Tập 14-Số 4 (2025)

Tạp chí Khoa học công nghệ Giao thông vận tải
Journal of Transportation Science and Technology
Trang tạp chí: https://tapchicongnghegtvt.vn

Nâng cao khả năng dự báo giá cổ phiếu bằng mô hình học sâu Mamba
Improving Stock Price Forecasting with Mamba Deep Learning Model

Trần Thế Vinh[1] , Nguyễn Thị Khánh Tiên[1,*], Trần Kim Thanh[1], Phạm Trần Nhật Linh[1], Mai Quỳnh Trâm[1]

[1]*Viện công nghệ thông tin và Điện, điện tử, Trường Đại học Giao thông vận tải Thành phố Hồ Chí Minh*

Từ khóa:

Dự đoán giá cổ phiếu

Mô hình Mamba

Mô hình dãy trại tháng có chọn lọc

Giảm tốc độ học

Hàm mất mát

TÓM TẮT

Dự báo giá chứng khoán là một thách thức quan trọng trong phân tích tài chính, đòi hỏi các mô hình có khả năng xử lý hiệu quả dữ liệu chuỗi thời gian và đảm bảo hiệu suất tính toán. Nghiên cứu này tập trung phát triển một mô hình học sâu dựa trên kiến trúc Mamba, sử dụng mô hình Không gian Trạng thái Chọn lọc để xử lý dữ liệu hiệu quả, cải thiện độ chính xác dự báo giá chứng khoán, đặc biệt trong thị trường biến động. Mô hình này đồng thời tận dụng các đặc trưng của Mamba như Kernel Fusion, Parallel Scan và Recomputation trong như tối ưu hóa phần cứng để giảm thiểu tài nguyên tính toán. Quá trình huấn luyện mô hình đã được áp dụng kỹ thuật tối ưu hoá riêng. Kết quả thực nghiệm cho thấy mô hình này phù hợp với các ứng dụng tài chính thời gian thực và mở ra tiềm năng phát triển mới.

Keywords:

Stock price forecasting

Mamba model

Selective State Space Model

Learning rate decay

Loss function

ABSTRACT

Stock price forecasting is an important challenge in financial analysis, requiring models that can efficiently process time series data and ensure computational efficiency. This study focuses on developing a deep learning model based on the Mamba architecture, using the Selective State Space model to efficiently process data, improving the accuracy of stock price forecasting, especially in volatile markets. This model also takes advantage of Mamba's features such as Kernel Fusion, Parallel Scan and Recomputation while optimizing hardware to minimize computational resources. The model training process has been applied with a unique optimization technique. Experimental results show that this model is suitable for real-time financial applications and opens up new development potential.

*Nguyễn Thị Khánh Tiên. Viện công nghệ thông tin và Điện, điện tử, Trường Đại học Giao thông vận tải Thành phố Hồ Chí Minh
Email: tienntk@ut.edu.vn

https://www.doi.org/10.55228/JTST14040801
Ngày nhận bài: 16/04/2025; Ngày nhận bài sửa: 16/05/2025; Ngày chấp nhận đăng: 7/7/2025
Ngày xuất bản trực tuyến: 15/7/2025
pISSN: 1859-4263; eISSN: 3030-4261

1. Giới thiệu

Dự đoán giá cổ phiếu là một bài toán phân tích chuỗi thời gian phức tạp [1] trong Khoa học Dữ liệu, thu hút sự quan tâm lớn trong lĩnh vực tài chính và đầu tư. Thị trường chứng khoán, với vai trò là chỉ báo sức khỏe kinh tế, tác động trực tiếp đến các quyết định đầu tư và chiến lược tài chính. Việc xây dựng mô hình dự đoán chính xác giá cổ phiếu mang lại lợi ích to lớn, giúp các nhà đầu tư tối đa hóa lợi nhuận, giảm thiểu rủi ro và nâng cao hiệu quả quản lý danh mục. Tuy nhiên, nhiệm vụ này đặt ra nhiều thách thức do đặc thù của dữ liệu tài chính:

- Tính trễ (Lagged Dependence) Giá cổ phiếu tại một thời điểm chịu ảnh hưởng mạnh mẽ từ các yếu tố quá khứ, đòi hỏi các mô hình Khoa học Dữ liệu phải có khả năng nắm bắt và phân tích mối quan hệ phụ thuộc theo thời gian;
- Tính không ổn định (Non-stationarity): Thị trường tài chính biến động khó lường do tác động của các yếu tố kinh tế vĩ mô, chính trị và tâm lý nhà đầu tư. Điều này đòi hỏi các mô hình phải có khả năng thích ứng với sự thay đổi của phân phối dữ liệu theo thời gian;
- Tính chu kỳ (Seasonality): Dữ liệu chứng khoán thường thể hiện các xu hướng lặp lại theo chu kỳ, như biến động theo ngày giao dịch, tháng hoặc mùa. Các mô hình cần có khả năng phát hiện và mô hình hóa các mẫu chu kỳ này.

Trong bối cảnh Khoa học Dữ liệu phát triển mạnh mẽ, đặc biệt là lĩnh vực Học Sâu (Deep Learning, DL), các phương pháp phân tích chuỗi thời gian tiên tiến đã được áp dụng để giải quyết bài toán dự đoán giá cổ phiếu [2-3]. Học Sâu, với khả năng học các mối quan hệ phi tuyến và trích xuất đặc trưng ẩn từ dữ liệu, cho phép xây dựng các mô hình có độ chính xác cao, vượt trội so với các phương pháp truyền thống [4]. Các mô hình Học Sâu như Mạng Nơ-ron Hồi quy Dài-Ngắn hạn (LSTM) [3] và Mạng Nơ-ron Tích chập Thời gian (TCN) [5] đã chứng minh được hiệu quả trong việc nắm bắt các đặc điểm phức tạp của dữ liệu tài chính, từ đó cải thiện đáng kể khả năng dự đoán giá cổ phiếu. Mở rộng hướng tiếp cận này, mô hình Mamba với kiến trúc mạng nơ-ron mới dựa trên mô hình Không gian Trạng thái Chọn lọc (Selective State Space model), được giới thiệu lần đầu vào năm 2023 trong nghiên cứu [6], đang cho thấy nhiều triển vọng trong các bài toán xử lý dữ liệu chuỗi, bao gồm cả dự báo giá cổ phiếu.

Nghiên cứu này hướng đến việc phát triển mô hình học sâu dựa trên kiến trúc Mamba cho bài toán dự báo giá cổ phiếu với các mục tiêu tăng hiệu suất dự đoán giá cổ phiếu, và đánh giá khả năng ứng dụng của mô hình này trong các lĩnh vực tài chính.

Để đạt được các mục tiêu trên, nghiên cứu được triển khai qua các bước sau:

- Thu thập, xử lý và phân tích dữ liệu cổ phiếu. Dữ liệu được thu thập bao gồm các đặc trưng như giá mở cửa (Open), giá đóng cửa (Close), giá cao nhất (Max), giá thấp nhất (Min), và khối lượng giao dịch (Volume) được thu thập từ các nguồn uy tín như Yahoo Finance. Xử lý và phân tích dữ liệu bao gồm các bước làm sạch dữ liệu, xử lý các giá trị bị thiếu, và chuẩn hóa nhằm đảm bảo chất lượng đầu vào cho mô hình;
- Khảo sát các nghiên cứu liên quan cho bài toán dự đoán giá cổ phiếu của các công ty nhằm lựa chọn phương án về mô hình phù hợp;
- Phát triển kiến trúc mô hình dựa trên kiến trúc Mamba [6] nhằm cải thiện hiệu quả tính toán thông qua các cơ chế như Kernel Fusion, Parallel Scan, và Recomputation, đồng thời tăng cường khả năng dự báo chuỗi thời gian dài;
- Huấn luyện và đánh giá mô hình trên tập dữ liệu lịch sử cổ phiếu, áp dụng các kỹ thuật tối ưu hóa như điều chỉnh siêu tham số và tinh chỉnh từng phần. Hiệu suất được đánh giá trên tập kiểm tra thông qua các chỉ số như MSE (Mean Squared Error), MAE (Mean Absolute Error), và MAPE (Mean Absolute Percentage Error);
- Tiến hành thực nghiệm trong các tình huống thực tế như: dự báo giá cổ phiếu hàng giờ hoặc hàng ngày. Đánh giá hiệu suất với mục tiêu cân bằng giữa độ chính xác và chi phí tính toán.

2. Cơ sở lý thuyết và phương pháp

Mamba là một kiến trúc mô hình hóa chuỗi mới, được thiết kế để xử lý dữ liệu chuỗi dài một cách hiệu quả hơn. Mô hình Không gian Trạng thái Chọn lọc (Selective State Space Model - S6) trong kiến trúc

Mamba được thay thế cho cơ chế Attention trong Transformer, giúp Mamba có khả năng xử lý các chuỗi dữ liệu rất dài mà không gặp phải vấn đề về chi phí tính toán và bộ nhớ như Transformer [7]. Các mô hình dựa trên Transformer thường gặp khó khăn với các chuỗi dài với độ phức tạp tính toán là $O(n^2)$, trong khi độ phức tạp tính toán của Mamba là tuyến tính O(n) tương ứng với tỉ lệ tăng của dữ liệu. LSTM có độ phức tạp tính toán tương tự như Mamba, tuy nhiên Mamba có lợi thế lớn nhờ khả năng tính toán song song, điều mà LSTM không có.

2.1. Mô hình dãy trạng thái và Mô hình dãy trạng thái có chọn lọc

Mô hình dãy trạng thái (State Space Model - SSM) được đề xuất trong bài báo [8] là một dạng mô hình không gian trạng thái có cấu trúc kết hợp các ý tưởng từ mạng nơ-ron hồi quy (RNNs) và mạng nơ-ron tích chập (CNNs). SSM được thiết kế để xử lý hiệu quả các chuỗi dữ liệu dài với khả năng mô hình hóa phụ thuộc xa. Một SSM thường được định nghĩa bởi một hệ thống các phương trình vi phân (trong trường hợp thời gian liên tục) hoặc phương trình sai phân (trong trường hợp thời gian rời rạc) mô tả sự tiến triển của một trạng thái ẩn theo thời gian dưới tác động của một đầu vào. Phương trình trạng thái liên tục của SSM được định nghĩa như sau:

$$\boldsymbol{h}'(\boldsymbol{t}) = \boldsymbol{A}\boldsymbol{h}(\boldsymbol{t}) + \boldsymbol{B}\boldsymbol{x}(\boldsymbol{t}) \quad (1)$$

$$y(t) = Ch(t) \quad (2)$$

trong đó: $h(t)$ là trạng thái ẩn, $x(t)$ là đầu vào, và $y(t)$ là đầu ra. A, B, C là các ma trận điều chỉnh động lực của hệ thống.

Sau khi rời rạc hóa, phương trình (1), (2) được biểu diễn dưới dạng tuyến tính thời gian bất biến (LTI) với:

$$h_t = \underline{A}h_{t-1} + \underline{B}x_t \quad (3)$$

$$y_t = Ch_t \quad (4)$$

Ma trận $\underline{A}$ và $\underline{B}$ (3) được rời rạc hóa bằng:

$$\underline{A} = exp(\Delta A) \quad (5)$$

$$B = (\Delta A)^{-1}(exp(\Delta A) - I).\Delta B \quad (6)$$

trong đó: Δ là bước thời gian; I – ma trận đơn vị.

Mô hình Không gian Trạng thái Chọn lọc (S6) [8] là phiên bản mở rộng của SSM, được thêm cơ chế chọn lọc phụ thuộc dữ liệu đầu vào. Có nghĩa là thay vì cố định các tham số *A, B, C,* các tham số này được điều chỉnh dựa trên dữ liệu đầu vào. Điều này giúp mô hình xử lý dữ liệu không liên tục và chọn lọc thông tin hiệu quả hơn.

Cơ chế chọn lọc được mô tả bằng các phương trình sau:

$$A_t = f_A(x_t) \quad (7)$$

$$B_t = f_B(x_t) \quad (8)$$

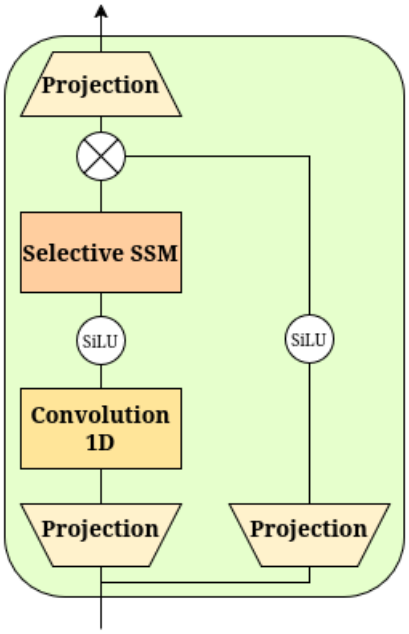
$$C_t = f_C(x_t) \quad (9)$$

trong đó các hàm $\boldsymbol{f_A}, \boldsymbol{f_B}, \boldsymbol{f_C}$ được huấn luyện để chọn lọc và xử lý thông tin theo nội dung chuỗi.

2.2. Kiến trúc Mamba

Mamba là một kiến trúc dựa trên mô hình Selective State Space Model (S6) [6],[8] với thiết kế tối ưu hóa cho việc xử lý dữ liệu chuỗi dài. Kiến trúc tổng thể của mô hình Mamba (hình1) được xây dựng bằng cách kết hợp các thành phần chính sau:

- Mô hình Không gian trạng thái Chọn lọc (Selective SSM): Đây là thành phần cốt lõi của Mamba, đảm nhiệm việc xử lý trạng thái của chuỗi dữ liệu. Cơ chế chọn lọc cho phép mô hình tập trung vào những thông tin quan trọng từ đầu vào tại mỗi bước thời gian;
- Tích chập 1D (Convolution 1D): Lớp tích chập một chiều được sử dụng ở giai đoạn đầu để trích xuất các đặc trưng cục bộ và nắm bắt các mẫu ngắn hạn trong chuỗi đầu vào;
- Hàm kích hoạt SiLU (SiLU Activation): Hàm kích hoạt SiLU (Sigmoid Linear Unit) được áp dụng để đảm bảo tính phi tuyến vào mô hình, cho phép nó học được các mối quan hệ phức tạp hơn trong dữ liệu;
- Lớp chiếu tuyến tính (Linear Projection): Các lớp chiếu tuyến tính được đặt ở đầu và cuối của mỗi khối Mamba. Lớp đầu tiên chuyển đổi đầu vào sang không gian trạng thái phù hợp, trong khi lớp cuối cùng chiếu trạng thái trở lại không gian đầu ra mong muốn;
- Kết nối dư (Residual Connection): Các kết nối dư được sử dụng để cộng trực tiếp thông tin từ đầu vào ban đầu với kết quả xử lý của các tầng Mamba. Điều này giúp duy trì luồng thông tin, ngăn chặn hiện tượng vanishing gradient và cải thiện quá trình huấn luyện mô hình.



Hình 1. Sơ đồ cấu trúc của khối Mamba.

2.3. Cơ chế tối ưu hóa của Mamba

Mamba tối ưu hóa hiệu suất và tính toán bằng ba cơ chế chính:

- Quét Song Song (Parallel Scan): Tăng tốc độ xử lý chuỗi dài bằng cách tận dụng tính toán song song trên GPU, giảm thời gian xử lý bằng cách giảm sự phụ thuộc lẫn nhau giữa các bước tính toán, tối ưu hóa số lượng phép toán dấu chấm động (FLOPs);

- Hợp Nhất Kernel (Kernel Fusion): Giảm số lần truy cập bộ nhớ (từ bộ nhớ băng thông cao HBM sang bộ nhớ truy cập nhanh SRAM và ngược lại) bằng cách kết hợp nhiều phép tính vào một kernel GPU duy nhất. Việc nàyse giúp tối ưu hóa băng thông và giảm độ trễ. Các tham số và các phép tính trung gian được xử lý trong SRAM rồi mới ghi kết quả cuối cùng trở lại HBM;

- Tính Toán Lại (Recomputation): Trong quá trình lan truyền ngược (backpropagation), thay vì lưu trữ tất cả các trạng thái trung gian (tốn nhiều bộ nhớ), Mamba sẽ tính toán lại các trạng thái này khi cần. Việc này giúp giảm đáng kể yêu cầu bộ nhớ, đặc biệt quan trọng khi xử lý chuỗi rất dài với hiệu suất tương đương với các kỹ thuật như Flash Attention [9].

2.4. Kiến trúc tổng quan của mô hình được đề xuất và quy trình hoạt động tổng thể

Kiến trúc của mô hình được đề xuất (hình 2) sử dụng các khối Mamba trong cả hai phần khung xương (Backbone) và phần tiêu đề (Headers) với các kiến trúc riêng biệt cho các lớp đầu ra tương ứng với các dự đoán giá cổ phiếu (Open, Close, High, Low), và Volume .

a) Backbone:

- Chức năng: Trích xuất đặc trưng từ dữ liệu cổ phiếu lịch sử;
- Cấu trúc: Gồm N khối Mamba xếp chồng lên nhau, xử lý tuần tự để nắm bắt các mối quan hệ dài hạn;
- Đầu vào: Chuỗi dữ liệu lịch sử (giá mở cửa, đóng cửa, cao nhất, thấp nhất, khối lượng giao dịch);
- Đầu ra: Vector embedding biểu diễn các đặc trưng đã trích xuất.

b) Headers:

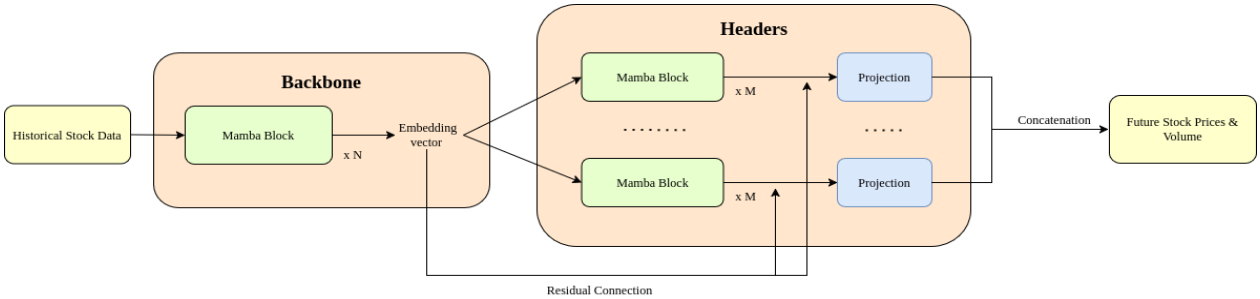
- Chức năng: Dự đoán các giá trị cụ thể trong tương lai (giá mở cửa, đóng cửa, cao nhất, thấp nhất, khối lượng giao dịch);
- Cấu trúc: Gồm M khối Mamba, mỗi khối xử lý một chiều dữ liệu. Kết quả từ các khối này được tổng hợp;
- Đầu vào: Vector embedding từ Backbone (được sao chép cho mỗi khối Mamba);
- Các bước xử lý:

- Projection: Mỗi khối Mamba chiếu vector embedding thành giá trị dự đoán cho một thông số cụ thể.

- Concatenation: Ghép nối các giá trị dự đoán từ tất cả các khối Mamba để tạo thành kết quả cuối cùng.

c) Quy trình hoạt động tổng thể:

- Trích xuất đặc trưng: Dữ liệu lịch sử được đưa vào Backbone để tạo ra vector embedding;
- Dự đoán: Vector embedding được đưa vào Headers. Các khối Mamba trong Headers thực hiện projection để dự đoán các giá trị cụ thể;
- Kết hợp: Các dự đoán từ các khối Mamba trong Headers được ghép nối để đưa ra dự đoán cuối cùng về giá và khối lượng giao dịch.



Hình 2. Kiến trúc tổng thể của mô hình được đề xuất.

3. Xử lý dữ liệu và huấn luyện mô hình

3.1. Tiền xử lý dữ liệu

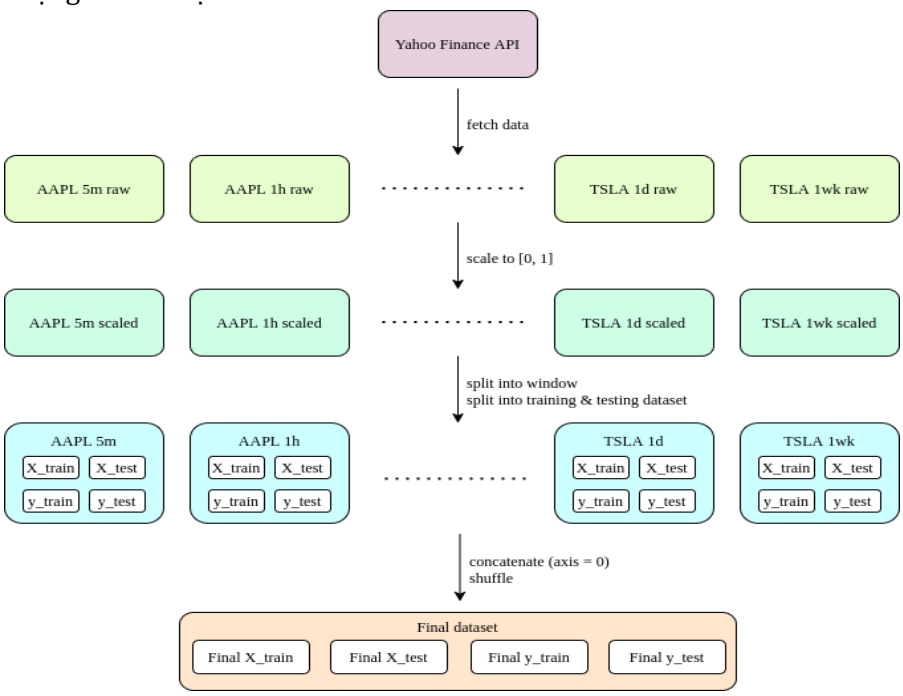
Tiền xử lý dữ liệu là bước quan trọng trong quy trình xây dựng mô hình dự đoán. Mục tiêu của phương pháp này là chuẩn hóa dữ liệu và tạo ra tập dữ liệu huấn luyện và kiểm thử chất lượng cao, được mô tả chi tiết
ở hình 3. Quy trình tiền xử lý dữ liệu bao gồm các bước sau:

i. Thu thập dữ liệu: Dữ liệu lịch sử cổ phiếu được thu thập từ Yahoo Finance API [10]. Dữ liệu thô gồm giá mở, đóng, cao, thấp và khối lượng giao dịch của 10 mã cổ phiếu (AAPL, AMZN, BRK-B, GOOGL, JNJ, META, MSFT, NVDA, TSLA, XOM) theo nhiều khung thời gian (Time frames) như 5 phút, 30 phút, 1 giờ, 4giờ, 5 giờ, 1 ngày, 1 tuần. nhằm đảm bảo tính đa dạng của dữ liệu.

ii. Chuẩn hóa dữ liệu: Dữ liệu được chuẩn hoá giá trị dữ liệu trong khoảng [0, 1] theo phương pháp Min-Max Scaling để đảm bảo tính đồng nhất đặc trưng.

iii. Tiến hành tạo các mẫu huấn luyện từ các cửa sổ trượt (sliding windows): Dữ liệu đã chuẩn hóa được chia thành các cửa sổ thời gian (sliding windows) để tạo ra các mẫu huấn luyện. Đối với mô hình đề xuất này chúng tôi xử lý mỗi cửa sổ bao gồm 32 time points . Dữ liệu của mỗi khung thời gian và mỗi mã cổ phiếu được tách thành tập huấn luyện (X_{train}, y_{train}) và tập kiểm thử (X_{test}, y_{test}).

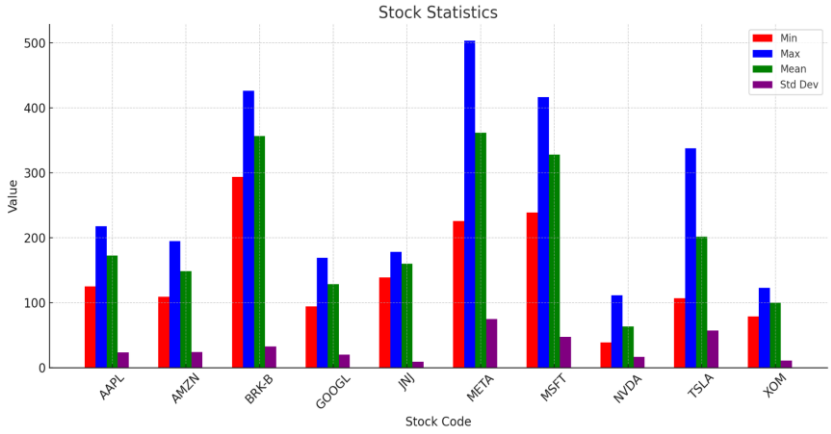
iv. Tổng hợp dữ liệu: Các tập dữ liệu từ các khung thời gian và mã cổ phiếu khác nhau được tổng hợp lại bằng cách nối dọc (concatenate along axis=0). Sau đó, dữ liệu được xáo trộn ngẫu nhiên (shuffle) để giảm thiểu thiên lệch trong quá trình huấn luyện.



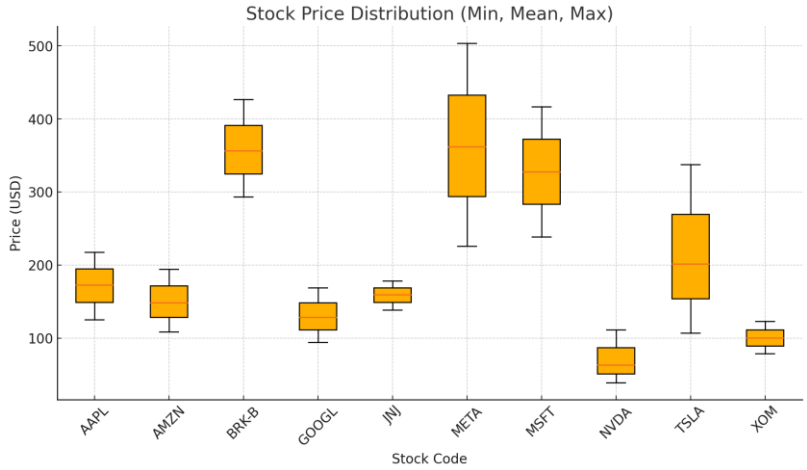
Hình 3. Quy trình tiền xử lý dữ liệu.

Bảng1. Thống kê lịch sử thu thập time points thông qua Yahoo Finance API theo các sessions.

Time frames	Start time	End time	Timepoints
5m	2024-08-20 09:30:00	2024-10-25 12:50:00	37070
30m	2024-08-20 09:30:00	2024-11-12 09:30:00	7680
1h	2022-10-20 09:30:00	2024-07-03 09:30:00	29710
4h	2022-10-20 08:00:00	2024-06-20 12:00:00	10030
1d	2019-01-02 00:00:00	2023-09-27 00:00:00	11930
1wk	2019-01-01 00:00:00	2023-10-03 00:00:00	2490



Hình 4. Biểu đồ biểu thị thống kê các đặc trưng của dữ liệu thu thập theo mã cổ phiếu.



Hình 5. Biểu đồ Plot Box thể hiện đặc trưng về giá của dữ liệu đã thu thập theo mã cổ phiếu.

3.2. Kỹ thuật tối ưu hoá quá trình huấn luyện mô hình

Nhằm tối ưu hoá hiệu quả và giảm thời gian đào tạo mô hình, quá trình huấn luyện được chia làm 3 giai đoạn:

- Giai đoạn 1: Huấn luyện toàn bộ mô hình;
- Giai đoạn 2: Tinh chỉnh từng Header độc lập (hình 8);
- Giai đoạn 3: Đánh giá và điều chỉnh lại toàn bộ mô hình.

Trong giai đoạn 1, mô hình được huấn luyện ban đầu trên toàn bộ tập dữ liệu. Giai đoạn này nhằm đảm bảo các thành phần Backbone và Headers học được các đặc trưng tổng quát từ dữ liệu lịch sử. Trong quá trình này, phương pháp giảm tốc độ học (learning rate decay) dựa trên epoch được áp dụng để tăng tính ổn định và hiệu quả hội tụ. Cụ thể:

- Backbone: Xử lý và trích xuất các đặc trưng quan trọng từ chuỗi dữ liệu đầu vào;
- Headers: Sử dụng các đặc trưng từ Backbone để dự đoán các chỉ số tài chính, bao gồm giá cổ phiếu và khối lượng giao dịch;
- Learning Rate Decay: Tốc độ học (learning rate) được giảm dần theo công thức:

$$\eta_t = \eta_0 \cdot \frac{1}{1 + k \cdot t}$$

trong đó, η_t - tốc độ học tại epoch t; η_0 - tốc độ học ban đầu; k - hệ số giảm tốc độ học.

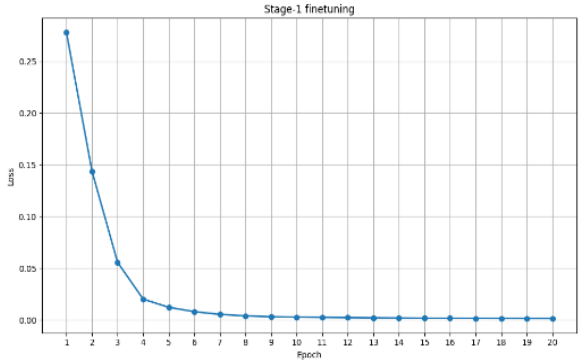
Biểu đồ mất mát sau khi huấn luyện mô hình 20 epoch được biểu diễn trong hình 6.

3.3. Quy trình huấn luyện

Giai đoạn 1 sẽ được tiến hành huấn luyện toàn bộ mô hình với tập dữ liệu đã được sử lý. Các tham số mô hình được thiết lập trong quá trình huấn luyện như sau:

- Kích thước vector ẩn $d_{model} = 32$;
- Số lớp chính trong backbone $n\_layers\_backbone = 3$;
- Số lớp trong phần đầu ra (header) $n\_layer\_header = 6$;
- Số đặc trưng đầu vào tại mỗi bước thời gian $num\_feature = 7$;
- Kích thước trạng thái ẩn $d\_state = 16$;
- Hệ số mở rộng tầng ẩn $expand\_factor = 2$;
- Kích thước kernel convolution – xử lý các tương tác cục bộ ngắn hạn giữa các thời điểm liền kề $d\_conv = 4$;
- Số lượng mẫu xử lý đồng thời trong mỗi bước huấn luyện – ảnh hưởng đến tốc độ và độ ổn định $batch\_size = 2048$;
- Số lần quét toàn bộ tập dữ liệu huấn luyện $epochs = 20$;
- Tốc độ cập nhật trọng số $learning\_rate = 1e - 4$.

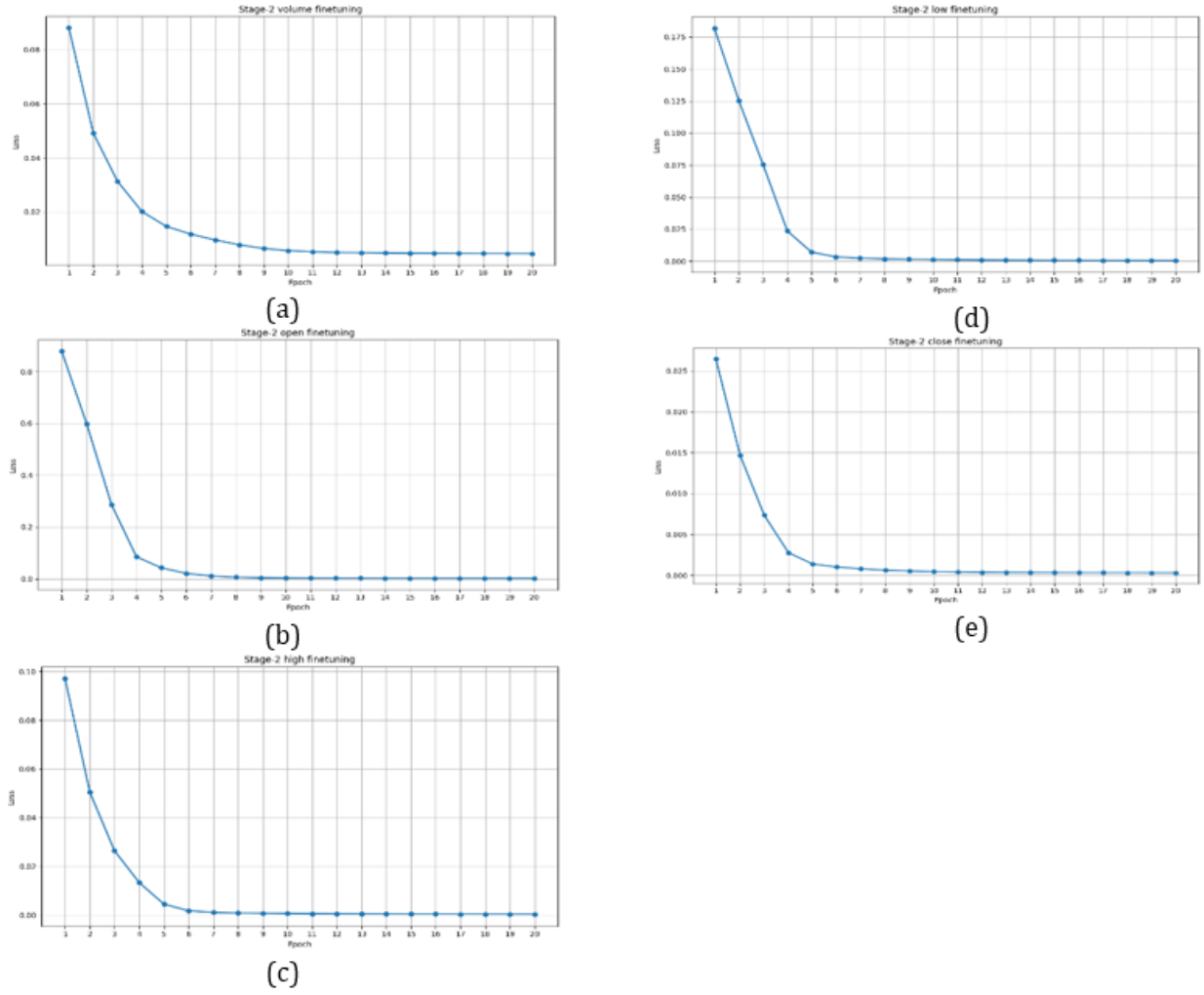
Biểu đồ mất mát của quá trình huấn luyện mô hình giai đoạn 1 được biểu diễn trong hình 6.



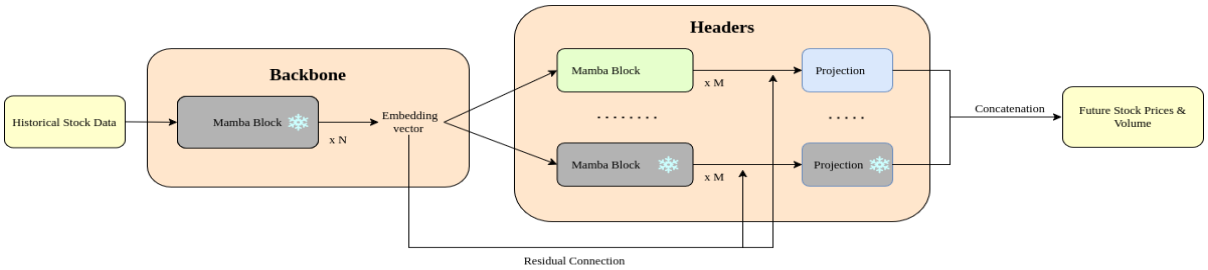
Hình 6. Biểu đồ mất mát của quá trình huấn luyện mô hình giai đoạn 1.

Sau khi hoàn thành giai đoạn 1, từng Header sẽ được tinh chỉnh riêng lẻ nhằm tối ưu hóa khả năng dự đoán của chúng. Quá trình này sử dụng embedding vector từ Backbone đã được huấn luyện trước đó, tập trung vào việc cải thiện độ chính xác của từng chỉ số tài chính. Để duy trì các đặc trưng đã học, Backbone được cố định (Freeze) trong suốt giai đoạn tinh chỉnh, đảm bảo rằng chỉ các Header mới được cập nhật trọng số. Mỗi Header được huấn luyện độc lập để dự đoán các nhãn cụ thể như giá mở cửa (Open), giá đóng cửa (Close), giá cao nhất (Max), giá thấp nhất (Min) và khối lượng giao dịch (Volume). Đặc biệt, quá trình tối ưu hóa phép chiếu (Projection Optimization) trong từng Header giúp cải thiện khả năng học các mối quan hệ phức tạp giữa embedding vector và nhãn dự đoán, từ đó nâng cao hiệu suất của mô hình.

Với kỹ thuật tối ưu hoá này, mô hình không chỉ cải thiện hiệu suất dự đoán cho từng chỉ số tài chính mà còn giảm thiểu lỗi dự đoán tổng thể. Kỹ thuật giảm tốc độ học và tinh chỉnh Header cho phép tăng cường tính linh hoạt của mô hình, đặc biệt trong các bài toán dự đoán dữ liệu chuỗi thời gian với nhiều biến số phức tạp.



Hình 7. Biểu đồ mất mát của quá trình huấn luyện mô hình giai đoạn 2 a- Volume, b-Open, c-High, d-Low, e-Close.



Hình 8. Tinh chỉnh từng Header sau khi huấn luyện toàn bộ mô hình.

4. Đánh giá, so sánh hiệu suất mô hình

4.1. Các chỉ số đánh giá hiệu suất của mô hình

Hiệu suất dự đoán của các mô hình được đánh giá thông qua các chỉ số:

- MSE (Mean Square Error): Đánh giá sai số trung bình bình phương;

$$MSE = \frac{1}{n}\sum_{t=1}^{n}\left|\hat{X}_t - X_t\right|^2$$

- MAE (Mean Absolute Error): Đo lường sai số trung bình tuyệt đối, ít nhạy cảm với các ngoại lệ;

$$MAE = \frac{1}{n}\sum_{t=1}^{n}\left|\hat{X}_t - X_t\right|$$

- MAPE (Mean Absolute Percentage Error): Đo lường sai số phần trăm, phù hợp để so sánh trên các tập dữ liệu khác nhau.

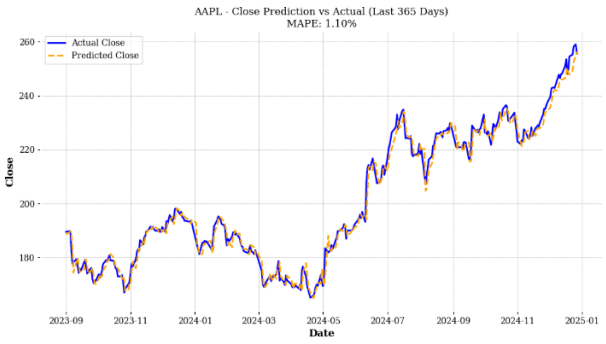
$$MAPE = \frac{1}{n}\sum_{t=1}^{n}\left|\frac{\hat{X}_t - X_t}{X_t}\right| \times 100$$

4.2. Kết quả thực nghiệm của mô hình đã được huấn luyện

Sau khi đã huấn luyện xong mô hình Mamba, quá trình thực nghiệm đã được tiến hành để dự đoán giá mở cửa (Open), giá đóng cửa (Close), giá cao nhất (High), giá thấp nhất (Low), lượng giao dịch (Volume) của 10 mã chứng khoán theo các phiên khác nhau (5 phút, 30 phút, 1 giờ, 4 giờ, 1 ngày, 1 tuần) gồm 6949 mẫu , kết quả được hiển thị trong bảng 2, và biểu đồ 9.

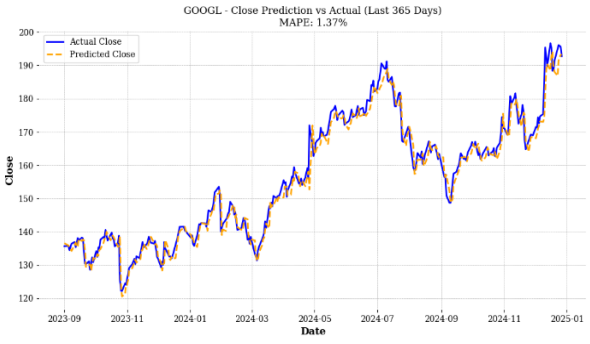
Bảng 2. Kết quả. đánh giá thực nghiệm của mô hình Mamba dựa trên tập dữ liệu của 10 mã chứng khoán theo các phiên giao dịch khác nhau.

Phiên giao dịch	Mamba (396KB)		
	MSE	MAE	MAPE
30min (529 mẫu)	1.74291	0.77535	0.31%
1 hour (3496 mẫu)	4.12556	1.08378	0.47%
4 hour (1170 mẫu)	12.8639	2.11662	0.92%
1 day (1452 mẫu)	19.70867	2.737	1.37%
1 wk (302 mẫu)	138.13575	8.93049	3.58%

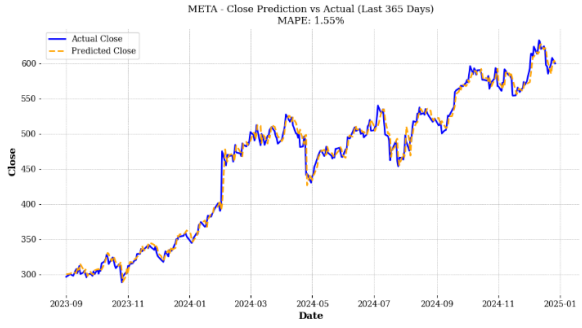


(a) Dự đoán trên AAPL phiên 1 ngày

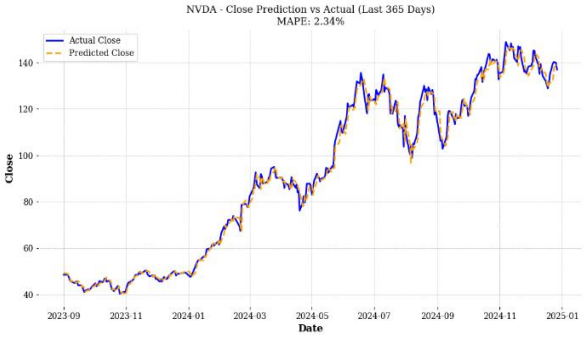
(b) Dự đoán trên AMZN phiên 1 ngày

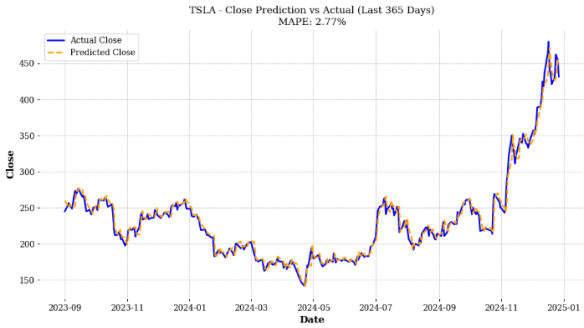


(c) Dự đoán trên GOOGL phiên 1 ngày



(d) Dự đoán trên META phiên 1 ngày





(e) Dự đoán trên NVDA phiên 1 ngày (f) Dự đoán trên TSLA phiên 1 ngày

Hình 9. Dự đoán giá đóng cửa của cổ phiếu theo phiên giao dịch 1 ngày của 6 mã cổ phiếu.

4.3. So sánh hiệu suất của mô hình đã được huấn luyện Mamba với các mô hình học máy và học sâu khác

Để so sánh hiệu suất của mô hình Mamba với các mô hình học máy và học sâu khác, chúng tôi đã tiến hành huấn luyện và dự đoán tương tự phần 3.2 cho các mô hình ElasticNet, CNN1D, LSTM. Kết quả đánh giá, so sánh dựa theo chỉ số MAPE theo các phiên giao dịch khác nhau và thời gian xử lý cho từng dự đoán được trình bày trong bảng 3 và 4.

Từ kết quả thu được có thể thấy rằng mô hình dựa trên kiến trúc Mamba có thể dự đoán ổn định với tất cả các phiên giao dịch có biên độ thời gian khác nhau với giá trị MAPE trong khoảng [0.111, 1.366], vượt qua cả lợi thế về độ chính xác của mô hình máy học cho phiên ngắn như 5 phút, 30 phút, đặc biệt trong phiên dài 1 tuần ba mô hình (ElasticNet, CNN1D, LSTM) đều không dự đoán hiệu quả với chỉ số MAPE trên 30%. Về thời gian xử lý, so với các mô hình học sâu khác, mô hình Mamba xử lý nhanh hơn mô hình LSTM 7,1 lần, CNN1D – 9,2 lần. Phương pháp máy học ElasticNet vẫn chiếm ưu thế về thời gian xử lý (nhanh hơn 4 lần), tuy nhiên phương pháp này không hiệu quả với những phiên giao dịch biên độ lớn từ 4h đến 1 tuần.

Bảng 3. Kết quả đánh giá các mô hình Elastic, CNN1D, LSTM, Mamba theo chỉ số MAPE.

Phiên giao dịch	MAPE (%)			
	ElasticNet	CNN1D (848.05MB)	LSTM (176.21MB)	Mamba (396KB)
5min (3169 samples)	0.145	1.735	0.411	0.111
30min (529 samples)	0.88	3.599	4.17	0.314
1 hour (3496 samples)	3.03	9.722	2.8	0.468
4 hour (1170 samples)	9.779	13.942	7.105	0.92
1 day (1452 samples)	25.563	15.375	8.515	1.366
1 wk (302 samples)	96.57	32.439	40.209	3.577

Bảng 4. Thời gian xử lý cho mỗi dự đoán giao dịch của các mô hình khác nhau.

Processing time (ms)			
ElasticNet	CNN1D (848.05MB)	LSTM (176.21MB)	Mamba (396KB)
3.2	29.7	22.8	12.6

5. Kết luận

Nghiên cứu này tập trung vào việc phát triển mô hình học sâu cho bài toán dự đoán giá cổ phiếu. Nghiên cứu này đã tiến hành thực hiện nâng cao hiệu quả dự đoán giá cổ phiếu của mô hình bằng các giải pháp như: xây dựng mô hình sử dụng các khối Mamba và kỹ thuật tối ưu hoá quá trình huấn luyện mô hình. Kết quả thực nghiệm cho thấy mô hình đề xuất này đạt được độ chính xác cao và khả năng tổng quát hóa tốt trên nhiều mã cổ phiếu và khung thời gian giao dịch khác nhau.

Các điểm nổi bật của nghiên cứu bao gồm:

- Kiến trúc mô hình: Thiết kế và triển khai kiến trúc mô hình kết hợp giữa Backbone và Headers, giúp nắm bắt hiệu quả các đặc trưng của dữ liệu chuỗi thời gian và tối ưu hóa khả năng dự đoán của mô hình;
- Khả năng dự đoán đa đầu ra: Mô hình có khả năng dự đoán đồng thời nhiều giá trị, bao gồm giá mở cửa, giá cao nhất, giá thấp nhất, giá đóng cửa và khối lượng giao dịch;
- Tinh chỉnh Header độc lập: Thực hiện tinh chỉnh độc lập từng Header để tối ưu hóa hiệu suất dự đoán cho từng chỉ số tài chính cụ thể;
- Kỹ thuật huấn luyện: Ứng dụng kỹ thuật giảm tốc độ học dựa trên epoch, giúp cải thiện độ ổn định và tốc độ hội tụ trong quá trình huấn luyện;

Mặc dù nghiên cứu đã đạt được nhiều kết quả đáng khích lệ, tuy nhiên vẫn cần phải tiếp tục nghiên cứu phương pháp hậu xử lý (post-processing) khi gặp phải các các vấn đề dưới đây:

- Tác động của các yếu tố ngẫu nhiên: Thị trường tài chính chịu ảnh hưởng lớn từ các yếu tố ngẫu nhiên, chẳng hạn như các sự kiện bất ngờ (thiên tai, khủng hoảng kinh tế) mà mô hình không thể dự đoán được;
- Ảnh hưởng của tin tức tài chính và chính trị: Các thông tin tài chính và chính trị (như thay đổi chính sách thuế, quy định pháp lý mới, hoặc các thông báo từ công ty) có thể tạo ra biến động lớn trong thị trường mà mô hình hiện tại chưa tích hợp đầy đủ.

Trong tương lai, chúng tôi dự định mở rộng nghiên cứu và tiến hành tích hợp ứng dụng theo các hướng sau:

- Tích hợp dữ liệu ngoại sinh: Tăng cường khả năng dự đoán bằng cách kết hợp phân tích các yếu tố tác động từ bên ngoài thị trường như tin tức chính trị, tài chính, dữ liệu kinh tế vĩ mô, và mạng xã hội. Để phân tích khả năng tác động tích cực, hay tiêu cực của dữ liệu ngoại sinh nêu trên, có thể sử dụng các mô hình như FinBERT [11];
- Áp dụng học tăng cường sâu (Deep Reinforcement Learning) trong giai đoạn post-processing để tối ưu hóa các quyết định giao dịch dựa trên dữ liệu dự đoán, giúp mô hình đưa ra các quyết định giao dịch hiệu quả hơn [12];
- Xây dựng hệ thống kiểm thử thời gian thực trên dữ liệu mới nhất để đánh giá và cải thiện hiệu suất mô hình ngay trong điều kiện thực tế.

Đóng góp của các tác giả trong bài báo

Trần Thế Vinh: Phát triển phương pháp; Phát triển thuật toán; Biên soạn dữ liệu; Thực hiện điều tra; Giám sát; Xác thực kết quả; Phân tích chính thức; Trực quan hóa; Viết – bản thảo gốc; Viết – chỉnh sửa và phản hồi. **Nguyễn Thị Khánh Tiên:** Phát triển thuật toán; Biên soạn dữ liệu; Thực hiện điều tra; Giám sát; Xác thực kết quả; Phân tích chính thức; Trực quan hóa; Viết – bản thảo gốc; Viết – chỉnh sửa và phản hồi. **Trần Kim Thanh**: Phát triển thuật toán; Biên soạn dữ liệu; Thực hiện điều tra; Xác thực kết quả; Phân tích chính thức; Giám sát. **Phạm Trần Nhật Linh**: Phát triển thuật toán, Biên soạn dữ liệu; Thực hiện điều tra. **Mai Quỳnh Trâm**: Biên soạn dữ liệu; Thực hiện điều tra

Tuyên bố không xung đột lợi ích và cam kết bản quyền

Nhóm tác giả tuyên bố về sự không xuất hiện những xung đột tiềm ẩn từ nghiên cứu này, và cam kết bài báo chưa từng được công bố trước đây.

Chia sẻ dữ liệu theo yêu cầu

Dữ liệu sẽ được cung cấp theo yêu cầu.

1st Trần Thế Vinh. *Viện công nghệ thông tin và Điện, điện tử, Trường Đại học Giao thông vận tải Thành phố Hồ Chí Minh*
2nd* Nguyễn Thị Khánh Tiên. *Viện công nghệ thông tin và Điện, điện tử, Trường Đại học Giao thông vận tải Thành phố Hồ Chí Minh*

3rd Trần Kim Thanh. *Viện công nghệ thông tin và Điện, điện tử, Trường Đại học Giao thông vận tải Thành phố Hồ Chí Minh*
4th Phạm Trần Nhật Linh. *Viện công nghệ thông tin và Điện, điện tử, Trường Đại học Giao thông vận tải Thành phố Hồ Chí Minh*
5th Mai Quỳnh Trâm. *Viện công nghệ thông tin và Điện, điện tử, Trường Đại học Giao thông vận tải Thành phố Hồ Chí Minh*
*Corresponding author: tienntk@ut.edu.vn

Tài liệu tham khảo

[1] M. Kolambe and S. Arora, "Forecasting the future: A comprehensive review of time series prediction techniques," *Journal of Engineering Science*, vol. 20, no. 2s, 2024, doi: 10.52783/jes.1478.
[2] B. Lim and S. Zohren, "Time series forecasting with deep learning: A survey," *arXiv preprint arXiv:2004.13408*, Apr. 2020, doi: 10.48550/arXiv.2004.13408.
[3] Y. Kong, Z. Wang, Y. Nie, T. Zhou, S. Zohren, X. Liang, P. Sun, and Q. Wen, "Unlocking the power of LSTM for long term time series forecasting," *arXiv preprint arXiv:2408.10006*, Aug. 2024, doi: 10.48550/arXiv.2408.10006.
[4] G. P. Zhang, "Time series forecasting using a hybrid ARIMA and neural network model," *Neurocomputing*, vol. 50, pp. 159–175, Jan. 2003, doi: 10.1016/S0925-2312(01)00702-0.
[5] Nguyen T.K.T., Antoshchuk S., Nikolenko A., Tran K.T., Babilunha O, “Non-stationary time series prediction using one-dimensional Convolutional Neural Network Models.” Herald of Advanced Information Technology, vol. 3, no. 1, pp. 362-372, Apr. 2020, doi:10.15276/hait.01.2020.3
[6] A. Gu and T. Dao, "Mamba: Linear-time sequence modeling with selective state spaces," *arXiv preprint arXiv:2312.00752*, Dec. 2023, doi: 10.48550/arXiv.2312.00752.
[7] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," *arXiv preprint arXiv:1706.03762*, Jun. 2017, doi: 10.48550/arXiv.1706.03762.
[8] A. Gu, K. Goel, and C. Ré, "Efficiently modeling long sequences with structured state spaces," arXiv preprint arXiv:2111.00396, Oct. 2021, doi: 10.48550/arXiv.2111.00396.
[9] T. Dao, D. Y. Fu, S. Ermon, A. Rudra, and C. Ré, "FlashAttention: Fast and memory-efficient exact attention with IO-awareness," arXiv preprint arXiv:2205.14135, May 2022, doi: 10.48550/arXiv.2205.14135.
[10] Yahoo Finance, "Stock market live, quotes, business & finance news," [Online]. Available: https://finance.yahoo.com/. [Accessed: Jun. 30, 2025].
[11] D. Araci, "FinBERT: Financial sentiment analysis with pre-trained language models," arXiv preprint arXiv:1908.10063, Aug. 2019, doi: 10.48550/arXiv.1908.10063.
[12] Z. Zhang, S. Zohren, and S. Roberts, "Deep reinforcement learning for trading," arXiv preprint arXiv:1911.10107, Nov. 2019, doi: 10.48550/arXiv.1911.10107.