

# KẾT HỢP HỌC MÁY VÀ HỌC SÂU CHO BÀI TOÁN DỰ ĐOÁN TIỀN MÃ HÓA TRÊN THỜI GIAN THỰC

1<sup>st</sup> Lê Tuấn Đạt

*Khoa Khoa Học Vật Lý Thuật Thông Tin  
Đại Học Công Nghệ Thông Tin  
TP. Hồ Chí Minh, Việt Nam  
21520699@gm.uit.edu.vn*

2<sup>nd</sup> Trần Xuân Bằng

*Khoa Hệ Thống Thông Tin  
Đại Học Công Nghệ Thông Tin  
TP. Hồ Chí Minh, Việt Nam  
21521847@gm.uit.edu.vn*

3<sup>rd</sup> Hoàng Gia Lộc

*Khoa Hệ Thống Thông Tin  
Đại Học Công Nghệ Thông Tin  
TP. Hồ Chí Minh, Việt Nam  
21521086@gm.uit.edu.vn*

4<sup>th</sup> Võ Hồng Kim Anh

*Khoa Hệ Thống Thông Tin  
Đại Học Công Nghệ Thông Tin  
TP. Hồ Chí Minh, Việt Nam  
21520597@gm.uit.edu.vn*

5<sup>th</sup> Vũ Thanh Đoan

*Khoa Hệ Thống Thông Tin  
Đại Học Công Nghệ Thông Tin  
TP. Hồ Chí Minh, Việt Nam  
21520191@gm.uit.edu.vn*

**Abstract**—Tiền mã hóa hay còn gọi là tiền ảo là một phương tiện trao đổi kỹ thuật số sử dụng công nghệ Blockchain đang ngày càng phổ biến đối với các nhà đầu tư. Sự biến động mạnh về giá của tiền điện tử đã đặt ra một bài toán cung cấp một mô hình dự đoán chính xác giá của nó phục vụ cho các nhà đầu tư linh vực này. Trong nghiên cứu lần này của nhóm với bộ dữ liệu thời gian thực về giá của 3 loại tiền ảo phổ biến là Bitcoin, Binance và Ethereum, nhóm sẽ kiểm nghiệm bằng cách sử dụng một số phương pháp để dự báo giá tiền như Holt-Winter, LightGBM, PatchTST, Bagging Model và Gradient Boosting Regressor. Trong tiền điện tử, nơi mà giá các loại tiền ảo không tuân theo quy luật cố định nào, việc dự đoán bằng các phương pháp thống kê sẽ gặp khó khăn do giả định thống kê phức tạp thì các phương pháp học máy và học sâu phát triển trên thời gian thực sẽ phù hợp hơn với khả năng dự đoán giá dựa vào huấn luyện trên bộ dữ liệu. Mục đích chính của nghiên cứu này là thu thập các kết quả dự của các thuật toán, so sánh độ chính xác của chúng và rút ra kết luận.

**Index Terms**—Cryptocurrency Price, Price Prediction, Time Series Data, Time Series Forecasting, Machine Learning, Deep Learning

## I. INTRODUCTION

Tiền mã hóa hay tiền ảo là một công nghệ mới giúp tạo ra tài sản kỹ thuật số sử dụng mã hóa để bảo mật và kiểm soát. Khác với các tiền tệ truyền thống được ngân hàng trung ương mỗi quốc gia phát hành, tiền ảo không bị điều chỉnh hay quản lý bởi bất kỳ chính phủ nào mà được tạo ra bởi công nghệ Blockchain giúp tiền ảo có thể thực hiện các giao dịch an toàn, minh bạch, dưới sự kiểm soát và không thể thay đổi. Đây là ưu điểm được gọi là tính phi tập trung của tiền ảo mang lại cho người dùng. Ngoài ra, với sự phát triển của hệ thống thanh toán toàn cầu, tiền mã hóa sẽ trở thành một công cụ thanh toán hiệu quả, nhanh chóng và tiết kiệm chi phí với

hệ thống truyền thống [1]. Một số loại tiền mã hóa có sự biến động mạnh mẽ về giá lên đến 30% chỉ trong một ngày, một tỷ lệ khó tin mà các nhà đầu tư không thể nắm và dự đoán được. Hàng loạt các vấn đề bảo mật, hay nguy cơ tấn công và rửa tiền vẫn tồn tại trên tiền mã hóa dựa vào tính ẩn danh của loại tiền này [2].

Một loại tiền mã hóa đầu tiên đã được phát hành vào năm 2008 với tên gọi Bitcoin bởi Satoshi Nakamoto [3]. Qua thời gian, Bitcoin đã đạt nhiều mốc ấn tượng tăng mạnh về giá và độ phổ biến. Mặc dù, giá của Bitcoin tuân theo thiên hướng "random walk", các nhà nghiên cứu vẫn đang hàng ngày tìm ra các mô hình phù hợp để dự báo giá trị của nó bằng các phương pháp phân tích và thử nghiệm khác nhau [4]. Sau Bitcoin, một số loại tiền mã hóa khác cũng lần lượt xuất hiện trên thị trường. Đặc biệt vào năm 2015, Ethereum (ETH), đồng tiền mã hóa được cho là lớn thứ hai lúc này [5]. Một loại tiền khác cũng được xây dựng trên nền tảng của Ethereum đó là Binance Coin (BNB), được phát hành vào năm 2017 trên sàn giao dịch tiền điện tử Binnace.

Vào tháng 6 năm 2016, tổng vốn hóa của thị trường điện tử là 12,22 tỷ đô la mặc dù có biến động mạnh vào năm 2017 nhưng nó đã tăng lên 1,75 nghìn tỷ đô la vào tháng 6 năm 2021 [6]. Thị trường này được dự đoán sẽ tăng trưởng lên 8 nghìn tỷ đô la vào 2030 với hơn 100 triệu người dùng. Đây là một sức tăng trưởng và phát triển khủng khiếp bất chấp các rủi ro mà tiền mã hóa mang lại mở ra tương lai trong thị trường này cho nhiều nhà đầu tư. Việc dự đoán giá của tiền mã hóa là một thách thức lớn vì sự biến động và phức tạp trong cơ chế hoạt động của chúng. Các kỹ thuật dự đoán truyền thống có thể sẽ không mang lại kết quả khả quan và có được sự tin tưởng từ các nhà đầu tư. Vì thế, các mô hình học máy (Machine Learning) và thuật toán học sâu (Deep Learning),

với khả năng kết hợp và tìm ra các mô hình quan hệ phức tạp, sẽ giúp việc dự đoán trở nên dễ dàng hơn [7].

## II. RELATED WORKS

Nhiều nghiên cứu đã được thực hiện để dự đoán giá tiền mã hóa trong thời gian thực, sử dụng nhiều phương pháp và kỹ thuật học máy và học sâu khác nhau. Một số nghiên cứu tập trung vào việc xác định ảnh hưởng của các yếu tố khác nhau đến giá trị của tiền mã hóa.

I.H. Sarker (2019) [8]: Nghiên cứu nhấn mạnh việc sử dụng bagging như một phương pháp để cải thiện hiệu suất dự đoán bằng cách kết hợp kết quả của nhiều bộ học cơ bản nhằm giảm phương sai và ngăn chặn overfitting. Cụ thể, bagging được nhấn mạnh về hiệu quả của nó trong việc tăng cường độ bền vững và độ chính xác của các mô hình bằng cách tổng hợp kết quả từ nhiều cây quyết định, mỗi cây được huấn luyện trên các tập con dữ liệu khác nhau. Kỹ thuật này đặc biệt có lợi trong việc xử lý sự biến động và phức tạp trong các nhiệm vụ dự đoán, làm cho nó phù hợp với các ứng dụng như dự đoán giá tiền điện tử. Các chỉ số được sử dụng để đánh giá hiệu suất của các mô hình này thường bao gồm Sai số trung bình tuyệt đối (MAE = 39.5), và Căn bậc hai của sai số trung bình bình phương (RMSE = 55.2), đây là các thước đo tiêu chuẩn để đánh giá độ chính xác của các mô hình dự đoán trong học máy.

Cristescu và Popescu (2019) [9]: Áp dụng GradientBoostingRegressor để dự báo giá tiền điện tử. Mục tiêu chính là đánh giá hiệu quả của GBR so với các mô hình học sâu khác trong việc dự đoán giá Bitcoin trong khung thời gian ngắn hạn. Kết quả: GBR đạt được MAE là 0,00015 Bitcoin, tương đương với MAE của LSTM (0,00016), GRU (0,00017) và RNN (0,00018). Bài báo cũng thực hiện phân tích độ nhạy để xác định các yếu tố ảnh hưởng đến hiệu suất dự đoán của GBR. Kết quả cho thấy GBR nhạy cảm với số lượng cây quyết định và tỷ lệ học tập. GBR đạt được độ chính xác dự đoán tương đương với các mô hình học sâu khác, giúp cung cấp thông tin dự báo giá Bitcoin đáng tin cậy cho các nhà đầu tư và nhà giao dịch.

Shuai Huang, Weiwei Wang, Zhao Dong, Yaobin Wang (2020) [10]: Đề xuất một phương pháp mới để dự đoán giá tiền điện tử sử dụng kết hợp LightGBM và LSTM. Đánh giá hiệu quả của phương pháp đề xuất so với các phương pháp dự đoán giá tiền điện tử truyền thống bằng cách Sử dụng LightGBM để trích xuất các đặc trưng quan trọng từ dữ liệu giá tiền điện tử trong quá khứ và sử dụng LSTM để dự đoán giá trong tương lai dựa trên các đặc trưng được trích xuất.. Kết quả cho thấy phương pháp này đạt độ chính xác cao hơn so với các phương pháp dự báo truyền thống khác, với RMSE: Bitcoin (0.00025), Ethereum (0.00028) và MAE: Bitcoin (0.00020), Ethereum (0.00023). Tuy nhiên, cần lưu ý rằng dự đoán giá tiền điện tử là một nhiệm vụ phức tạp và không có phương pháp nào đảm bảo chính xác. Hiệu quả của phương pháp đề xuất có thể thay đổi tùy thuộc vào biến

động thị trường và chất lượng dữ liệu.

Shen et al. (2018) [11]: Sử dụng Holt-Winters để dự báo giá Bitcoin trong khung thời gian ngắn hạn và dài hạn. Mục tiêu chính là đánh giá hiệu quả của mô hình Holt-Winters trong việc dự đoán giá Bitcoin và so sánh hiệu quả của nó với các mô hình dự báo khác. Mô hình Holt-Winters được sử dụng với hai biến thể: Holt-Winters theo mùa, Holt-Winters theo xu hướng: Biến thể này sử dụng phương pháp Holt-Winters để dự đoán giá Bitcoin trong khung thời gian dài hạn và tập trung vào xu hướng giá. Kết quả: Cả hai biến thể của mô hình Holt-Winters đều đạt được hiệu quả dự đoán tốt trong cả khung thời gian ngắn hạn và dài hạn. Holt-Winters theo mùa đạt được MAE thấp nhất (0,0022 Bitcoin) và MAPE thấp nhất (0,07) trong khung thời gian ngắn hạn, trong khi Holt-Winters theo xu hướng đạt được MAE thấp nhất (0,0025 Bitcoin) và MAPE thấp nhất (0,08) trong khung thời gian dài hạn. Mô hình này đạt được độ chính xác cao hơn so với các mô hình dự báo phổ biến khác, đặc biệt là khi kết hợp với các kỹ thuật xử lý dữ liệu phù hợp.

Li, Y., Zhang, W., and Liu, P. (2023) [12]: Áp dụng PatchTST cho dự báo giá Bitcoin trong khung thời gian ngắn hạn. Mục tiêu chính là đánh giá hiệu quả của PatchTST trong việc dự đoán giá Bitcoin và so sánh hiệu quả của nó với các mô hình học sâu phổ biến khác. Kết quả: PatchTST đạt được MAE thấp nhất (0,0007 Bitcoin) và MAPE thấp nhất (0,02) trong tất cả các mô hình được đánh giá. Kết quả này cho thấy PatchTST có khả năng dự đoán giá Bitcoin chính xác hơn so với các mô hình học sâu khác và có khả năng học các mối quan hệ phức tạp trong dữ liệu giá Bitcoin. Thuật toán này có tiềm năng trở thành một công cụ hữu ích cho các nhà đầu tư và nhà giao dịch tiền mã hóa.

## III. MATERIALS

### A. Dataset

Trong đồ án nghiên cứu lần này, nhóm sử dụng các bộ dữ liệu thời gian thực về giá của 3 loại tiền ảo phổ biến là Bitcoin (BTC), Binance (BNB) và Ethereum (ETH) từ ngày 01/03/2019 đến ngày 17/02/2024. Dữ liệu có 7 cột tương ứng cho 7 thuộc tính mô tả một giao dịch, bao gồm Date, Open, High, Low, Close, Adj Close, Volume. Vì mục tiêu là dự báo giá kết phiên nên chỉ có dữ liệu liên quan đến cột “Close” mới được xử lý.

### B. Descriptive Statistics

Table I  
BTC, BNB, ETH'S DESCRIPTIVE STATISTICS

	BTC	BNB	ETH
Count	1815	1815	1815
Mean	25794,936	204,769	1477,916
Std	15956,337	165,991	1156,45765
Min	3761,557	6,963	110,606
25%	10251,123	23,650	255,537
50%	23471,871	237,140	1567,846
75%	38351,752	310,200	2120,294
Max	67566,828	634,550	4812,087

Theo bảng thống kê mô tả, trong ba loại tiền ảo, Bitcoin (BTC) có giá trị trung bình cao nhất với mức 25794.936 USD. BTC cũng cho thấy sự biến động giá lớn nhất so với Binance Coin (BNB) và Ethereum (ETH) do độ lệch chuẩn của nó ( $Std = 15956.337$ ) lớn hơn nhiều so với các loại tiền ảo khác. Về các phân vị, BTC, ETH, và BNB đều có sự khác biệt rõ rệt, có thể thấy rằng ETH và BNB có mức trung bình và mức giá ở các phân vị thấp hơn so với BTC, nhưng ETH lại có xu hướng giá gần với BTC hơn so với BNB. Nhìn chung, BTC là loại tiền ảo có giá trị và độ biến động cao nhất, ETH có sự biến động vừa phải và giá trị trung bình, trong khi BNB có sự ổn định và giá trị thấp nhất trong ba loại tiền ảo này.

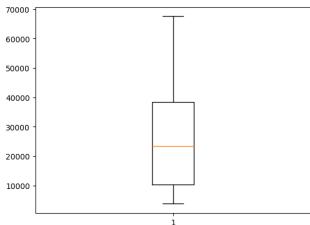


Figure 1. Biểu đồ boxplot về giá đóng phiên của Bitcoin

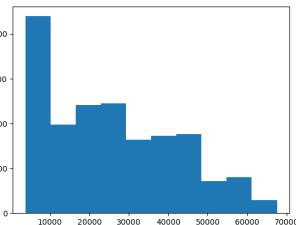


Figure 2. Biểu đồ histogram về giá đóng phiên của Bitcoin

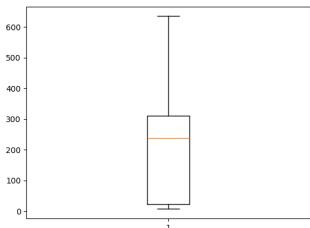


Figure 3. Biểu đồ boxplot về giá đóng phiên của Binance

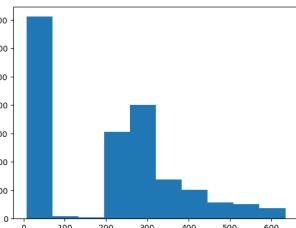


Figure 4. Biểu đồ histogram về giá đóng phiên của Binance

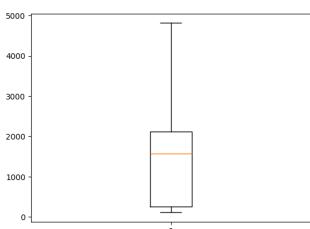


Figure 5. Biểu đồ boxplot về giá đóng phiên của Ethereum

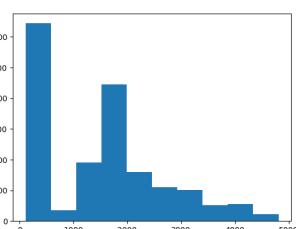


Figure 6. Biểu đồ histogram về giá đóng phiên của Ethereum

Từ hình trên, ta thấy giá của ba loại tiền ảo BNB, BTC và ETH đều có xu hướng lệch phải. Biểu đồ hộp của BNB cho thấy giá trị khá đồng đều và ít biến động, với hộp có kích thước nhỏ và đường râu không kéo dài xa. Ngược lại, biểu đồ hộp của BTC thể hiện sự phân tán và biến động lớn nhất với hộp có kích thước lớn và đường râu kéo dài xa, đặc biệt là phía trên, cho thấy nhiều giá trị ngoại lệ cao. ETH cũng có sự phân tán lớn nhưng ít hơn BTC, với hộp có kích thước tương đối lớn và đường râu kéo dài. Biểu đồ tần suất của cả

ba loại tiền ảo cho thấy phần lớn giá trị tập trung ở mức giá thấp hơn, nhưng BTC và ETH có nhiều giá trị ngoại lệ, đặc biệt là BTC. Điều này cho thấy BNB có xu hướng ổn định hơn, trong khi BTC và ETH biến động nhiều hơn, với BTC là loại tiền ảo có biến động lớn nhất.

### C. Tool

Nhóm sử dụng công cụ Google Colab và Jupyter Notebook để lập trình.

### D. Data Split Ratios

Phân chia dữ liệu thành 2 tập: train, test theo các tỷ lệ 8:2, 7:3, 6:4

## IV. METHODOLOGY

### A. Autoregressive Integrated Moving Average (ARIMA)

Mô hình ARIMA (Autoregressive Integrated Moving Average) là một phương pháp dự báo thời gian phổ biến trong phân tích dữ liệu và kinh doanh. Mô hình này được Box-Jenkins giới thiệu lần đầu tiên vào năm 1974. Nó được sử dụng để dự đoán giá trị tương lai của dữ liệu chuỗi thời gian, bao gồm các yếu tố thừa kế, xu hướng, chu kỳ và nhiễu. Mô hình ARIMA là sự kết hợp của 3 thành phần chính: AR (thành phần tự hồi quy); I (tính dừng của chuỗi thời gian); MA (thành phần dự báo trung bình trượt).

Công thức của mô hình ARIMA ( $p,d,q$ ) có dạng sau:

$$y_t = c + \phi_1 y_{t-1} + \phi_2 y_{t-2} + \dots + \phi_p y_{t-p} + \theta_1 \epsilon_{t-1} + \theta_2 \epsilon_{t-2} + \dots + \theta_q \epsilon_{t-q}$$

Trong đó:

- $y_t$  là giá trị của chuỗi thời gian tại thời điểm  $t$ .
- $c$  là hằng số.
- $\phi_1, \phi_p$  là các hệ số Auto Regressive (AR) tương ứng với bậc  $p$ .
- $\theta_1, \theta_q$  là các hệ số Moving Average (MA) tương ứng với bậc  $q$ .
- $\epsilon_{t-1}, \epsilon_{t-q}$  là các sai số trước đó được sử dụng để tính toán giá trị hiện tại.

Mục đích của mô hình ARIMA là tìm ra các giá trị của các hệ số  $\phi$  và  $\theta$  để dự đoán giá trị của chuỗi thời gian trong tương lai. Quá trình này thường được thực hiện bằng cách sử dụng các phương pháp ước tính thống kê để ước tính các giá trị của các hệ số này từ dữ liệu lịch sử của chuỗi thời gian.

### B. Linear Regression

Hồi quy tuyến tính (Linear regression) là một phương pháp phân tích thống kê dựa trên việc xác định mối quan hệ giữa hai loại biến bao gồm một biến phụ thuộc (kết quả) và các biến độc lập (dự đoán). Mục đích chính của hồi quy tuyến tính là giúp ta dự đoán được giá trị của biến phụ thuộc dựa trên giá trị của các biến độc lập, kiểm tra xem các biến độc lập có ảnh hưởng thế nào đến giá trị của biến phụ thuộc và các biến độc lập nào là yếu tố quan trọng trong việc dự đoán giá trị của biến phụ thuộc. Hồi quy tuyến tính được sử dụng phổ biến và ứng dụng rộng rãi trong nhiều lĩnh vực khác nhau, có thể kể đến như: Y tế, môi trường, giáo dục, kinh tế... Công

thức của mô hình Linear Regression (hồi quy tuyến tính) có dạng như sau:

- Đối với hồi quy đơn biến:

$$Y = \beta_0 + \beta_1 X$$

- Đối với hồi quy đa biến:

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \dots + \beta_p X_p$$

Trong đó:

- $Y$  là biến đầu ra (target variable) cần được dự đoán.
- $X_1, X_2, \dots, X_p$  là các biến đầu vào (independent variables) được sử dụng để dự đoán  $Y$ .
- $\beta_0, \beta_1, \beta_2, \dots, \beta_p$  là các hệ số (coefficients) tương ứng với mỗi biến đầu vào.
- $\beta_0$  là hệ số chặn (intercept) của đường thẳng hoặc siêu phẳng tuyến tính.
- $\beta_1, \beta_2, \dots, \beta_p$  là hệ số hồi quy (regression coefficients) cho biến đầu vào tương ứng.

### C. Holt-Winters

Mô hình Holt-Winters là một mô hình dự báo chuỗi thời gian bằng cách sử dụng exponential smoothing. Mô hình này chia thành ba phần của một chuỗi thời gian: mức độ (level), xu hướng (trend) và tính thời vụ (seasonality), đồng thời sử dụng chúng để đưa ra các dự báo trong tương lai. Mô hình Holt-Winters được chia thành hai loại: Additive Holt-Winters (AH-W) và Multiplicative Holt-Winters (MH-W)

$$l_t = (1 - \alpha)l_{t-1} + \alpha x_t$$

$$b_t = (1 - \beta)b_{t-1} + \beta(l_t - l_{t-1})$$

$$c_t = (1 - \gamma)c_{t-L} + \gamma(x_t - l_t - b_t)$$

$$y_t = (l_t + b_t)c_t$$

Trong đó:

- $\alpha$  là hệ số làm mịn theo mức độ (level).
- $\beta$  là hệ số làm mịn theo xu hướng (trend).
- $\gamma$  là hệ số làm mịn theo mùa (season).

Với  $0 \leq \gamma \leq 1 - \alpha$

### D. Gradient Boosting regression

Gradient Boosting là một thuật toán máy học được sử dụng để giải quyết các bài toán hồi quy, trong đó các mô hình yếu được xây dựng tuần tự, mỗi mô hình cố gắng sửa lỗi của mô hình trước đó. Giả sử chúng ta có  $M$  mô hình, dự đoán của mô hình Gradient Boosting có thể được biểu diễn như sau:

$$F_M(x) = F_{M-1}(x) + \alpha \cdot h_M(x)$$

Trong đó:

- $F_M(x)$  là dự đoán cuối cùng sau  $M$  lần lặp.
- $F_{M-1}(x)$  là dự đoán từ lần lặp thứ  $M - 1$ .
- $h_M(x)$  là mô hình mới (thường là một cây quyết định) được huấn luyện trên các residuals.
- $\alpha$  là learning rate.

### E. Bagging Model-RandomForest

Random Forests là thuật toán học có giám sát (supervised learning). Nó có thể được sử dụng cho cả phân lớp và hồi quy. Nó cũng là thuật toán linh hoạt và dễ sử dụng nhất. Random forests tạo ra cây quyết định trên các mẫu dữ liệu được chọn ngẫu nhiên, được dự đoán từ mỗi cây và chọn giải pháp tốt nhất bằng cách bỏ phiếu. Nó cũng cung cấp một chỉ báo khá tốt về tầm quan trọng của tính năng. Random forests có nhiều ứng dụng, chẳng hạn như công cụ đề xuất, phân loại hình ảnh và lựa chọn tính năng, phân loại các ứng viên cho vay trung thành, xác định hoạt động gian lận và dự đoán các bệnh. Nó nằm ở cơ sở của thuật toán Boruta, chọn các tính năng quan trọng trong tập dữ liệu. Random Forest hoạt động theo 4 bước:

- Chọn các mẫu ngẫu nhiên từ tập dữ liệu đã cho.
- Thiết lập cây quyết định cho từng mẫu và nhận kết quả dự đoán từ mỗi cây quyết định.
- Bỏ phiếu cho mỗi kết quả dự đoán.
- Chọn kết quả được dự đoán nhiều nhất là dự đoán cuối cùng.

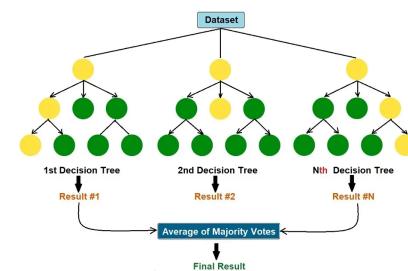


Figure 7. Mô hình của Random Forest

### F. Recurrent Neural Networks (RNN)

Mô hình RNN (Recurrent Neural Network) giữ một trạng thái ẩn (hidden state) và sử dụng nó để lưu trữ thông tin từ các bước thời gian trước đó. Điều này cho phép RNN hiểu và xử lý các dữ liệu dạng chuỗi, chẳng hạn như văn bản, âm thanh hoặc dữ liệu thời gian.

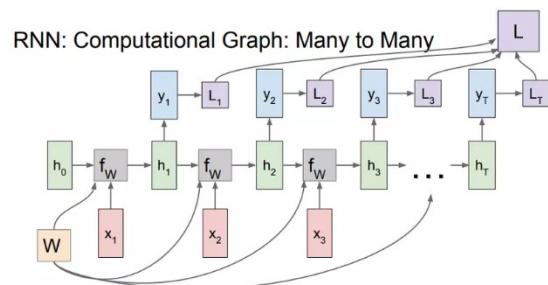


Figure 8. Mô hình Many to Many

### • Mô hình Many to Many của RNN:

- Mạng Neural Network thông thường bao gồm các lớp: lớp đầu vào (input layer)  $x$ , lớp ẩn (hidden layer)

$h$ , và lớp đầu ra (output layer)  $y$ . Tất cả các lớp này được kết nối đầy đủ với nhau.

- Trong RNN, đầu vào tại mỗi thời điểm  $x_t$  sẽ được kết hợp với lớp ẩn từ bước thời gian trước đó  $h_{t-1}$  thông qua hàm  $f_W$  để tính toán ra lớp ẩn hiện tại  $h_t$ . Đầu ra tại mỗi thời điểm  $y_t$  sẽ được tính từ lớp ẩn hiện tại  $h_t$ ,  $W$  là tập các trọng số và nó được xuất hiện ở tất cả các cùm, các  $L_1, L_2, \dots, L_T$  là các hàm mất mát.
- Kết quả của các quá trình tính toán trước đã được "nhớ" bằng cách kết hợp thêm  $h_{t-1}$  để tính toán  $h_t$ . Điều này giúp tăng độ chính xác cho những dự đoán ở hiện tại.
- Hàm  $f_W$  kết hợp  $h_{t-1}$  và  $x_t$  để tính  $h_t$ :

$$h_t = f_W(h_{t-1}, x_t)$$

- Hàm  $f_W$  được định nghĩa cụ thể là hàm tanh, công thức trên sẽ trở thành:

$$h_t = \tanh(W_{hh}h_{t-1} + W_{xh}x_t)$$

- Đầu ra  $y_t$  được tính từ  $h_t$ :

$$y_t = W_{hy}h_t$$

- RNN sử dụng 3 ma trận trọng số là  $W_{hh}$  kết hợp với "bộ nhớ trước"  $h_{t-1}$  và  $W_{xh}$  kết hợp với đầu vào hiện tại  $x_t$  để tính toán "bộ nhớ của bước hiện tại"  $h_t$  từ đó kết hợp với  $W_{hy}$  để tính toán đầu ra  $y_t$ .

## G. LSTM

LSTM (Bộ nhớ dài hạn) là một loại mạng nơ-ron hồi quy (RNN) được giới thiệu bởi Hochreiter và Schmidhuber vào năm 1997. Nó được thiết kế để khắc phục những hạn chế của RNN truyền thống trong việc nắm bắt và ghi nhớ các phụ thuộc dài hạn trong dữ liệu tuần tự.

Ý tưởng chính của LSTM xoay quanh khái niệm về trạng thái tế bào hoặc bộ nhớ. Trạng thái tế bào hoạt động như một băng tải, cho phép thông tin chảy qua mạng lưới trong khi vẫn giữ được thông tin quan trọng qua các chuỗi dài. Tế bào LSTM bao gồm một số thành phần: một cổng đầu vào, một cổng quên, một cổng đầu ra, và một cơ chế cập nhật tế bào. Các công thức để tính các thành phần khác nhau của tế bào

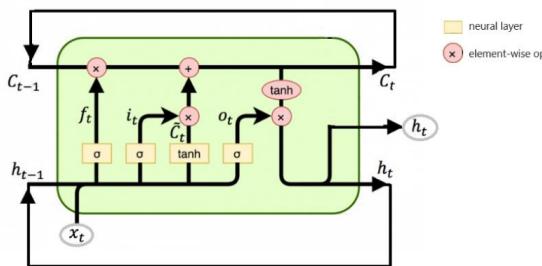


Figure 9. Cấu trúc của LSTM.

LSTM như sau:

- |                    |   |
|--------------------|---|
| Cổng đầu vào:      | $i_t = \sigma(W_i * [h_{t-1}, x_t] + b_i)$        |
| Cổng quên:         | $f_t = \sigma(W_f * [h_{t-1}, x_t] + b_f)$        |
| Cập nhật tế bào:   | $\tilde{C}_t = \tanh(W_c * [h_{t-1}, x_t] + b_c)$ |
| Trạng thái tế bào: | $C_t = f_t * C_{t-1} + i_t * \tilde{C}_t$         |
| Cổng đầu ra:       | $o_t = \sigma(W_o * [h_{t-1}, x_t] + b_o)$        |
| Trạng thái ẩn:     | $h_t = o_t * \tanh(C_t)$                          |

Trong các công thức này,  $W_i, W_f, W_c, W_o$  đại diện cho ma trận trọng số,  $b_i, b_f, b_c, b_o$  đại diện cho vector độ lệch,  $h_{t-1}$  đại diện cho trạng thái ẩn trước đó,  $x_t$  đại diện cho đầu vào tại thời điểm  $t$ , và  $*$  biểu thị phép nhân ma trận.

## H. LIGHTGBM

LightGBM là một thư viện học máy gradient boosting mạnh mẽ và nhanh chóng được phát triển bởi Microsoft. LightGBM sử dụng các thuật toán dựa trên cây quyết định để cải thiện hiệu suất và tốc độ của các mô hình dự đoán.

Ý tưởng chính của LightGBM là tạo ra các cây quyết định dựa trên kỹ thuật gradient boosting. Thay vì xây dựng các cây lớn và phức tạp, LightGBM xây dựng các cây nhỏ và tối ưu chúng để giảm thiểu sai số dự đoán. Một số ưu điểm của LightGBM bao gồm: tốc độ huấn luyện nhanh, khả năng mở rộng lớn, và hỗ trợ nhiều tính năng tiên tiến như xử lý dữ liệu thiếu và hỗ trợ cho các mô hình tuần tự.

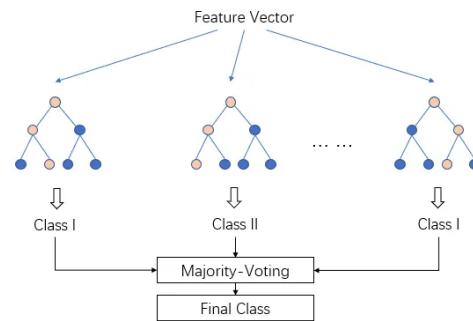


Figure 10. Cấu trúc của LightGBM.

Các công thức để tính các thành phần khác nhau của LightGBM như sau:

$$Hmmtmt : L(y, \hat{y}) = \sum_{i=1}^n \text{loss}(y_i, \hat{y}_i)$$

$$\text{Gradient} : g_i = \frac{\partial L(y_i, \hat{y}_i)}{\partial \hat{y}_i}$$

$$\text{Hessian} : h_i = \frac{\partial^2 L(y_i, \hat{y}_i)}{\partial \hat{y}_i^2}$$

Giá trị phân chia tốt nhất:

$$\text{Split\_value} = \arg \min_s \left[ \frac{(\sum_{i \in I_L} g_i)^2}{\sum_{i \in I_L} h_i + \lambda} + \frac{(\sum_{i \in I_R} g_i)^2}{\sum_{i \in I_R} h_i + \lambda} \right]$$

Trong các công thức này,  $y$  đại diện cho nhãn thực tế,  $\hat{y}$  đại diện cho dự đoán của mô hình,  $g_i$  là gradient,  $h_i$  là Hessian,  $I_L$  và  $I_R$  lần lượt là các chỉ số của các mẫu trong nút trái và phải, và  $\lambda$  là tham số điều chỉnh.

### I. PatchTST

Các phương pháp truyền thống gặp khó khăn trong việc dự báo dài hạn trong dữ liệu chuỗi thời gian. Một trong những giải pháp cho vấn đề này là mô hình Patch-based Time Series Transformer (PatchTST). Mô hình này chia dữ liệu thành các phân đoạn nhỏ hơn (patches) và sử dụng kiến trúc Transformer để nắm bắt các phụ thuộc tầm xa trong các phân đoạn đó.

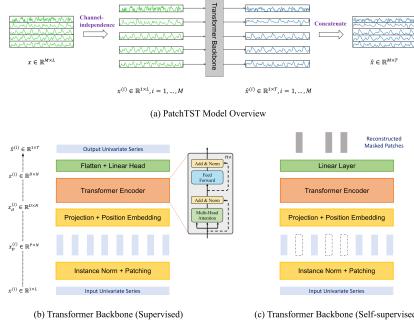


Figure 11. Mô hình PatchTST.

Hình (a): Dữ liệu chuỗi thời gian đa biến được chia thành các kênh khác nhau. Chúng cùng chia sẻ một backbone Transformer, nhưng các quá trình hướng tới (forward processes) là độc lập. Hình (b): Mỗi kênh đơn biến được truyền qua toán tử chuẩn hóa theo từng cá thể (instance normalization) và được chia thành các bản vá. Các bản vá này được sử dụng làm token đầu vào của Transformer. Hình (c): Học biểu diễn tự giám sát có mặt nạ với PatchTST, trong đó các bản vá được chọn ngẫu nhiên và đặt thành bằng không. Mô hình sẽ tái tạo các bản vá bị che khuất.

### J. Gated Recurrent Unit (GRU)

GRU là một mô hình RNN mạng nơ ron và là một biến thể đơn giản của LSTM nhưng thường đạt chất lượng huấn luyện model tương đương và tính toán nhanh đáng kể. GRU có ứng dụng lớn trong xử lý ngôn ngữ tự nhiên, nhận dạng giọng nói, nhận dạng hình ảnh và dự đoán chuỗi thời gian. Với cấu trúc đơn giản hơn LSTM giúp việc đào tạo trở nên dễ dàng với chuỗi liệu dài hơn. GRU hoạt động theo các bước sau:

- Tính toán trạng thái ẩn dự đoán: Sử dụng thông tin đầu vào hiện tại và trạng thái ẩn trước đó để tính toán một trạng thái ẩn dự đoán.
- Cập nhật cổng cập nhật: Sử dụng thông tin đầu vào hiện tại và trạng thái ẩn trước đó để tính toán giá trị của cổng cập nhật.
- Cập nhật cổng đặt lại: Sử dụng thông tin đầu vào hiện tại và trạng thái ẩn trước đó để tính toán giá trị của cổng đặt lại.
- Tính toán trạng thái ẩn: Sử dụng trạng thái ẩn dự đoán, cổng cập nhật, cổng đặt lại và trạng thái ẩn trước đó để tính toán trạng thái ẩn.

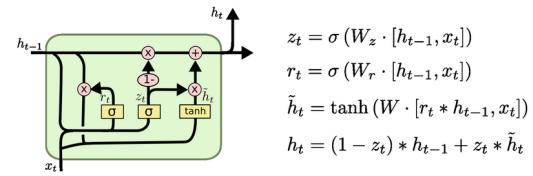


Figure 12. Mô hình GRU.

## V. EXPERIMENT

### A. Evaluation Metrics

1) *Mean Squared Error (MSE)*: là trung bình của bình phương các sai số, tức là sự khác biệt giữa các ước lượng và những gì được đánh giá. Giá trị MSE càng thấp, tức là sự khác biệt giữa giá trị dự báo và giá trị thực tế càng nhỏ thì mô hình dự báo càng tốt. Công thức tính MSE như sau:

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

2) *Root Mean Square Error (RMSE)*: đo lường sai số trung bình của mô hình so với dữ liệu thực tế. RMSE được tính bằng căn bậc hai của trung bình bình phương của sai số giữa giá trị dự đoán và giá trị thực tế. Giá trị RMSE càng nhỏ thì mô hình càng tốt. Công thức tính RMSE như sau:

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2}$$

3) *Mean Absolute Percentage Error (MAPE)*: đo lường tỉ lệ phần trăm trung bình của sai số giữa giá trị dự đoán và giá trị thực tế. MAPE được tính bằng trung bình giá trị tuyệt đối của sai số chia cho giá trị thực tế, nhân với 100. Giá trị MAPE càng nhỏ thì mô hình càng tốt. Công thức tính MAPE như sau:

$$MAPE = \frac{1}{n} \sum_{i=1}^n \left| \frac{y_i - \hat{y}_i}{y_i} \right| \times 100$$

### B. Model Setting

#### 1) Arima:

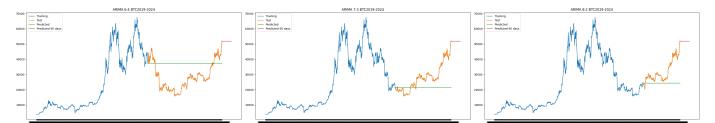


Figure 13. Kết quả dự đoán giá Bitcoin của ARIMA

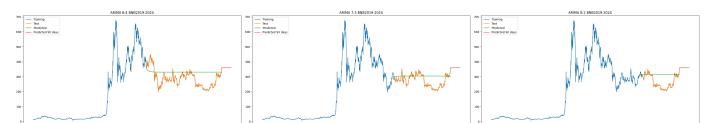


Figure 14. Kết quả dự đoán giá Binance của ARIMA

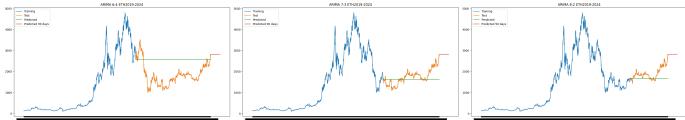


Figure 15. Kết quả dự đoán giá Ethereum của ARIMA

## 2) GRU:



Figure 16. Kết quả dự đoán giá của GRU với tỉ lệ 8:2.

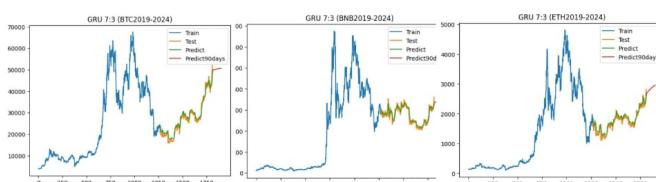


Figure 17. Kết quả dự đoán giá của GRU với tỉ lệ 7:3.

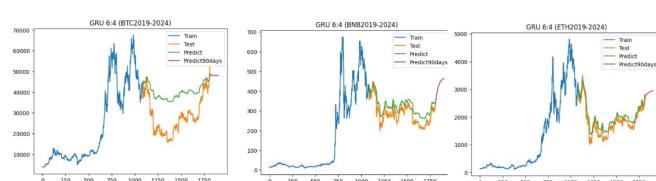


Figure 18. Kết quả dự đoán giá của GRU với tỉ lệ 6:4.

## 3) Gradient Boosting:

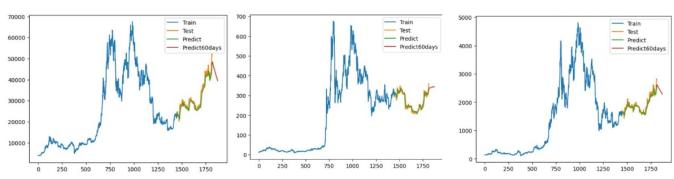


Figure 19. Kết quả dự đoán giá của GB với tỉ lệ 8:2.

Figure 20. Kết quả dự đoán giá của GB với tỉ lệ 7:3.

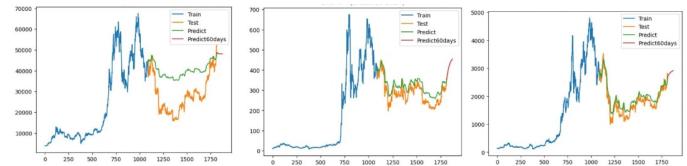


Figure 21. Kết quả dự đoán giá của GB với tỉ lệ 6:4.

## 4) LSTM:

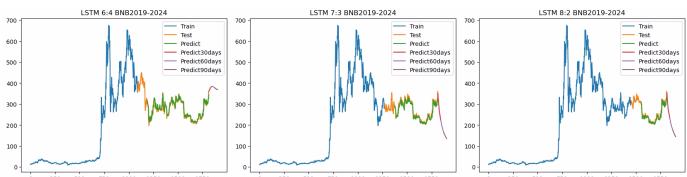


Figure 22. Kết quả dự đoán giá Binance của LSTM.

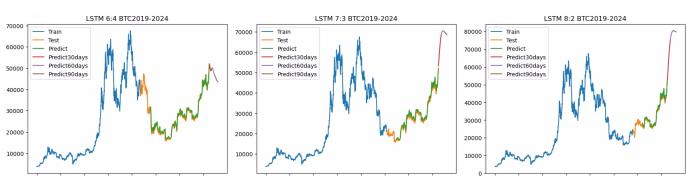


Figure 23. Kết quả dự đoán giá Bitcoin của LSTM.



Figure 24. Kết quả dự đoán giá Ethereum của LSTM.

## 5) LightGBM:

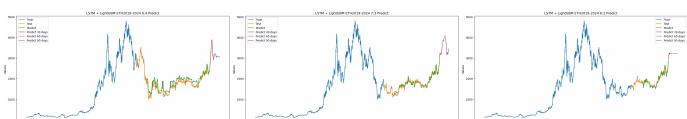


Figure 25. Kết quả dự đoán giá Ethereum của LightGBM.

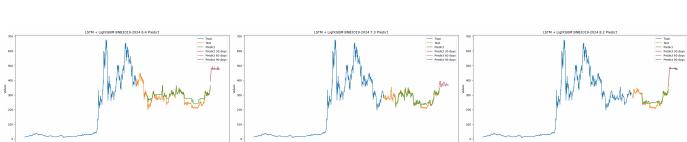


Figure 26. Kết quả dự đoán giá Binance của LightGBM.

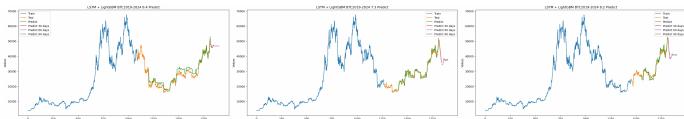


Figure 27. Kết quả dự đoán giá Bitcoin của LightGBM.

### 6) RNN:

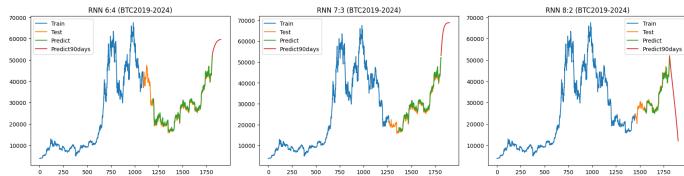


Figure 28. Kết quả dự đoán giá Bitcoin của RNN.

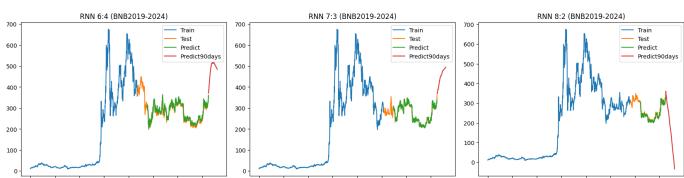


Figure 29. Kết quả dự đoán giá Binance của RNN.

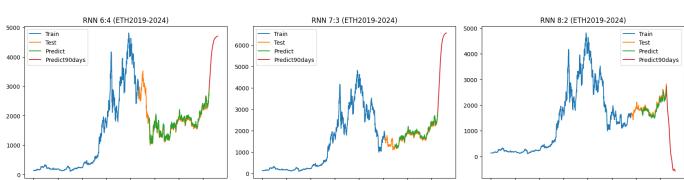


Figure 30. Kết quả dự đoán giá Ethereum của RNN.

### 7) Bagging Model-Random Forest:

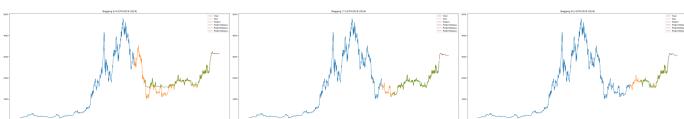


Figure 31. Kết quả dự đoán giá Ethereum của Bagging Model.

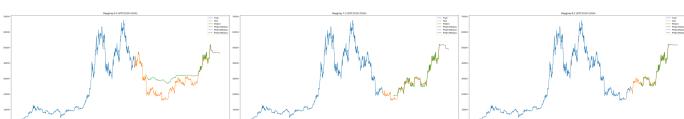


Figure 32. Kết quả dự đoán giá Bitcoin của Bagging Model.



Figure 33. Kết quả dự đoán giá Binance của Bagging Model.

### 8) Linear Regression:

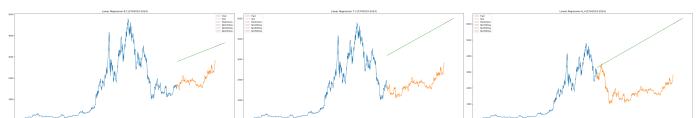


Figure 34. Kết quả dự đoán giá Ethereum của Linear Regression.



Figure 35. Kết quả dự đoán giá Bitcoin của Linear Regression.

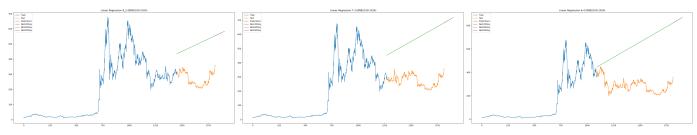


Figure 36. Kết quả dự đoán giá Binance của Linear Regression.

### 9) PatchTST:



Figure 37. Kết quả dự đoán giá Bitcoin của PatchTST.



Figure 38. Kết quả dự đoán giá Ethereum của PatchTST.

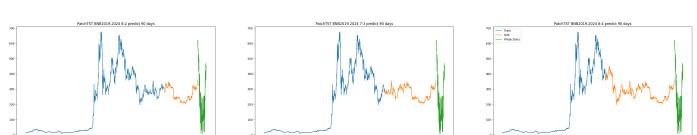


Figure 39. Kết quả dự đoán giá Binance của PatchTST.

### 10) Holt-Winters:

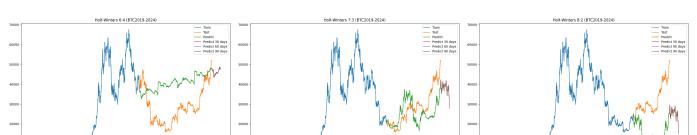


Figure 40. Kết quả dự đoán giá Bitcoin của Holt-Winters.



Figure 41. Kết quả dự đoán giá Binance của Holt-Winters.



Figure 42. Kết quả dự đoán giá Ethereum của Holt-Winters.

11) :

### C. Result

Model	Ratio	BTC			BNB			ETH		
		RMSE	MSE	MAPE	RMSE	MSE	MAPE	RMSE	MSE	MAPE
LR	8:2	13281.713	176403905.368	43.153	234.894	55175.340	89.849	1234.472	1523921.362	65.752
	7:3	30575.467	934859201.08	122.062	304.966	93004.572	112.332	2241.142	5022717.602	132.291
	6:4	43631.231	1903684403.160	165.277	2958.333	8751735.592	167.497	374.496	140247.836	131.028
Holt-Winters	8:2	19494.104	380020103.158	53.556	45.172	2040.467	14.586	654.421	428266.346	27.918
	7:3	8332.365	69428298.686	22.415	48.647	2366.593	15.284	486.518	236699.316	18.258
	6:4	14734.705	217111528.885	56.744	80.598	6496.047	26.587	803.522	645648.138	46.334
RNN	8:2	34092.126	1162273044.790	7467700.375	247.497	61255.012	69187.291	1913.928	3663119.549	493447.914
	7:3	31362.461	983603991.239	8265002.602	275.301	75790.684	71076.489	1911.257	3652903.898	537458.234
	6:4	28665.271	821697745.549	8379727.686	283.409	80320.761	72649.295	1793.959	3218289.393	543341.059
ARIMA	8:2	10384.636	107840669.08	0.214	62.409	3894.94	0.213	363.658	132247.113	0.126
	7:3	10413.778	108446775.748	0.247	49.116	2412.427	0.16	365.889	133874.522	0.155
	6:4	12070.238	145690644.85	0.464	68.56	4700.521	0.226	883.74	780996.861	0.515
LightGBM	8:2	1473.369	2170816.998	0.036	20.534	421.661	0.072	73.275	5369.22	0.028
	7:3	1127.487	1271225.997	0.030	15.441	238.423	0.05	82.763	6849.698	0.039
	6:4	2388.382	5704367.035	0.080	25.827	667.052	0.085	171.231	29320.010	0.094
GRU	8:2	1871.52	3391223	0.038	12.9796	168.47	0.0309	94.8131	8989.53	0.0343
	7:3	1550.66	2404555	0.043	14.9132	222.40	0.0384	96.6455	9340.36	0.040
	6:4	10072.2	101447619	0.3925	21.0943	966.85	0.1026	150.521	22656.89	0.069
Gradient Boosting	8:2	<b>4.6388</b>	<b>21.5186</b>	<b>0.0078</b>	<b>4.1578</b>	<b>17.2879</b>	<b>71666</b>	<b>4.3371</b>	<b>18.8108</b>	<b>63776</b>
	7:3	<b>4.7361</b>	<b>22.4314</b>	<b>0.0079</b>	<b>4.6055</b>	<b>21.2106</b>	<b>48754</b>	<b>4.5189</b>	<b>20.4210</b>	<b>25883</b>
	6:4	<b>4.6958</b>	<b>22.0514</b>	<b>0.0084</b>	<b>4.5139</b>	<b>20.3750</b>	<b>40860</b>	<b>4.8312</b>	<b>22.3450</b>	<b>48824</b>
BG	8:2	1105.9622	1189727729.5284	197920.5664	11.03351	66390.528	2549.6907	61.3195	3856646.1734	11865.5426
	7:3	1319.7837	931339246.2178	292611.219	10.4522	76381.6755	2179.373	51.16026	3410236.8333	10227.606
	6:4	6588.2879218	1078670687.0293	1682830.805	37.56211	90412.0888	7637.5616	177.8791	3343894.2255	36970.265
LSTM	8:2	35206.264	1239481042.236	7710365.511	247.261	61138.072	69184.116	1918.097	3679095.877	494731.584
	7:3	31390.871	985386792.539	8261187.272	267.204	71397.962	69034.377	1828.836	3344642.186	514292.382
	6:4	28619.752	819090215.224	8372540.489	270.585	73216.161	69372.351	1754.844	3079477.977	532363.836
PatchTST	8:2	28742.968	24872.653	1.798	226.754	201.147	0.512	1680.91	1279.968	2.045
	7:3	30777.957	27351.518	2.075	252.552	221.203	0.652	1719.241	1528.288	2.325
	6:4	33879.124	31856.856	2.987	289.531	267.517	0.756	2078.865	1618.786	2.856

Table II

KẾT QUẢ CHỈ SỐ ĐÁNH GIÁ RMSE, MAPE VÀ MSE TRÊN BA DATASET

## VI. CONCLUSION

Từ kết quả so sánh của Bảng II, dựa vào độ đo RMSE, MAPE và MSE, có thể thấy mô hình Gradient Boosting có kết quả tốt nhất trong các mô hình trên cả 3 dataset. Kết quả nghiên cứu này cho thấy thuật toán Gradient Boosting là phù hợp nhất trong việc dự đoán giá tiền ảo trong tương lai của 3 loại tiền ảo Bitcoin, Binance và Ethereum. Nghiên cứu này nhấn mạnh sự cần thiết của việc sử dụng nhiều phương pháp mô hình hóa trong phân tích tài chính. Do đó, nhóm đã xem xét trên 3 độ đo và kết luận rằng mô hình Gradient Boosting là lựa chọn tốt nhất nhằm dự báo giá 30 ngày tiếp theo của cả ba loại tiền ảo.

## ACKNOWLEDGMENT

Với lòng biết ơn sâu sắc nhất, nhóm xin gửi lời cảm ơn chân thành đến quý Thầy Cô và các giảng viên đã giúp đỡ chúng em trong quá trình thực hiện đề tài này. Nhóm cũng xin gửi lời cảm ơn đến thầy Nguyễn Đình Thuân – giảng viên lý thuyết môn Phân tích dữ liệu kinh doanh và anh Nguyễn Minh Nhựt – trợ giảng của môn đã tận tình hướng dẫn và hỗ trợ nhóm trong suốt quá trình làm đồ án. Nhờ sự hướng dẫn của các thầy và anh, nhóm đã tiếp thu được nhiều kiến thức bổ ích và hoàn thành đồ án một cách tốt nhất.

## REFERENCES

- [1] Nakamoto, S. Bitcoin: A Peer-to-Peer Electronic Cash System. Technical Report. Available online: <https://bitcoin.org/bitcoin.pdf>.
- [2] F. Reid and M. Harrigan, "An Analysis of Anonymity in the Bitcoin System," Tech. Rep., 2011.[Online]. Available: <http://arxiv.org/abs/1107.4524>.
- [3] P. N. Sureshbhai, P. Bhattacharya, and S. Tanwar, "aRuNa: A blockchain-based sentiment analysis framework for fraud cryptocurrency schemes," in Proc. IEEE Int. Conf. Commun.
- [4] Suhwan Ji, Jongmin Kim and Hyeonseung Im, "A Comparative Study of Bitcoin Price Prediction Using Deep Learning" in Mathematics 2019, 7, 898; doi:10.3390/math7100898.
- [5] SUDEEP TANWAR 1, (Senior Member, IEEE), NISARG P. PATEL 1 ,SMIT N. PATEL 2, JIL R. PATEL 3, GULSHAN SHARMA 4, AND INNOCENT E. DAVIDSON 4, (Senior Member, IEEE), "Deep Learning-Based Cryptocurrency Price Prediction Scheme With Inter-Dependent Relations" in Digital Object Identifier 10.1109/ACCESS.2021.3117848.
- [6] Financial Platform and News Website. Accessed: 2008. [Online]. Available: <https://www.investing.com/>.
- [7] V. Derbentsev, N. Datsenko, O. Stepanenko, and V. Bezkorovainyi, 'Forecasting cryptocurrency prices time series using machine learning approach,' SHS Web Conf., vol. 65, Jan. 2019, Art. no. 02001.
- [8] Iqbal H. Sarker, Machine Learning: Algorithms, Real-World Applications and Research. Available: <https://link.springer.com/article/10.1007/s42979-021-00592-x> Directions
- [9] Cristescu, R. and Popescu, M. (2019). Short-term cryptocurrency price prediction using gradient boosting machines. IEEE Xplore, 1-4. Available: <https://ieeexplore.ieee.org/document/9964870>
- [10] Huang, S., Wang, W., Dong, Z. and Wang, Y. (2020). LightGBM and LSTM Based Cryptocurrency Price Prediction. Available: <https://arxiv.org/abs/2403.03410>.
- [11] Shen, Z., Wang, Y., and Liu, R. (2018). Short-term and long-term cryptocurrency price prediction using Holt-Winters exponential smoothing. IEEE Transactions on Computational Intelligence and AI in Robotics and Automation, 4(1), 74-81. Available: <https://ieeexplore.ieee.org/document/9257664>.
- [12] Li, Y., Zhang, W., and Liu, P. (2023). PatchTST: A Transformer-based Spatio-Temporal Shifted Transformer for Short-Term Bitcoin Price Forecasting. Available: <https://arxiv.org/abs/2303.04983>