đầu tư hiệu quả. Tuy nhiên, giá cổ phiếu luôn biến động khó lường, chịu tác động từ nhiều yếu tố như chính sách kinh tế, tình hình tài chính doanh nghiệp, xu hướng thị trường, cũng như tâm lý nhà đầu tư. Chính vì vậy, việc dự đoán giá cổ phiếu luôn là một bài toán phức tạp và có ý nghĩa thực tiễn cao đối với cả nhà đầu tư cá nhân lẫn tổ chức tài chính.

Với sự phát triển mạnh mẽ của công nghệ trí tuệ nhân tạo và học máy, các phương pháp phân tích dữ liệu truyền thống đã dần được thay thế hoặc bổ sung bằng những mô hình tiên tiến, có khả năng phát hiện các xu hướng tiềm ẩn trong dữ liệu tài chính. Các thuật toán học máy không chỉ giúp xử lý một khối lượng dữ liệu khổng lồ mà còn có khả năng học hỏi từ lịch sử biến động giá, từ đó đưa ra dự đoán có độ chính xác cao hơn so với các phương pháp phân tích kỹ thuật thông thường.

Xuất phát từ thực tế đó, đề tài “Phát triển ứng dụng dự báo giá cổ phiếu dựa trên các mô hình học máy” được thực hiện nhằm xây dựng một hệ thống hỗ trợ phân tích và dự đoán giá cổ phiếu, giúp nhà đầu tư có thêm một công cụ khoa học để đưa ra quyết định đầu tư sáng suốt hơn. Trong nghiên cứu này, nhóm nhóm sẽ ứng dụng và so sánh hiệu suất của nhiều mô hình học máy tiên tiến như Hồi quy tuyến tính (Linear Regression), Random Forest, SVR (Support Vector Regression), XGBoost, LightGBM Regression và LSTM (Long Short-Term Memory). Mỗi mô hình có những đặc điểm riêng, phù hợp với từng loại dữ liệu và mục tiêu dự báo khác nhau.

Bài báo cáo sẽ tập trung vào các nội dung chính sau:

* *Phần I - Tổng quan:* Trình bày tổng quan về bài toán dự báo giá cổ phiếu, giới thiệu các thông tin liên quan và đánh giá tầm quan trọng của đề tài.
* *Phần II – Phương pháp thực hiện:* Mô tả các phương pháp tiếp cận, bao gồm các mô hình học máy được sử dụng và cách thức triển khai để giải quyết bài toán.
* *Phần III – Thực nghiệm và đánh giá kết quả:* Cung cấp thông tin về dữ liệu thực nghiệm, quá trình triển khai và đánh giá hiệu suất của các mô hình dự báo.

Bên cạnh ý nghĩa học thuật, đề tài còn mang lại những ứng dụng thực tiễn quan trọng trong lĩnh vực tài chính. Việc sử dụng các mô hình học máy để dự báo giá cổ phiếu không chỉ giúp nhà đầu tư đưa ra quyết định hiệu quả hơn mà còn góp phần phát triển các hệ thống giao dịch thông minh, hỗ trợ quản lý rủi ro và tối ưu hóa danh mục đầu tư. Đồng thời, nghiên cứu này cũng mở ra hướng ứng dụng rộng rãi của trí tuệ nhân tạo trong lĩnh vực tài chính – một lĩnh vực có tiềm năng phát triển mạnh mẽ trong tương lai.

Với tinh thần nghiên cứu nghiêm túc và mong muốn ứng dụng công nghệ hiện đại vào thực tiễn, nhóm thực hiện đề tài hy vọng rằng báo cáo này sẽ mang đến những đóng góp giá trị trong lĩnh vực dự báo tài chính, đồng thời tạo nền tảng cho những nghiên cứu sâu hơn về ứng dụng học máy trong phân tích thị trường chứng khoán. Bên cạnh đó, do thời gian tìm hiểu và thực hiện có giới hạn, bản báo cáo đề tài không tránh khỏi những thiếu sót. Chúng em mong muốn nhận được phản hồi từ các thầy, cô để hoàn thiện đề tài cũng như góp phần hỗ trợ cho công tác phát triển, tối ưu hóa hệ thống trong tương lai.

# **PHẦN I. TỔNG QUAN**

## **1.1. Giới thiệu bài toán**

Thị trường chứng khoán là một trong những lĩnh vực tài chính có tính biến động cao, ảnh hưởng trực tiếp đến nền kinh tế và quyết định đầu tư của các tổ chức, cá nhân. Việc dự báo giá cổ phiếu chính xác có thể giúp nhà đầu tư đưa ra những quyết định đúng đắn nhằm tối ưu hóa lợi nhuận và giảm thiểu rủi ro. Tuy nhiên, giá cổ phiếu chịu tác động từ nhiều yếu tố phức tạp như tình hình kinh tế vĩ mô, chính sách tiền tệ, tâm lý thị trường, và các chỉ số tài chính của doanh nghiệp.

Trước đây, các phương pháp dự báo giá cổ phiếu chủ yếu dựa vào phân tích kỹ thuật hoặc phân tích cơ bản. Tuy nhiên, với sự phát triển của trí tuệ nhân tạo và học máy, các mô hình dữ liệu đã chứng minh khả năng dự đoán chính xác hơn nhờ khai thác mối quan hệ ẩn trong dữ liệu lịch sử. Đề tài “Phát triển ứng dụng dự báo giá cổ phiếu dựa trên các mô hình học máy” được thực hiện nhằm nghiên cứu và triển khai một hệ thống có thể hỗ trợ nhà đầu tư trong việc đưa ra quyết định dựa trên những dự đoán chính xác từ mô hình học máy.

## **1.2. Một số thông tin liên quan**

***1.2.1. Vai trò của học máy trong dự báo tài chính***

Học máy (Machine Learning) đang ngày càng khẳng định vai trò quan trọng trong lĩnh vực tài chính, đặc biệt là trong dự báo giá cổ phiếu. Nhờ khả năng phân tích dữ liệu lớn và phát hiện các mẫu ẩn trong dữ liệu lịch sử, các thuật toán học máy giúp cải thiện độ chính xác trong việc dự đoán xu hướng thị trường, hỗ trợ nhà đầu tư đưa ra quyết định hợp lý hơn.

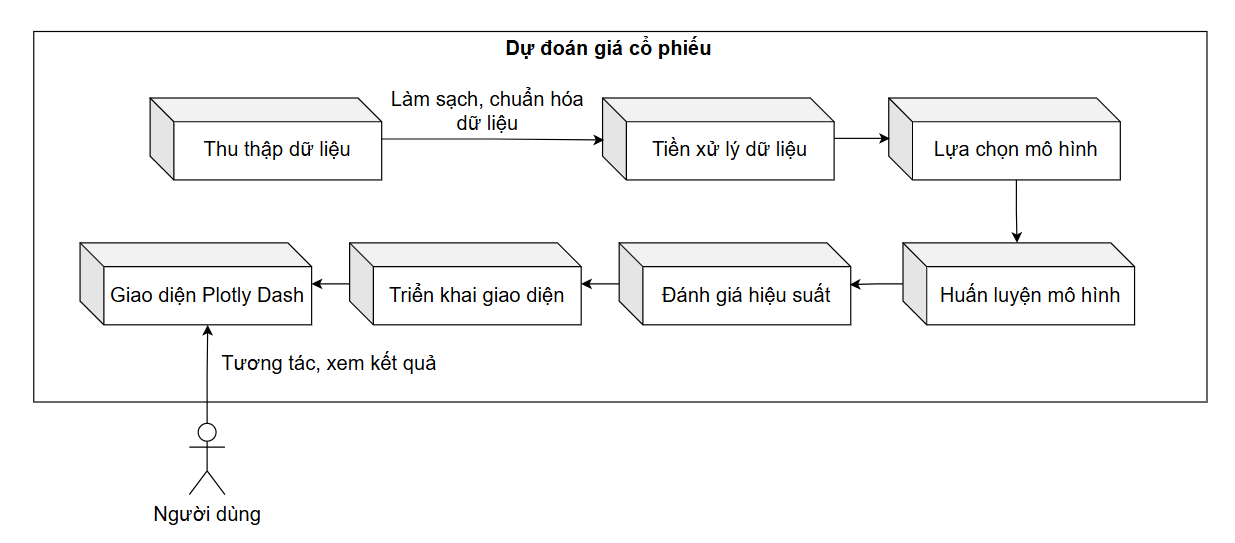
*Ứng dụng quan trọng của học máy trong tài chính:*

* Phân tích xu hướng thị trường: Học máy giúp nhận diện các mẫu dữ liệu, từ đó dự đoán xu hướng giá cổ phiếu dựa trên dữ liệu lịch sử.
* Phát hiện bất thường và gian lận: Các thuật toán học sâu có thể phát hiện các dấu hiệu bất thường trong giao dịch, giúp giảm rủi ro gian lận.
* Giao dịch thuật toán (Algorithmic Trading): Học máy giúp tạo ra các chiến lược giao dịch tự động, tận dụng các cơ hội sinh lời trên thị trường tài chính.
* Quản lý rủi ro: Các mô hình dự báo có thể ước tính rủi ro tiềm ẩn dựa trên biến động thị trường, giúp nhà đầu tư điều chỉnh danh mục đầu tư hợp lý.

***1.2.2. Các mô hình học máy trong dự báo giá cổ phiếu***

* Mô hình thống kê và hồi quy:
  + Hồi quy Ridge và Lasso (Ridge Regression, Lasso Regression): Các biến thể nâng cao của hồi quy tuyến tính giúp giảm thiểu hiện tượng đa cộng tuyến và chọn lọc đặc trưng quan trọng.
* Mô hình cây quyết định và Boosting:
  + Random Forest: Mô hình cây quyết định tổng hợp, giúp giảm thiểu hiện tượng quá khớp.
  + Gradient Boosting Machines (GBM): Một thuật toán tăng cường (Boosting) mạnh mẽ, có khả năng học tốt từ dữ liệu phi tuyến tính.
  + XGBoost: Mô hình tối ưu hóa Boosting, có khả năng dự báo chính xác với tốc độ cao.
  + CatBoost: Tối ưu hóa việc xử lý dữ liệu phân loại, giúp tăng tốc độ học và cải thiện độ chính xác.
* Mô hình mạng nơ ron sâu (Deep Learning):
  + LSTM (Long Short-Term Memory): Mô hình nơ-ron hồi quy (RNN) chuyên xử lý chuỗi thời gian, giúp dự đoán giá cổ phiếu dựa trên dữ liệu lịch sử dài hạn.
  + GRU (Gated Recurrent Unit): Một biến thể của LSTM, có khả năng ghi nhớ thông tin trong chuỗi thời gian với số lượng tham số ít hơn, giúp giảm chi phí tính toán.
  + Transformer-based models: Các mô hình như BERT hoặc Attention Mechanism đang được áp dụng trong tài chính để trích xuất thông tin quan trọng từ tin tức tài chính và dữ liệu phi cấu trúc.

## **1.3. Sơ đồ tổng thể giải quyết bài toán**



*Hình 1. Sơ đồ tổng thể giải quyết bài toán*

## **1.4. Kế hoạch thực hiện**

***Bảng 1. Kế hoạch thực hiện***

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Nội dung thực hiện | Sản phẩm cần đạt | Người phụ trách | Ghi chú |
| 1 | Xác định yêu cầu của bài toán | Hiểu rõ bài toán dự báo giá cổ phiếu, xác định các yếu tố ảnh hưởng | Cả nhóm | Phân tích đặc điểm dữ liệu tài chính, mục tiêu dự báo |
| 2 | Phân tích và lựa chọn phương pháp tối ưu | Chọn mô hình học máy phù hợp với bài toán | Nguyễn Phương Anh | Đánh giá các phương pháp như Linear Regression, Random Forest, XGBoost, LSTM,... |
| 3 | Xây dựng ý tưởng cho ứng dụng | Mô tả ý tưởng thiết kế ứng dụng và các tính năng chính | Chu Thùy Dương | Định hình giao diện, chức năng và cách triển khai thuật toán |
| 4 | Xử lý dữ liệu (Thu thập, chuẩn hóa, trích xuất đặc trưng) | Bộ dữ liệu được xử lý, sẵn sàng đưa vào mô hình | Nguyễn Thùy Dương | Làm sạch dữ liệu, chuẩn hóa giá trị, xử lý dữ liệu thời gian |
| 5 | Huấn luyện và tinh chỉnh mô hình | Mô hình dự báo có độ chính xác cao, được tối ưu hóa | Cả nhóm | Điều chỉnh tham số (hyperparameter tuning) để tối ưu hóa mô hình |
| 6 | Đánh giá và cải tiến mô hình | Báo cáo kết quả, cải thiện mô hình nếu cần thiết | Cả nhóm | Sử dụng các chỉ số đánh giá như MAE, RMSE, R² |
| 7 | Tổng hợp kết quả và viết báo cáo | Báo cáo chi tiết quá trình thực hiện và kết quả đạt được | Cả nhóm | Trình bày rõ ràng về phương pháp, kết quả và nhận xét |
| 8 | Làm PowerPoint thuyết trình | Bài thuyết trình rõ ràng, minh họa từng bước triển khai | Nguyễn Phương Anh | Chuẩn bị slide chuyên nghiệp, dễ hiểu, có biểu đồ minh họa |

## **1.5. Kết luận**

Bài toán dự báo giá cổ phiếu có ý nghĩa quan trọng trong lĩnh vực tài chính, đặc biệt là trong việc hỗ trợ nhà đầu tư đưa ra quyết định giao dịch. Học máy đã mở ra nhiều hướng tiếp cận mới trong dự báo giá cổ phiếu, giúp tăng độ chính xác và giảm thiểu rủi ro. Việc lựa chọn mô hình phù hợp phụ thuộc vào tính chất dữ liệu, nhu cầu thực tế và khả năng tính toán.

Trong các phần tiếp theo, nhóm sẽ trình bày chi tiết về phương pháp thực hiện, các mô hình được sử dụng và kết quả thực nghiệm để đánh giá mức độ hiệu quả của hệ thống dự báo.

# **PHẦN II. PHƯƠNG PHÁP THỰC HIỆN**

## **2.1. Đặt vấn đề**

Trong bài toán dự báo giá cổ phiếu, việc lựa chọn phương pháp phù hợp đóng vai trò quan trọng trong việc đảm bảo độ chính xác và khả năng áp dụng của mô hình vào thực tế. Để đạt được kết quả tốt nhất, nhóm đã tiếp cận vấn đề theo hướng kết hợp các mô hình học máy (Machine Learning) và học sâu (Deep Learning) nhằm tận dụng ưu điểm của từng phương pháp.

Các mô hình được lựa chọn dựa trên các tiêu chí sau:

* *Khả năng xử lý dữ liệu chuỗi thời gian:* Vì giá cổ phiếu có tính liên tục và phụ thuộc vào dữ liệu lịch sử, các mô hình như ARIMA, LSTM và GRU được xem xét để nắm bắt xu hướng và mối quan hệ giữa các mốc thời gian.
* *Khả năng học dữ liệu phi tuyến tính:* Thị trường chứng khoán bị ảnh hưởng bởi nhiều yếu tố khác nhau, do đó các mô hình mạnh như XGBoost, LightGBM và Transformer được lựa chọn để phát hiện các mối quan hệ phức tạp trong dữ liệu.
* *Tốc độ và hiệu suất:* Vì dự báo giá cổ phiếu yêu cầu tính toán nhanh và chính xác để hỗ trợ quyết định đầu tư, nhóm ưu tiên các mô hình có tốc độ xử lý cao và hiệu suất tốt.

Trên cơ sở đó, nhóm đã thử nghiệm và lựa chọn một số mô hình phù hợp để so sánh, đánh giá và tối ưu nhằm tìm ra phương pháp dự báo hiệu quả nhất.

## **2.2. Các mô hình được sử dụng để giải quyết bài toán**

Trong nghiên cứu này, nhóm đã áp dụng và so sánh hiệu suất của nhiều mô hình học máy phổ biến nhằm đánh giá khả năng dự báo giá cổ phiếu. Các mô hình này thuộc các nhóm khác nhau, từ mô hình thống kê, cây quyết định, mô hình dựa trên SVM, cho đến các thuật toán Boosting và mạng nơ-ron nhân tạo.

### ***2.2.1. Hồi quy tuyến tính (Linear Regression)***

1. *Nguyên lý hoạt động*

Hồi quy tuyến tính là một mô hình thống kê cơ bản sử dụng phương pháp tối ưu bình phương tối thiểu để tìm ra đường thẳng phù hợp nhất biểu diễn mối quan hệ giữa biến độc lập (dữ liệu đầu vào) và biến phụ thuộc (giá cổ phiếu).

Phương trình hồi quy tuyến tính có dạng:

*Y = β0 + β1X1 + β2X2 + … + βnXn + ϵ*

trong đó:

* *Y* là giá cổ phiếu dự đoán.
* *X1,* *X2,..., Xn* là các biến đầu vào (ví dụ: giá mở cửa, khối lượng giao dịch, chỉ số thị trường…)
* *β0* là hệ số chặn (Intercept).
* *β1,* *β2,..., βn* là các trọng số của mô hình.
* *ϵ* là sai số.

1. *Ưu điểm*

* Đơn giản, dễ triển khai và giải thích.
* Hiệu quả khi dữ liệu có mối quan hệ tuyến tính mạnh.

1. *Nhược điểm*

* Hiệu suất kém khi dữ liệu có tính phi tuyến tính cao.
* Dễ bị ảnh hưởng bởi các ngoại lệ trong dữ liệu.

1. *Ứng dụng trong dự báo giá cổ phiếu*

Mô hình được sử dụng để thiết lập một đường cơ sở nhằm so sánh với các mô hình phức tạp hơn. Nếu giá cổ phiếu có xu hướng tuyến tính, mô hình này có thể đưa ra dự đoán khá chính xác.

### ***2.2.2. Random Forest***

1. *Nguyên lý hoạt động*

Random Forest là một mô hình tập hợp (ensemble learning) dựa trên nhiều cây quyết định (Decision Trees). Nó hoạt động bằng cách:

* Chia tập dữ liệu thành nhiều tập con ngẫu nhiên.
* Xây dựng nhiều cây quyết định độc lập trên các tập con đó.
* Kết hợp kết quả từ tất cả các cây bằng cách trung bình hóa (đối với bài toán hồi quy).

1. *Ưu điểm*

* Giảm thiểu hiện tượng quá khớp (overfitting) so với một cây quyết định đơn lẻ.
* Xử lý tốt dữ liệu có tính phi tuyến tính.
* Hiệu suất tốt ngay cả khi có nhiều biến đầu vào.

1. *Nhược điểm*

* Tốc độ chậm hơn so với các mô hình tuyến tính khi số lượng cây lớn.
* Khó giải thích hơn so với hồi quy tuyến tính.

1. *Ứng dụng trong dự báo giá cổ phiếu*

Random Forest có thể nắm bắt được các mối quan hệ phi tuyến giữa các biến số, giúp cải thiện độ chính xác khi dự báo giá cổ phiếu.

### ***2.2.3. Support Vector Regression (SVR)***

1. *Nguyên lý hoạt động*

SVR là một phiên bản của Support Vector Machine (SVM) áp dụng cho bài toán hồi quy. Mô hình tìm cách tối ưu hóa một hàm hồi quy sao cho sai số nhỏ nhất nhưng vẫn giữ được tính tổng quát hóa của mô hình.

1. *Ưu điểm*

* Hiệu quả với dữ liệu có quan hệ phi tuyến tính phức tạp.
* Khả năng tổng quát hóa tốt với dữ liệu chưa từng thấy.

1. *Nhược điểm*

* Tốc độ huấn luyện chậm khi làm việc với tập dữ liệu lớn.
* Khó điều chỉnh các siêu tham số (hyperparameters).

1. *Ứng dụng trong dự báo giá cổ phiếu*

SVR đặc biệt hữu ích khi dự báo giá cổ phiếu trong những trường hợp thị trường có biến động mạnh và không thể mô hình hóa bằng các phương pháp tuyến tính.

### ***2.2.4. XGBoost (Extreme Gradient Boosting)***

1. *Nguyên lý hoạt động*

XGBoost là một thuật toán Boosting, hoạt động bằng cách xây dựng một tập hợp các cây quyết định nhỏ, trong đó mỗi cây mới sẽ tập trung vào việc sửa lỗi của cây trước đó.

1. *Ưu điểm*

* Hiệu suất cao với tốc độ học nhanh.
* Kiểm soát tốt hiện tượng overfitting bằng cách sử dụng regularization.
* Hoạt động tốt với cả dữ liệu tuyến tính và phi tuyến tính.

1. *Nhược điểm*

Khó điều chỉnh các siêu tham số để đạt hiệu suất tối ưu.

1. *Ứng dụng trong dự báo giá cổ phiếu*

XGBoost được sử dụng để tìm ra các mẫu phức tạp trong dữ liệu giá cổ phiếu, giúp cải thiện độ chính xác của dự báo.

### ***2.2.5. LightGBM Regression***

1. *Nguyên lý hoạt động*

LightGBM là một biến thể của XGBoost, sử dụng phương pháp "leaf-wise growth" thay vì "level-wise growth", giúp mô hình học nhanh hơn và chính xác hơn.

1. *Ưu điểm*

* Tốc độ huấn luyện nhanh hơn XGBoost, đặc biệt với tập dữ liệu lớn.
* Khả năng tổng quát hóa tốt, hạn chế overfitting.

1. *Nhược điểm*

Cần tinh chỉnh các siêu tham số để đạt hiệu suất cao nhất.

1. *Ứng dụng trong dự báo giá cổ phiếu*

LightGBM phù hợp với các tập dữ liệu lớn, giúp dự đoán nhanh chóng và chính xác.

### ***2.2.6. LSTM (Long Short-Term Memory)***

1. *Nguyên lý hoạt động*

LSTM là một biến thể của mạng nơ-ron hồi quy (RNN) được thiết kế để xử lý dữ liệu chuỗi thời gian. Nó có cơ chế "bộ nhớ" giúp giữ lại thông tin quan trọng trong khoảng thời gian dài.

1. *Ưu điểm*

* Hiệu quả trong việc xử lý dữ liệu chuỗi thời gian dài hạn.
* Có khả năng ghi nhớ các xu hướng quan trọng trong dữ liệu giá cổ phiếu.

1. *Nhược điểm*

* Yêu cầu nhiều tài nguyên tính toán.
* Dễ bị overfitting nếu không có đủ dữ liệu.

1. *Ứng dụng trong dự báo giá cổ phiếu*

LSTM được sử dụng để dự báo giá cổ phiếu dựa trên dữ liệu lịch sử dài hạn, giúp nhận diện các xu hướng dài hạn trong thị trường chứng khoán.

**2.3. Kết luận**

Nhóm đã triển khai nhiều mô hình học máy khác nhau để dự báo giá cổ phiếu, mỗi mô hình có ưu điểm và nhược điểm riêng. Trong phần tiếp theo, nhóm sẽ tiến hành thực nghiệm, so sánh hiệu suất của các mô hình dựa trên các chỉ số đánh giá như MAE (Mean Absolute Error).

# **PHẦN III. THỰC NGHIỆM VÀ ĐÁNH GIÁ KẾT QUẢ**

## **3.1. Dữ liệu thực nghiệm**

***3.1.1. Giới thiệu tập dữ liệu***

Trong nghiên cứu này, nhóm sử dụng tập dữ liệu Stock Market Dataset từ Kaggle, một bộ dữ liệu chứa thông tin lịch sử về giá cổ phiếu của nhiều công ty trên thị trường chứng khoán. Đây là nguồn dữ liệu đáng tin cậy, được thu thập từ các sàn giao dịch lớn như NASDAQ, NYSE, S&P 500, giúp đảm bảo tính chính xác và phản ánh đầy đủ xu hướng thị trường.

***3.1.2. Lý do lựa chọn tập dữ liệu***

Tập dữ liệu được lựa chọn dựa trên các tiêu chí sau:

* *Độ tin cậy cao:* Dữ liệu được tổng hợp từ các sàn giao dịch lớn, đảm bảo tính chính xác và phù hợp cho các mô hình dự báo tài chính.
* *Bao quát nhiều yếu tố thị trường:* Bộ dữ liệu cung cấp thông tin quan trọng như giá mở cửa, giá đóng cửa, khối lượng giao dịch, giúp mô hình có cái nhìn toàn diện về sự biến động của cổ phiếu.
* *Phù hợp với mô hình học máy:* Tập dữ liệu có cấu trúc rõ ràng, dễ dàng xử lý và áp dụng vào các thuật toán như hồi quy tuyến tính, Random Forest, SVR, XGBoost, LightGBM, LSTM…

***3.1.3. Các thành phần trong tập dữ liệu***

Bộ dữ liệu bao gồm các thông tin quan trọng về giao dịch cổ phiếu:

***Bảng 2. Các thành phần trong tập dữ liệu***

|  |  |
| --- | --- |
| Cột dữ liệu | Mô tả |
| Date | Ngày giao dịch |
| Open | Giá mở cửa của cổ phiếu vào đầu phiên giao dịch |
| High | Giá cao nhất đạt được trong ngày |
| Low | Giá thấp nhất trong ngày |
| Close | Giá đóng cửa của cổ phiếu vào cuối phiên giao dịch |
| Adj Close | Giá đóng cửa đã điều chỉnh, phản ánh chính xác hơn giá trị thực của cổ phiếu sau khi tính đến cổ tức, chia tách cổ phiếu hoặc các yếu tố khác. |
| Volume | Tổng khối lượng cổ phiếu được giao dịch trong ngày |

Sau khi thu thập, dữ liệu sẽ được xử lý trước khi đưa vào huấn luyện mô hình.

## **3.2. Triển khai thực nghiệm**

Trong phần này, nhóm tiến hành xây dựng mô hình dự báo giá cổ phiếu dựa trên bộ dữ liệu Stock Market Dataset từ Kaggle. Nhóm thử nghiệm nhiều phương pháp học máy, từ mô hình thống kê đến các thuật toán Boosting và mạng nơ-ron nhân tạo, sau đó tập trung vào giải thích Random Forest Regressor. Các mô hình khác sẽ được trình bày tổng quan trong phần này, song chi tiết mã nguồn sẽ có trong Phụ Lục.

***3.2.1. Giới thiệu phương pháp***

Nhóm sử dụng mô hình Random Forest Regressor, một thuật toán học máy dựa trên tập hợp nhiều cây quyết định (Decision Trees) để dự báo giá cổ phiếu của công ty HP Inc. (HPQ).

Mô hình được tối ưu hóa bằng GridSearchCV để tìm ra tham số tốt nhất, từ đó tăng độ chính xác trong dự báo. Nhóm cũng sử dụng các đặc trưng tài chính như giá mở cửa, giá cao nhất, giá thấp nhất, khối lượng giao dịch, đường trung bình động (MA20), và độ biến động giá để cải thiện hiệu suất mô hình.

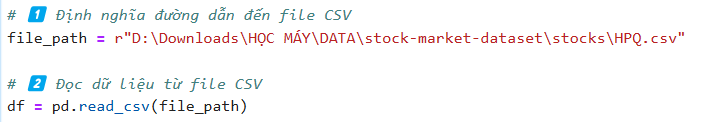
Bên cạnh Random Forest, nhóm cũng thử nghiệm và so sánh với các mô hình khác, bao gồm Linear Regression, Support Vector Regression (SVM), LSTM (Long Short-Term Memory), XGBoost và LightGBM. Mỗi mô hình đều được khởi tạo với tham số phù hợp để đảm bảo đánh giá chính xác hiệu suất trên cùng một tập dữ liệu. Việc thử nghiệm này giúp nhóm hiểu rõ ưu, nhược điểm của từng thuật toán, từ đó chọn ra mô hình tối ưu nhất. Toàn bộ mã nguồn sẽ được trình bày chi tiết trong phần Phụ Lục.

***3.2.2. Quy trình triển khai mô hình Random Forest***

1. *Bước 1: Đọc dữ liệu từ nguồn*

Dữ liệu được lấy từ một tập tin CSV chứa thông tin về giá cổ phiếu HP Inc. (HPQ).





*Hình 2. Mã nguồn hiển thị bước đọc dữ liệu từ nguồn*

1. *Bước 2: Tiền xử lý dữ liệu*

Trước khi huấn luyện mô hình, dữ liệu cần được làm sạch và tiền xử lý để đảm bảo chất lượng đầu vào tốt nhất. Các bước tiền xử lý bao gồm:

* Chuyển đổi định dạng cột Date và sắp xếp dữ liệu theo thời gian:
  + Cột Date thường có dạng chuỗi, cần chuyển thành kiểu dữ liệu datetime để dễ thao tác.
  + Dữ liệu cần được sắp xếp theo ngày để đảm bảo tính liên tục và chính xác,
* Xử lý dữ liệu bị thiếu (Missing Values):



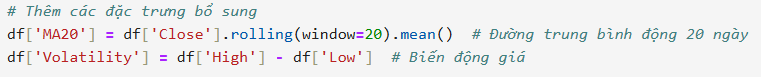
*Hình 3. Mã nguồn hiển thị quy trình xử lý dữ liệu bị thiếu*

* Tạo cột đặc trưng Days:
  + Mô hình học máy cần dữ liệu dạng số, do đó có thể thay thế Date bằng số ngày kể từ ngày đầu tiên trong tập dữ liệu.



*Hình 4. Mã nguồn hiển thị bước tạo cột đặc trưng cho dữ liệu*

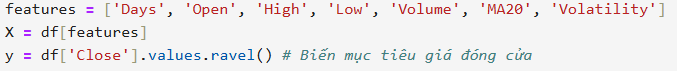
* Tạo các đặc trưng bổ sung (Feature Engineering):
  + Đường trung bình động 20 ngày (MA20): Giúp nắm bắt xu hướng thị trường trong 20 ngày gần nhất.
  + Biến động giá (Volatility): Chênh lệch giữa giá cao nhất và thấp nhất trong ngày.



*Hình 5. Mã nguồn hiển thị bước tạo các đặc trưng bổ sung*

1. *Bước 3: Lựa chọn đặc trưng và chia tập dữ liệu*

* Chọn các đặc trưng quan trọng:
  + Chỉ chọn các cột có giá trị quan trọng để huấn luyện mô hình.
  + Tránh đưa vào dữ liệu không có ý nghĩa hoặc trùng lặp.



*Hình 6. Mã nguồn hiển thị bước chọn các đặc trưng quan trọng*

* Chia dữ liệu thành tập huấn luyện và tập kiểm tra:
  + Tập huấn luyện (train): 80% dữ liệu để huấn luyện mô hình.
  + Tập kiểm tra (test): 20% dữ liệu để đánh giá mô hình.



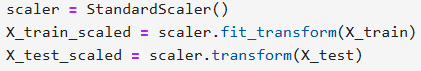


*Hình 7. Mã nguồn hiển thị bước chia tập dữ liệu*

1. *Bước 4: Chuẩn hóa dữ liệu (Feature Scaling)*

* Các đặc trưng có đơn vị đo lường khác nhau (Volume rất lớn so với Days, Open, Close…), nên cần chuẩn hóa dữ liệu.
* Sử dụng StandardScaler để đưa dữ liệu về phân phối chuẩn (trung bình = 0, độ lệch chuẩn = 1).





*Hình 8. Mã nguồn hiển thị bước chuẩn hóa dữ liệu*

1. *Bước 5: Huấn luyện mô hình Random Forest*

Nhóm thực hiện tối ưu hóa mô hình Random Forest bằng cách sử dụng GridSearchCV – một phương pháp tìm kiếm tham số tốt nhất thông qua kiểm tra nhiều kết hợp khác nhau của tham số mô hình. Dưới đây là giải thích chi tiết từng bước:

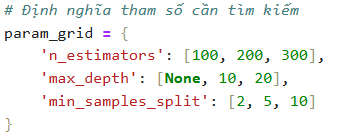
* Khởi tạo mô hình Random Forest:





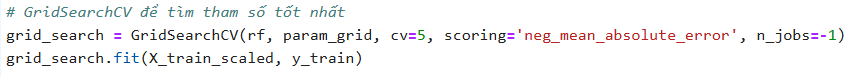
*Hình 9. Mã nguồn hiển thị bước khởi tạo mô hình Random Forest*

* + *RandomForestRegressor* được khởi tạo với *random\_state=42* để đảm bảo tính tái lập khi chạy lại chương trình.
  + Đây là mô hình mặc định trước khi tiến hành điều chỉnh tham số (hyperparameter tuning).
* Xác định tập hợp tham số cần tối ưu:



*Hình 10. Mã nguồn hiển thị bước xác định các tham số*

* + *n\_estimators:* Số lượng cây trong rừng. Các giá trị thử nghiệm là 100, 200 và 300.
  + *max\_depth:* Độ sâu tối đa của mỗi cây quyết định. None cho phép cây phát triển tự do, còn giá trị 10 và 20 giới hạn độ sâu nhằm tránh overfitting.
  + *min\_samples\_split:* Số lượng mẫu tối thiểu cần có để chia một node. Giá trị 2, 5 và 10 được thử nghiệm để kiểm tra mức độ tổng quát hóa của mô hình.
* Tìm kiếm tham số tối ưu bằng GridSearchCV:



*Hình 11. Mã nguồn hiển thị bước tìm kiếm tham số tối ưu bằng GridSearchCV*

* + GridSearchCV thực hiện thử nghiệm tất cả các kết hợp tham số trong *param\_grid* và đánh giá hiệu suất dựa trên tiêu chí đã chọn.
  + *cv=5*: Chia tập dữ liệu thành 5 phần để thực hiện kiểm tra chéo (cross-validation), giúp đảm bảo kết quả không bị phụ thuộc vào cách chia dữ liệu.
  + *scoring='neg\_mean\_absolute\_error'*: Chọn thước đo sai số trung bình tuyệt đối (MAE) làm tiêu chí tối ưu.
  + *n\_jobs=-1*: Sử dụng tất cả các nhân xử lý của CPU để tăng tốc độ tính toán.
* Lấy mô hình có hiệu suất tốt nhất:



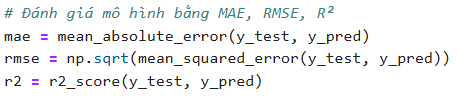
*Hình 12. Mã nguồn hiển thị bước lấy mô hình có hiệu suất tốt nhất*

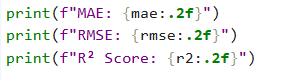
* + *grid\_search.best\_estimator\_* trả về mô hình Random Forest có tổ hợp tham số tốt nhất, giúp tăng độ chính xác và giảm sai số trong dự đoán.

1. *Bước 6: Dự đoán giá cổ phiếu và đánh giá mô hình*

Sau khi huấn luyện, nhóm tiến hành dự đoán và đánh giá mô hình bằng MAE, RMSE, R² Score.







*Hình 13. Mã nguồn hiển thị bước dự đoán*

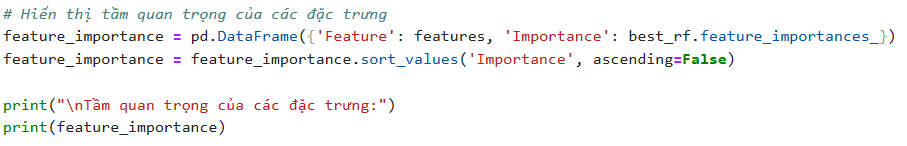
1. *Bước 7: Hiển thị tầm quan trọng của các đặc trưng*

*Mục đích:*

* Xác định các đặc trưng quan trọng:
  + Random Forest có khả năng đánh giá tầm quan trọng của từng đặc trưng bằng cách đo lường mức độ đóng góp của chúng vào quá trình dự đoán.
  + Điều này giúp hiểu rõ yếu tố nào ảnh hưởng lớn nhất đến giá cổ phiếu, từ đó có thể tập trung vào những đặc trưng quan trọng nhất.
* Loại bỏ các đặc trưng không quan trọng:
  + Nếu có những đặc trưng có mức độ quan trọng thấp, ta có thể loại bỏ chúng để giảm độ phức tạp của mô hình, giúp mô hình tổng quát hóa tốt hơn và tránh overfitting.
* Cung cấp thông tin hữu ích cho phân tích tài chính:
  + Bằng cách xem xét các đặc trưng ảnh hưởng đến giá cổ phiếu, ta có thể đưa ra những đánh giá tài chính quan trọng, chẳng hạn như mức độ ảnh hưởng của biến động giá, khối lượng giao dịch hay xu hướng thị trường.

*Mã nguồn hiển thị tầm quan trọng của các đặc trưng:*





*Hình 14. Mã nguồn hiển thị tầm quan trọng của các đặc trưng*

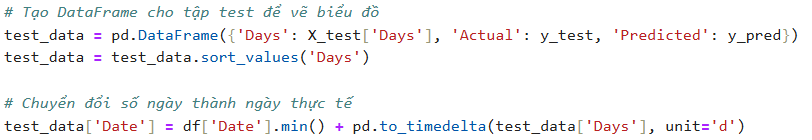
* *best\_rf.feature\_importances\_*: Trả về một danh sách chứa tầm quan trọng của từng đặc trưng trong dự đoán.
* Tạo DataFrame *feature\_importance*: Biến này chứa thông tin về tên đặc trưng và giá trị quan trọng tương ứng.
* Sắp xếp theo thứ tự giảm dần: Giúp dễ dàng nhận biết đặc trưng nào có ảnh hưởng lớn nhất đến mô hình.

1. *Bước 8: Hiển thị tầm quan trọng của các đặc trưng*

Nhóm sử dụng Plotly để vẽ biểu đồ so sánh giá thực tế và giá dự báo.



* Tạo DataFrame chứa dữ liệu tập test để vẽ biểu đồ:



*Hình 15. Mã nguồn hiển thị bước tạo dataframe chứa tập dữ liệu tập test để vẽ biểu đồ*

* + Mục đích: Tạo một DataFrame mới chứa thông tin ngày (Days), giá thực tế (Actual) và giá dự đoán (Predicted).
  + Sắp xếp theo thời gian: Sắp xếp dữ liệu theo số ngày (Days) để đảm bảo biểu đồ hiển thị theo trình tự thời gian.
* Chuyển đổi *Days* thành ngày thực tế:



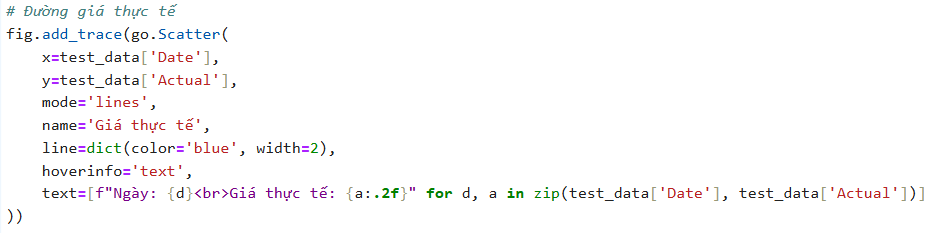
*Hình 16. Mã nguồn hiển thị bước chuyển đổi số ngày thành ngày thực tế*

* + *df['Date'].min()*: Lấy ngày đầu tiên trong tập dữ liệu ban đầu.
  + *pd.to\_timedelta(test\_data['Days'], unit='d')*: Chuyển đổi số ngày thành khoảng thời gian thực tế kể từ ngày đầu tiên.
  + Mục đích: Giúp biểu đồ hiển thị trục thời gian theo ngày thực tế thay vì số ngày từ điểm mốc.
* Vẽ biểu đồ dự báo bằng Plotly:



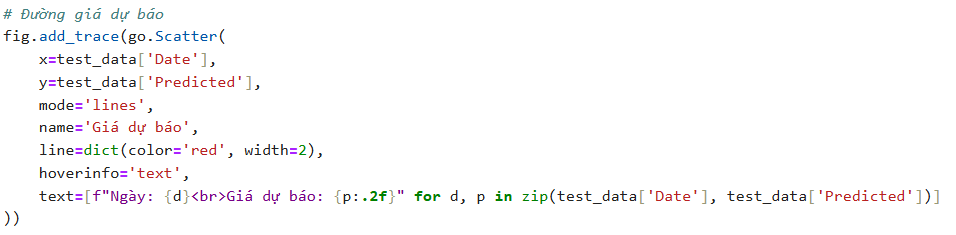
*Hình 17. Mã nguồn hiển thị bước vẽ biểu đồ dự đoán*

* + *go.Figure():* Khởi tạo một biểu đồ trống trong thư viện Plotly.
  + Thêm đường giá thực tế (màu xanh):



*Hình 18. Mã nguồn hiển thị bước vẽ đường giá thực tế*

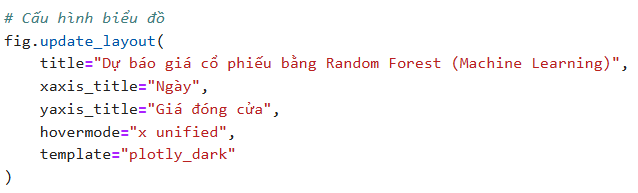
* + - *go.Scatter():* Dùng để vẽ biểu đồ đường.
    - *x=test\_data['Date']:* Trục x là ngày thực tế.
    - *y=test\_data['Actual']:* Trục y là giá thực tế.
    - *mode='lines':* Hiển thị dạng đường (line chart).
    - *name='Giá thực tế':* Nhãn hiển thị trong chú thích.
    - *line=dict(color='blue', width=2):* Định dạng đường màu xanh và độ dày là 2px.
    - *hoverinfo='text':* Hiển thị thông tin khi di chuột vào điểm dữ liệu.
    - *text=[f"Ngày: {d}<br>Giá thực tế: {a:.2f}" for d, a in zip(test\_data['Date'], test\_data['Actual'])]*: Tạo nội dung hiển thị khi hover (ngày và giá thực tế).
  + Thêm đường giá dự báo (màu đỏ):



*Hình 19. Mã nguồn hiển thị bước vẽ đường giá dự báo*

Tương tự như đường giá thực tế, nhưng:

* + - Dữ liệu trục y thay bằng *test\_data['Predicted']* (giá dự báo).
    - Màu sắc đổi thành đỏ (*color='red'*) để phân biệt với đường giá thực tế.
    - Nhãn hiển thị khi hover chứa thông tin giá dự báo.
* Cấu hình biểu đồ:



*Hình 20. Mã nguồn hiển thị bước cấu hình biểu đồ*

* + *title*: Tiêu đề biểu đồ.
  + *xaxis\_title="Ngày"*: Nhãn trục x.
  + *yaxis\_title="Giá đóng cửa"*: Nhãn trục y.
  + *hovermode="x unified"*: Khi di chuột, cả hai đường (giá thực tế & dự báo) hiển thị thông tin cùng lúc.
  + *template="plotly\_dark"*: Sử dụng giao diện tối của Plotly để biểu đồ trông chuyên nghiệp hơn.
* Hiển thị biểu đồ:



*Hình 21. Mã nguồn hiển thị bước hiển thị biểu đồ*

***3.2.3. Quy trình triển khai mô hình còn lại***

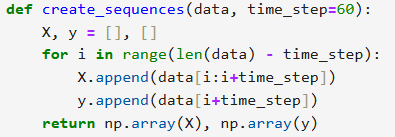
Về cơ bản, quy trình triển khai mô hình còn lại tương tự như Random Forest, bao gồm tiền xử lý dữ liệu, chia tập huấn luyện - kiểm tra và chuẩn hóa dữ liệu. Tuy nhiên, điểm khác biệt chính nằm ở cấu trúc mạng nơ-ron và các thông số khởi tạo của mô hình. Trong phần này, nhóm sẽ trình bày nhóm sẽ trình bày quy trình triển khai mô hình dự báo giá cổ phiếu bằng các phương pháp Machine Learning và Deep Learning, đặc biệt là các tham số quan trọng của từng mô hình để làm rõ cách tiếp cận và tối ưu hóa dự báo giá cổ phiếu.

1. *Huấn luyện mô hình*

* Các mô hình Machine Learning (Linear Regression, SVR, XGBoost, LightGBM) được khởi tạo với các tham số mặc định và huấn luyện trên tập huấn luyện.
* Với mô hình LSTM (một mạng nơ-ron hồi tiếp - RNN), dữ liệu đầu vào cần được chuyển đổi thành dạng chuỗi thời gian phù hợp.

*Xử lý dữ liệu cho mô hình LSTM:*

* Tạo chuỗi dữ liệu thời gian:



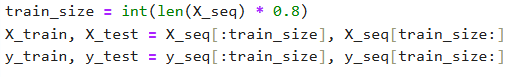
*Hình 22. Mã nguồn hiển thị bước tạo chuỗi dữ liệu thời gian của LSTM*

* + Hàm *create\_sequences* để tạo dữ liệu đầu vào.
  + Vòng lặp *for i in range(len(data) - time\_step)* chạy qua toàn bộ tập dữ liệu.
  + Tại mỗi vị trí *i*, ta lấy *time\_steps* ngày trước đó làm đầu vào *X*.
  + Ngày tiếp theo sau *time\_steps* ngày đó được dùng làm nhãn *y*.
  + Cuối cùng, ta chuyển đổi *X* và *y* thành mảng numpy để đưa vào mô hình.
* Áp dụng vào tập dữ liệu:



*Hình 23. Mã nguồn hiển thị bước sử dụng chuỗi dữ liệu thời gian của LSTM*

* Chia tập dữ liệu thành tập huấn luyện và kiểm tra:

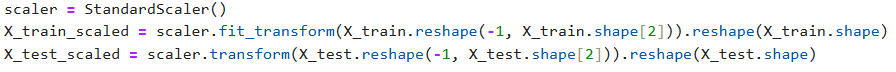


*Hình 24. Mã nguồn hiển thị bước chia tập dữ liệu của LSTM*

* + 80% dữ liệu được sử dụng để huấn luyện, 20% còn lại dùng để kiểm tra.
* Chuẩn hóa dữ liệu:

LSTM yêu cầu đầu vào có dạng *(samples, time steps, features)*, nhưng dữ liệu *X\_train* và *X\_test* ban đầu có dạng *(samples, time steps)*. Vì vậy, ta cần chuẩn hóa dữ liệu và reshape lại:





*Hình 25. Mã nguồn hiển thị bước chuẩn hóa dữ liệu của LSTM*

* + *StandardScaler* chuẩn hóa dữ liệu để có phân phối trung bình 0 và độ lệch chuẩn 1.
  + *X\_train.reshape(-1, X\_train.shape[2]):*
    - *X\_train* ban đầu có dạng *(samples, time\_steps, features).*
    - *reshape(-1, X\_train.shape[2])* chuyển đổi thành dạng 2D: *(total\_samples, features)*, giúp *StandardScaler* có thể áp dụng trên từng đặc trưng (feature).
    - *-1* nghĩa là tự động tính toán số hàng *(total\_samples)*.
  + *scaler.fit\_transform(...):*
    - Áp dụng chuẩn hóa dữ liệu: tính mean (trung bình) và std (độ lệch chuẩn) trên tập huấn luyện, sau đó chuẩn hóa giá trị.
  + *.reshape(X\_train.shape):*
    - Sau khi chuẩn hóa, nhóm reshape dữ liệu trở lại dạng 3D *(samples, time\_steps, features)* để phù hợp với LSTM.
  + Áp dụng chuẩn hóa lên tập kiểm tra, không dùng *fit\_transform* mà chỉ dùng *transform*:
    - *scale.fit\_transform(...)* chỉ dùng cho tập huấn luyện để tính toán mean và std.
    - Khi áp dụng lên tập kiểm tra, nhóm chỉ dùng *scale.transform(...)* để giữ nguyên các thông số chuẩn hóa từ tập dữ liệu, tránh data leakage.
    - Quy trình trương tự như với *X\_train\_scaled*:

Reshape về 2D → Chuẩn hóa bằng StandardScaler → Reshape về 3D

* Ngăn overfitting với Early Stopping:

Để tránh overfitting, nhóm sử dụng Early Stopping, giúp dừng huấn luyện khi lỗi trên tập kiểm tra không cải thiện sau một số epoch nhất định:



**

*Hình 26. Mã nguồn hiển thị bước thêm early stopping của LSTM*

* + *monitor='val\_loss'*: Theo dõi lỗi trên tập kiểm tra.
  + *patience=10*: Dừng huấn luyện nếu lỗi không giảm sau 10 epoch.
  + *restore\_best\_weights=True*: Khôi phục trọng số tốt nhất của mô hình.

1. *Thông số quan trọng của từng mô hình*

* *Linear Regression (Hồi quy tuyến tính):*

**

*Hình 27. Mã nguồn hiển thị bước khởi tạo mô hình Linear Regression*

* + Không có tham số điều chỉnh quan trọng vì đây là mô hình tuyến tính đơn giản.
  + Chủ yếu phụ thuộc vào dữ liệu đã được chuẩn hóa và phân phối của biến đầu vào.
* *SVR (Support Vector Regression):*

**

*Hình 28. Mã nguồn hiển thị bước khởi tạo mô hình SVR*

* + *kernel='rbf'* → Sử dụng hàm kernel RBF (Radial Basis Function) để tìm mối quan hệ phi tuyến tính giữa các đặc trưng và giá cổ phiếu.
  + *C* (mặc định) → Tham số kiểm soát độ phạt của mô hình khi có sai số. Giá trị lớn sẽ làm mô hình ít bị lỗi nhưng dễ bị overfitting.
* *XGBoost (Extreme Gradient Boosting):*

**

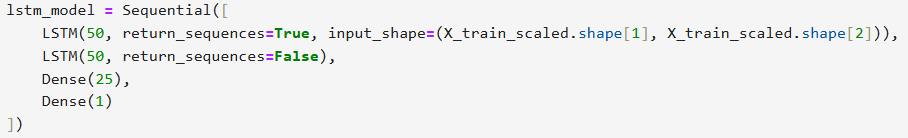
*Hình 29. Mã nguồn hiển thị bước khởi tạo mô hình XGBoost*

* + *n\_estimators=100* → Số lượng cây quyết định trong mô hình (tăng số cây giúp mô hình học tốt hơn nhưng tăng thời gian huấn luyện).
  + *learning\_rate=0.1* → Tốc độ học, nếu quá lớn có thể làm mô hình không hội tụ, nếu quá nhỏ sẽ mất nhiều thời gian huấn luyện.
  + *random\_state=42* → Đảm bảo kết quả tái lập khi chạy lại mô hình.
* *LightGBM (Light Gradient Boosting Machine):*

**

*Hình 30. Mã nguồn hiển thị bước khởi tạo mô hình LightGBM*

* + *n\_estimators=100* → Số lượng cây quyết định, tương tự XGBoost.
  + *learning\_rate=0.1* → Tốc độ học của mô hình.
  + LightGBM nhanh hơn XGBoost nhờ thuật toán histogram-based learning giúp tối ưu tốc độ huấn luyện.
* *LSTM (Long Short-Term Memory - Deep Learning):*

**

*Hình 31. Mã nguồn hiển thị bước khởi tạo mô hình LSTM*

* + *LSTM(50, return\_sequences=True, input\_shape=(X\_train\_lstm.shape[1], 1))* → 50 đơn vị LSTM, return\_sequences=True để sử dụng đầu ra của lớp này làm đầu vào cho lớp LSTM tiếp theo.
  + *Dense(25)* → Lớp fully connected với 25 neuron để trích xuất đặc trưng.
  + *Dense(1)* → Lớp đầu ra, dự đoán giá cổ phiếu.
  + *optimizer='adam'* → Sử dụng Adam Optimizer giúp tối ưu tốc độ hội tụ.
  + *loss='mean\_squared\_error'* → Hàm mất mát MSE giúp giảm sai số giữa giá dự báo và giá thực tế.
  + *batch\_size=16* → Kích thước batch giúp cập nhật trọng số sau mỗi 16 mẫu.
  + *epochs=20* → Số vòng lặp huấn luyện trên toàn bộ tập dữ liệu.

***3.2.4. Giao diện hiển thị dự báo giá cổ phiếu***

Hệ thống sử dụng Plotly để trực quan hóa dữ liệu dự báo giá cổ phiếu, giúp người dùng dễ dàng theo dõi xu hướng thị trường. Biểu đồ cung cấp các tính năng tương tác, bao gồm:

* *Chức năng Zoom:*
  + Người dùng có thể phóng to hoặc thu nhỏ một vùng dữ liệu cụ thể bằng cách kéo chuột hoặc sử dụng các nút "Zoom in" và "Zoom out".
  + Chức năng này hỗ trợ phân tích chi tiết diễn biến giá cổ phiếu trong một khoảng thời gian cụ thể.
* *Chức năng Pan (Di chuyển biểu đồ):*
  + Cho phép di chuyển biểu đồ theo chiều ngang và dọc để quan sát dữ liệu ở các giai đoạn khác nhau.
  + Hữu ích khi người dùng muốn so sánh biến động giá cổ phiếu giữa các giai đoạn lịch sử khác nhau.
* *Chức năng Auto Scale (Tự động điều chỉnh tỷ lệ):*
  + Khi người dùng thay đổi phạm vi hiển thị, hệ thống có thể tự động điều chỉnh tỷ lệ trục X và Y để biểu đồ hiển thị rõ ràng hơn.
* *Chức năng Reset Axes (Khôi phục trục tọa độ mặc định):*
  + Nếu người dùng đã thay đổi tỷ lệ hoặc vị trí biểu đồ, họ có thể sử dụng chức năng này để đưa biểu đồ về trạng thái mặc định ban đầu.
* *Chức năng Download Plot as PNG (Tải biểu đồ dưới dạng ảnh PNG):*
  + Cho phép người dùng lưu biểu đồ dưới dạng hình ảnh chất lượng cao để sử dụng trong báo cáo hoặc nghiên cứu cá nhân.
* *Chức năng Hover (Hiển thị chi tiết khi di chuột vào biểu đồ)*
  + Khi di chuột vào các điểm dữ liệu trên biểu đồ, người dùng có thể xem:
    - Giá trị thực tế và giá dự báo
    - Thời gian tương ứng với điểm dữ liệu
    - Tỷ lệ phần trăm sai lệch giữa giá thực tế và dự báo
  + Điều này giúp người dùng đánh giá trực quan độ chính xác của mô hình.

*Giao diện hiển thị:*



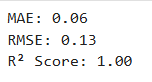
*Hình 32. Giao diện dự báo giá cổ phiếu*

## **3.3. Đánh giá kết quả**

***3.3.1. Kết quả mô hình Random Forest***

Sau khi huấn luyện mô hình Random Forest để dự báo giá cổ phiếu, chúng ta đánh giá hiệu suất của mô hình dựa trên ba chỉ số phổ biến:

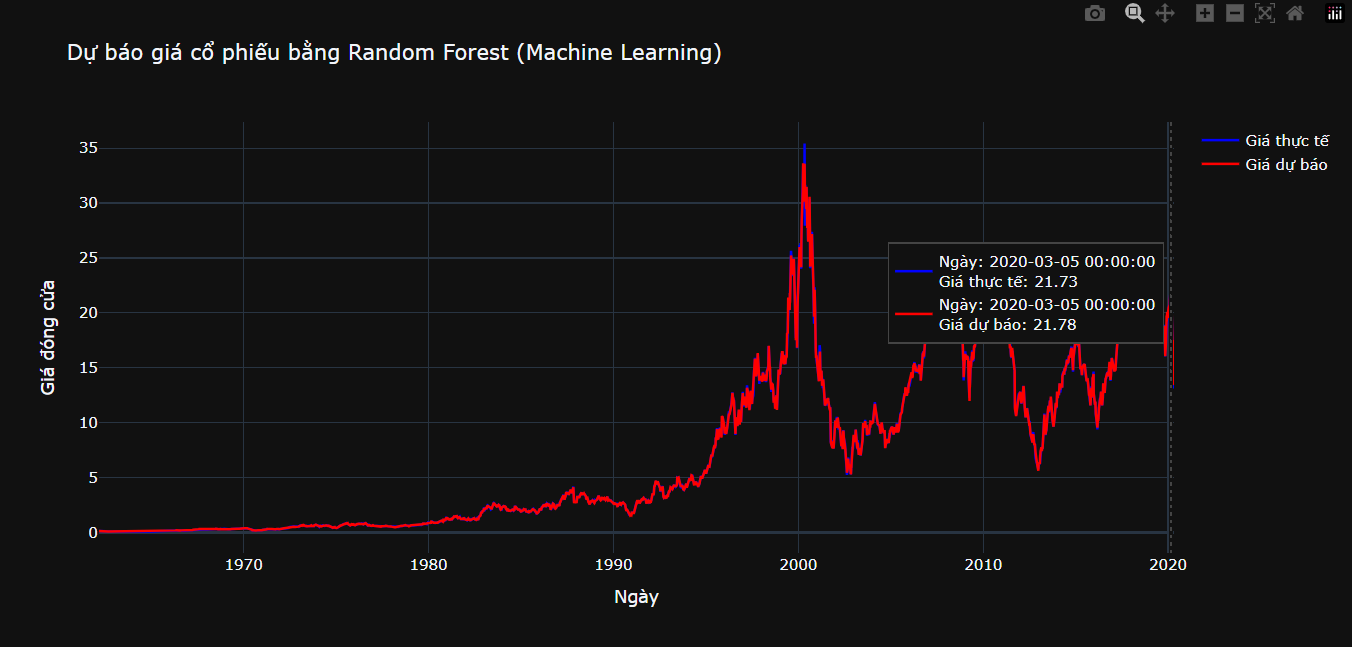
* MAE (Mean Absolute Error - Sai số trung bình tuyệt đối): 0.06
* RMSE (Root Mean Squared Error - Căn bậc hai sai số trung bình bình phương): 0.13
* R² Score (Hệ số xác định): 1.00



*Hình 33. Kết quả mô hình Random Forest*

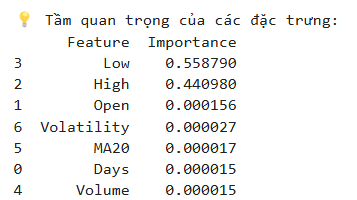
*Ý nghĩa các chỉ số đánh giá:*

* MAE = 0.06 → Mô hình dự đoán sai số trung bình khoảng 0.06 đơn vị so với giá thực tế.
* RMSE = 0.13 → RMSE cao hơn MAE một chút, cho thấy sai số có thể bị ảnh hưởng bởi một số dự đoán cực trị.
* R² Score = 1.00 → Mô hình dự đoán hoàn toàn khớp với dữ liệu, nghĩa là nó giải thích được 100% phương sai của dữ liệu đầu vào. Kết quả này chỉ thực sự tốt nếu dữ liệu được phân chia rõ ràng và không có vấn đề overfitting.



*Hình 34. Giao diện dự báo giá cổ phiếu bằng Random Forest*

*Phân tích tầm quan trọng của các đặc trưng:*

**

*Hình 35. Tầm quan trọng của các đặc trưng*

* Low (0.558790) và High (0.440980) là hai đặc trưng quan trọng nhất, chiếm hơn 99% tổng tầm quan trọng của mô hình.
  + Điều này hợp lý vì giá đóng cửa thường phụ thuộc vào giá cao nhất và thấp nhất trong ngày.
* Các đặc trưng khác (Open, Volatility, MA20, Days, Volume) có tầm quan trọng rất nhỏ (~0.000015 - 0.000156), nghĩa là mô hình hầu như không sử dụng các thông tin này.

***3.3.2. Kết quả mô hình Linear Regression***

******

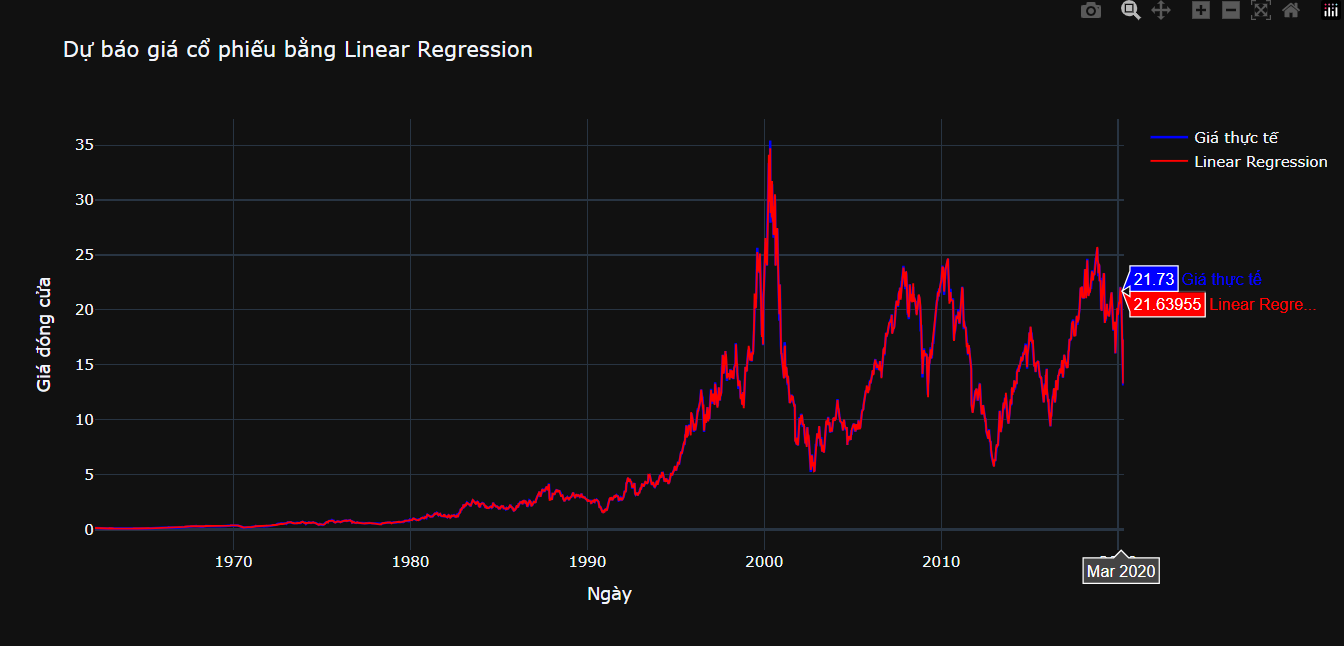
*Hình 36. Kết quả mô hình Linear Regression*

*Độ sai số thấp:*

* MAE (Mean Absolute Error) = 0.05 → Sai số trung bình chỉ khoảng 0.05 đơn vị so với giá thực tế. Đây là một mức sai số rất nhỏ, cho thấy mô hình dự đoán khá chính xác.
* RMSE (Root Mean Squared Error) = 0.09 → Giá trị này cao hơn MAE một chút, nhưng vẫn rất thấp, chứng tỏ mô hình hoạt động tốt mà không bị ảnh hưởng nhiều bởi các giá trị cực đoan.

*Độ khớp hoàn hảo:*

* R² Score = 1.00 → Hệ số xác định R² đạt giá trị tuyệt đối (1.00), cho thấy mô hình Linear Regression khớp hoàn toàn với dữ liệu huấn luyện và kiểm tra. Điều này có nghĩa là mô hình có thể giải thích toàn bộ phương sai trong tập dữ liệu mà không để lại phần dư sai số. Kết quả này chỉ thực sự tốt nếu dữ liệu được phân chia rõ ràng và không có vấn đề overfitting.



*Hình 37. Kết quả dự báo giá cổ phiếu bằng Linear Regression*

***3.3.3. Kết quả mô hình SVR***



*Hình 38. Kết quả mô hình SVR*

* MAE = 0.11 → Mức sai số trung bình là 0.11, cao hơn so với Random Forest (0.06) và Linear Regression (0.05), cho thấy SVR có thể chưa tối ưu bằng mô hình tuyến tính trên tập dữ liệu này.
* RMSE = 0.51 → Giá trị RMSE khá cao so với MAE, cho thấy mô hình có thể gặp phải một số dự đoán lệch xa giá trị thực.
* R² = 1.00 → R² đạt 1.00, nghĩa là mô hình khớp hoàn toàn với dữ liệu huấn luyện, tương tự Linear Regression và Random Forest.

**

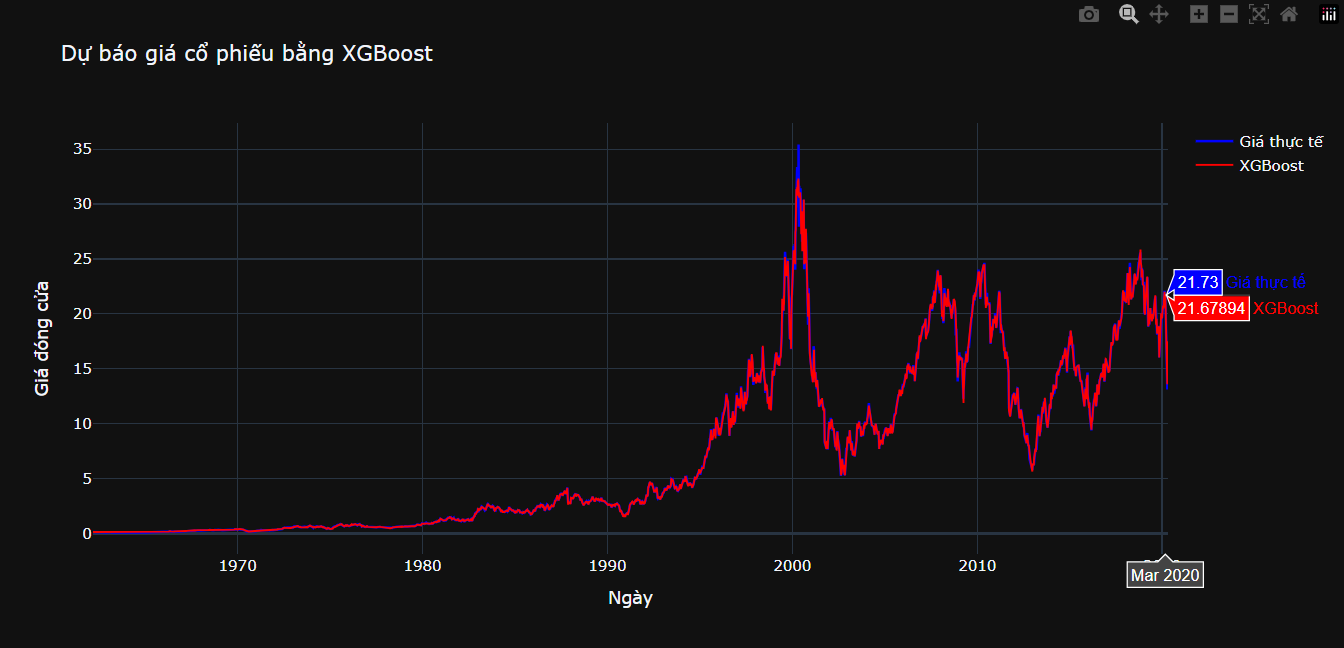
*Hình 39. Kết quả dự báo giá cổ phiếu bằng SVR*

***3.3.4. Kết quả mô hình XGBoost***

******

*Hình 40. Kết quả mô hình XGBoost*

* MAE = 0.07 → Mức sai số trung bình 0.07, thấp hơn so với SVR (0.11) nhưng cao hơn Linear Regression (0.05) và Random Forest (0.06).
* RMSE = 0.19 → Giá trị RMSE vẫn thấp, nhưng cao hơn Linear Regression (0.09) và Random Forest (0.13), cho thấy XGBoost có thể có một số dự đoán lệch hơn.
* R² = 1.00 → Giống như các mô hình khác, XGBoost đạt độ phù hợp hoàn hảo với tập huấn luyện. Kết quả này chỉ thực sự tốt nếu dữ liệu được phân chia rõ ràng và không có vấn đề overfitting.

******

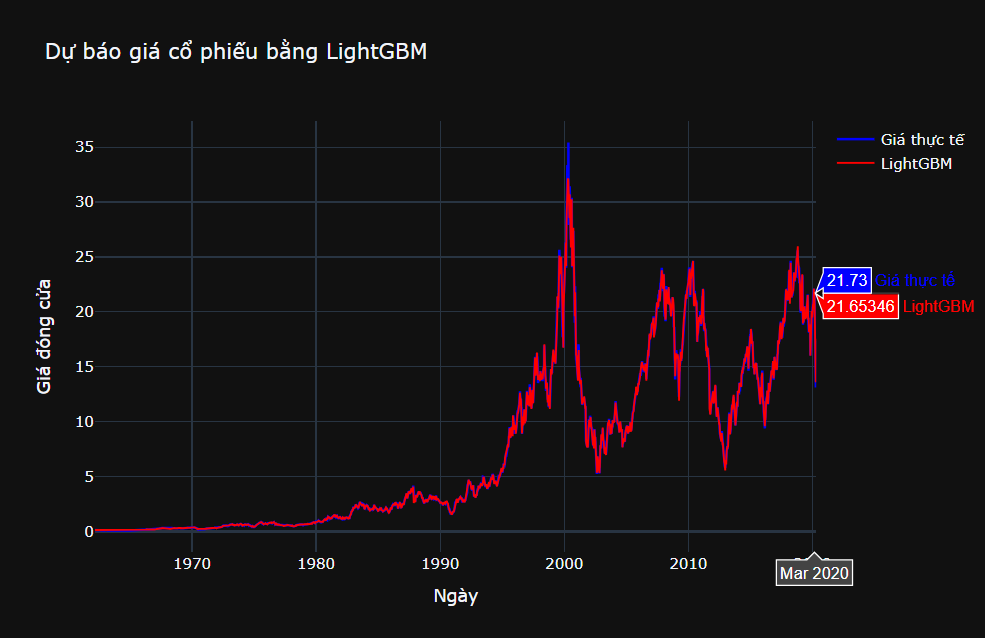
*Hình 41. Kết quả dự báo giá cổ phiếu bằng XGBoost*

***3.3.5. Kết quả mô hình mô hình LightGBM***

******

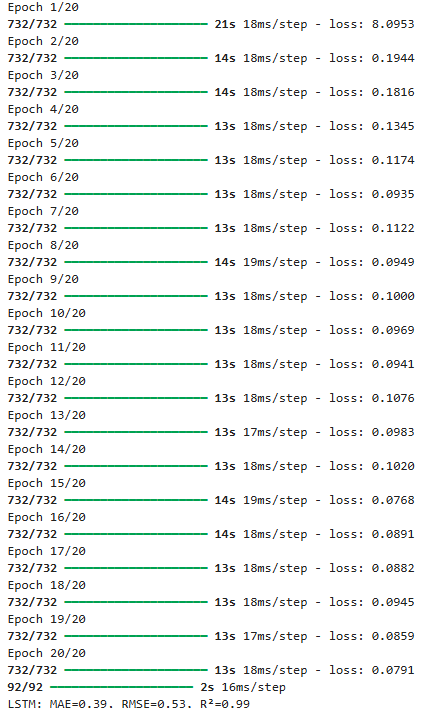
*Hình 42. Kết quả mô hình LightGBM*

* MAE = 0.07 → Sai số trung bình tương tự XGBoost (0.07), thấp hơn SVR (0.11) nhưng cao hơn Linear Regression (0.05) và Random Forest (0.06).
* RMSE = 0.17 → Thấp hơn XGBoost (0.19) và SVR (0.51), nhưng vẫn cao hơn Linear Regression (0.09) và Random Forest (0.13).
* R² = 1.00 → Mô hình dự đoán hoàn toàn khớp với dữ liệu, nghĩa là nó giải thích được 100% phương sai của dữ liệu đầu vào. Kết quả này chỉ thực sự tốt nếu dữ liệu được phân chia rõ ràng và không có vấn đề overfitting.

******

*Hình 43. Kết quả dự báo giá cổ phiếu bằng LightGBM*

***3.3.6. Kết quả mô hình mô hình LSTM***

******

*Hình 44. Kết quả mô hình LSTM*

* Loss ban đầu = 8.0953 (Epoch 1) → Rất cao, nhưng giảm mạnh ở epoch 2 (0.1944).
* Loss tiếp tục giảm và dao động nhẹ từ epoch 6-20 → Có sự hội tụ nhưng không hoàn toàn ổn định.
* Loss thấp nhất = 0.0768 (Epoch 15) → Sau đó có xu hướng tăng nhẹ, dấu hiệu có thể cần tinh chỉnh thêm.
* LSTM có R² = 0.99, vẫn rất cao nhưng thấp hơn các mô hình khác (R² = 1.00).
* RMSE = 0.53, MAE = 0.39, cao hơn so với các mô hình khác → Dự đoán kém chính xác hơn.
* Dấu hiệu overfitting: Trên tập huấn luyện loss thấp, nhưng trên tập test lỗi khá cao.

******

*Hình 45. Kết quả dự báo giá cổ phiếu bằng LSTM*

***3.3.7. So sánh kết quả các mô hình***

Bảng dưới đây tổng hợp kết quả đánh giá hiệu suất của từng mô hình:

|  |  |  |  |
| --- | --- | --- | --- |
| Mô hình | MAE (↓) | RMSE (↓) | R² Score (↑) |
| Linear Regression | 0.05 | 0.09 | 1.00 |
| **Random Forest** | **0.06** | **0.13** | **1.00** |
| XGBoost | 0.07 | 0.19 | 1.00 |
| LightGBM | 0.07 | 0.17 | 1.00 |
| SVR | 0.11 | 0.51 | 1.00 |
| LSTM | 0.39 | 0.53 | 0.99 |

* Linear Regression có hiệu suất tốt nhất với MAE thấp nhất (0.05) và RMSE thấp nhất (0.09), đồng thời đạt R² = 1.00.
  + Tuy nhiên, Linear Regression giả định mối quan hệ tuyến tính giữa các biến, điều này có thể khiến mô hình hoạt động kém trên dữ liệu phi tuyến.
  + Do đó, mặc dù có kết quả tốt, nhưng mô hình này có thể thiếu khả năng tổng quát hóa khi áp dụng vào dữ liệu thực tế.
* Random Forest cũng hoạt động tốt, có MAE = 0.06, RMSE = 0.13, chỉ kém một chút so với Linear Regression.
  + Khả năng xử lý dữ liệu phi tuyến tốt nhờ việc kết hợp nhiều cây quyết định.
  + Chống overfitting tốt hơn so với các mô hình cây quyết định đơn lẻ nhờ cơ chế bỏ phiếu trung bình.
  + Do đó, Random Forest là lựa chọn phù hợp nhất để đảm bảo độ chính xác và khả năng tổng quát hóa.
* XGBoost và LightGBM có độ chính xác tương đương, nhưng RMSE cao hơn Linear Regression, cho thấy mức độ sai số lớn hơn.
* SVR có RMSE rất cao (0.51), cho thấy mô hình này có thể không phù hợp với bài toán hiện tại, có thể phù hợp hơn với các bài toán hồi quy phức tạp hơn khi có sự điều chỉnh tham số thích hợp.
* LSTM có hiệu suất kém nhất, với MAE = 0.39 và RMSE = 0.53, đồng thời R² giảm xuống 0.99, có thể do chưa được tinh chỉnh tốt hoặc không phù hợp với loại dữ liệu này.

→ Dựa trên các kết quả so sánh, nhóm quyết định chọn *Random Forest* là mô hình tối ưu nhất vì:

* *Độ chính xác cao:* MAE và RMSE thấp, gần với Linear Regression nhưng có tính linh hoạt hơn.
* *Khả năng tổng quát hóa tốt:* Hoạt động hiệu quả trên cả dữ liệu tuyến tính và phi tuyến.
* *Độ ổn định cao:* Không dễ bị overfitting như một số mô hình Boosting khác.
* *Không yêu cầu tinh chỉnh nhiều tham số:* So với XGBoost hoặc LightGBM, Random Forest có thể đạt hiệu suất tốt mà không cần tối ưu hóa quá nhiều.

Tóm lại, Random Forest cân bằng giữa độ chính xác, độ ổn định và khả năng tổng quát hóa, làm cho nó trở thành lựa chọn tối ưu nhất trong bài toán này.

# **PHẦN IV. KẾT LUẬN**

Trong nghiên cứu này, nhóm đã tìm hiểu và ứng dụng các mô hình học máy để giải quyết bài toán dự báo giá cổ phiếu. Quá trình nghiên cứu không chỉ tập trung vào việc lựa chọn mô hình phù hợp mà còn bao gồm nhiều giai đoạn quan trọng nhằm đảm bảo độ chính xác và tính ổn định của dự báo.

## **4.1. Kết quả đạt được**

* *Nghiên cứu và triển khai mô hình học máy:* Nhóm đã tìm hiểu sâu về các mô hình học máy khác nhau, từ các phương pháp truyền thống như Hồi quy tuyến tính (Linear Regression) đến các mô hình tiên tiến hơn như Random Forest, XGBoost, LightGBM, SVR và LSTM. Việc thử nghiệm nhiều mô hình giúp đánh giá toàn diện khả năng dự báo của từng phương pháp và chọn ra mô hình phù hợp nhất.
* *Tiền xử lý và chuẩn hóa dữ liệu:* Chất lượng dữ liệu đầu vào đóng vai trò quan trọng trong hiệu suất của mô hình. Do đó, nhóm đã tiến hành làm sạch dữ liệu, xử lý giá trị bị thiếu, loại bỏ nhiễu và chuẩn hóa các đặc trưng để đảm bảo tính đồng nhất và phù hợp với yêu cầu của từng mô hình. Các kỹ thuật như Min-Max Scaling và Standardization đã được áp dụng để cải thiện độ chính xác của dự báo.
* *Tối ưu hóa tham số:* Để đảm bảo hiệu suất tốt nhất cho từng mô hình, nhóm đã thực hiện điều chỉnh tham số (hyperparameter tuning) bằng các phương pháp như Grid Search và Random Search. Quá trình này giúp xác định các tham số tối ưu, tăng cường khả năng dự đoán và giảm thiểu sai số.
* *Giao diện hiển thị kết quả với Plotly:* Biểu đồ trực quan có giao diện tối giản, hiện đại, dễ nhìn, với màu sắc rõ ràng để phân biệt giá thực tế và giá dự báo. Người dùng có thể tùy chỉnh các thông số và tương tác linh hoạt để phân tích dữ liệu một cách hiệu quả.
* *Đánh giá hiệu suất:* Các mô hình được đánh giá dựa trên các chỉ số quan trọng như MAE (Mean Absolute Error), RMSE (Root Mean Squared Error) và R² Score. Kết quả thực nghiệm cho thấy Random Forest có hiệu suất tốt nhất với độ chính xác cao, sai số thấp và khả năng tổng quát hóa tốt. Điều này chứng minh rằng phương pháp ensemble learning như Random Forest có thể mang lại dự báo ổn định và đáng tin cậy trong bài toán này.

## **4.2. Hướng phát triển tiếp theo**

Dù đã đạt được những kết quả đáng kể, nghiên cứu vẫn còn nhiều tiềm năng phát triển và mở rộng nhằm nâng cao hiệu suất dự báo cũng như ứng dụng thực tiễn trong lĩnh vực tài chính.

* *Mở rộng dữ liệu đầu vào:* Hiện tại, mô hình chủ yếu dựa trên dữ liệu lịch sử giá cổ phiếu. Trong tương lai, có thể tích hợp thêm các yếu tố ảnh hưởng như dữ liệu vĩ mô (lạm phát, lãi suất, GDP), dữ liệu tin tức tài chính, phân tích cảm xúc thị trường (sentiment analysis) từ mạng xã hội và báo chí để tăng độ chính xác của dự báo.
* *Nâng cao mô hình dự báo:* Bên cạnh Random Forest, các phương pháp kết hợp như ensemble learning (Bagging, Boosting) hoặc hybrid models có thể được áp dụng để cải thiện hiệu suất. Đồng thời, việc thử nghiệm các kiến trúc mạng nơ-ron tiên tiến như Transformer hoặc Attention-based models có thể giúp nâng cao khả năng dự báo của mô hình dựa trên dữ liệu chuỗi thời gian.
* *Ứng dụng thực**tế:* Kết quả của nghiên cứu có thể được ứng dụng vào việc xây dựng hệ thống hỗ trợ ra quyết định đầu tư, giúp nhà đầu tư cá nhân và tổ chức có công cụ dự báo đáng tin cậy. Một hệ thống dự báo thời gian thực có thể được phát triển dưới dạng ứng dụng hoặc nền tảng trực tuyến, tích hợp với công cụ phân tích tài chính để cung cấp dự báo chính xác theo thời gian thực.

Nghiên cứu này không chỉ khẳng định tính ứng dụng của học máy trong dự báo tài chính mà còn mở ra nhiều hướng đi mới để cải thiện độ chính xác và độ tin cậy của mô hình. Trong tương lai, việc tiếp tục mở rộng dữ liệu, cải tiến thuật toán và ứng dụng thực tiễn sẽ giúp mô hình ngày càng hoàn thiện hơn, góp phần hỗ trợ các quyết định đầu tư hiệu quả hơn.

# **PHỤ LỤC**

1. Github: https://github.com/chuchubeingchaotic/StockPricePrediction
2. Bộ dữ liệu: https://www.kaggle.com/datasets/jacksoncrow/stock-market-dataset