DỰ BÁO GIÁ BITCOIN BẰNG KẾT HỢP MÔ HÌNH ARIMA VÀ MẠNG NƠRON

Lê Hữu Vinh[1], Nguyễn Đình Thuân[2]

[1,2] Trường Đại học Công nghệ Thông tin, Đại học Quốc gia TP. Hồ Chí Minh

vinhlh.10@grad.uit.edu.vn, thuannd@uit.edu.vn

TÓM TẮT*— Trong những năm gần đây, Bitcoin nổi lên như là một đồng tiền ảo được sử dụng rộng rãi nhất trên thế giới. Bitcoin có thể được dùng để thanh toán trực tuyến hoặc đơn giản là một kênh đầu tư sinh lời. Bitcoin gần như không phụ thuộc vào biến động của thị trường hoặc sự điều chỉnh của chính phủ. Giá Bitcoin thay đổi thường xuyên nên vấn đề dự báo gặp nhiều thử thách. Trong bài báo này, chúng tôi thử nghiệm một số mô hình dự báo như ARIMA, mạng nơron, kết hợp ARIMA và mạng nơron để dự báo giá đóng cửa (USD) của đồng Bitcoin trong ngày tiếp theo. Dữ liệu giá đóng cửa của đồng Bitcoin được thu thập từ ngày 28/04/2013 đến ngày 7/11/2017 gồm 1655 ngày trên website của cộng đồng Kaggle. Kết quả dự báo của các mô hình sẽ được so sánh để xem xét mô hình nào phù hợp hơn trong việc dự báo giá Bitcoin.*

Từ khóa*— Chuỗi thời gian, ARIMA, mạng nơron, dự báo Bitcoin*

I. GIỚI THIỆU

Vào tháng 10 năm 2008, đồng tiền ảo Bitcoin được Satoshi Nakamoto lần đầu tiên giới thiệu trong báo cáo "Bitcoin: A Peer-to-Peer Electronic Cash System" [1]. Năm 2009, Nakamoto đã phát hành phần mềm để tạo ra Bitcoin và đến nay đã có một cộng đồng rộng lớn sử dụng Bitcoin trên khắp thế giới. Bitcoin là một loại tiền điện tử, các giao dịch của nó hoàn toàn không thông qua các tổ chức tài chính như ngân hàng, quỹ đầu tư. Ưu điểm lớn nhất của Bitcoin là được kiểm soát bởi thuật toán và giúp minh bạch hóa các giao dịch. Nakamoto tạo ra Bitcoin với mong muốn đồng tiền ảo này có thể thay thế các loại tiền tệ đang được giao dịch thông qua ngân hàng và tốn nhiều chi phí quản lý.

Với những đặc trưng của mình, Bitcoin gần như không phụ thuộc vào biến động của thị trường hoặc sự điều chỉnh của chính phủ. Giá của Bitcoin biến động thường xuyên dẫn đến việc dự báo gặp nhiều khó khăn. Nhưng đó cũng là động lực cho các nghiên cứu trong dự báo giá Bitcoin. Devavrat Shah [2] trình bày phương pháp hồi quy Bayes để dự đoán sự thay đổi giá của Bitcoin sau mỗi 10 giây. Dựa trên phương pháp này, tác giả đã đưa ra một chiến lược đơn giản để giao dịch Bitcoin. Với chiến lược đó, tác giả thực hiện 2872 các giao dịch mua bán Bitcoin, lợi nhuận đạt được trong 50 ngày khoảng 89%.

Siddhi Velankar [3] cố gắng dự báo giá của Bitcoin trên cơ sở xem xét các yếu tố ảnh hưởng đến giá trị của Bitcoin. Trong giai đoạn đầu tiên của nghiên cứu, tác giả tìm hiểu và lựa chọn các đặc trưng ảnh hưởng đến giá Bitcoin. Tác giả thu thập dữ liệu giá Bitcoin từ Quandl và CoinMarketCap. Song song đó, Siddhi Velankar [3] tìm hiểu hai phương pháp hồi quy Bayes và GLM/Random forest để thực nghiệm sau khi đã sử dụng các phương pháp chuẩn hóa dữ liệu như chuẩn hóa log, z-score, độ lệch chuẩn, Box-Cox hoặc sử dụng hàm 'normc' trong MATLAB để chuẩn hóa dữ liệu giá Bitcoin. Trong công trình nghiên cứu tiếp theo, tác giả sẽ sử dụng hồi quy Bayes và GLM/Random forest để thực nghiệm nhằm tìm ra phương pháp cho kết quả dự báo tốt nhất trên dữ liệu giá Bitcoin.

Ngoài mô hình hồi quy, các mô hình máy học cũng được áp dụng trong dự báo Bitcoin. João Almeida [4] đã áp dụng mạng nơron nhân tạo để dự báo xu hướng giá Bitcoin trong ngày kế tiếp dựa vào giá và khối lượng giao dịch của Bitcoin trong những ngày trước đó. Các mô hình mạng noron được cài đặt và thực nghiệm với thư viện Theano và công cụ MATLAB trên dữ liệu giá Bitcoin thu thập từ website Quandl. Thực nghiệm cho thấy việc thêm khối lượng giao dịch Bitcoin để làm giá trị đầu vào cho mạng nơron không phải lúc nào cũng làm tăng độ chính xác dự báo.

Trong một nghiên cứu gần đây, Huisu Jang [5] đề xuất sử dụng mạng nơron Bayes để phân tích biến động của giá Bitcoin. Song song đó, tác giả cũng lựa chọn một số đặc trưng từ thông tin Blockchain có liên quan đến sự cung và cầu của Bitcoin để cải thiện kết quả dự báo. Tác giả thực nghiệm mạng nơron Bayes và một số phương pháp tuyến tính và phi tuyến tính khác trên dữ liệu giá Bitcoin. Kết quả thực nghiệm cho thấy, mạng nơron Bayes thực hiện tốt việc dự báo giá Bitcoin và mô tả được sự biến động lớn của giá Bitcoin.

Việc dự báo giá Bitcoin đang được cộng đồng nghiên cứu rất quan tâm bởi những lợi ích về giá trị kinh tế mà loại tiền điện tử này mang lại. Bài báo này sẽ áp dụng các mô hình dự báo chuỗi thời gian như ARIMA, mạng nơron để dự báo giá Bitcoin trong ngày tiếp theo. Những mô hình dự báo như ARIMA, mạng FFNN (Feed Forward Neural Network), mạng CNN (Convolutional Neural Network), kết hợp ARIMA-FFNN, kết hợp ARIMA-CNN sẽ được cài đặt và thực nghiệm nhằm xem xét mô hình nào phù hợp hơn cho dự báo giá Bitcoin.

II. CÁC MÔ HÌNH DỰ BÁO CHUỖI THỜI GIAN

A. Mô hình ARIMA (Autoregressive Integrated Moving Average)

Mô hình ARIMA là một trong những mô hình được sử dụng rộng rãi nhất trong dự báo chuỗi thời gian. Mô hình này được Box và Jenkins giới thiệu lần đầu vào năm 1970 trong quyển sách "Time-Series Analysis: Forecasting

and Control". Theo đó, một mô hình ARIMA là sự kết hợp của quá trình tự hồi quy (AR) và quá trình trung bình trượt (MA) trong chuỗi thời gian dừng. Một mô hình ARIMA (p, d, q) gồm 3 thành phần:

- AR (Autoregression) là mô hình tự hồi quy biểu diễn sự phụ thuộc của điểm thời gian đang xét vào một số điểm thời gian trước đó (p).
- I (Integrated) là số lần lấy sai phân để làm cho chuỗi thời gian có tính dừng (d).
- MA (Moving Average) là mô hình trung bình trượt biểu diễn sự phụ thuộc của điểm thời gian đang xét vào một vài số hạng sai số ngẫu nhiên và giá trị trung bình của các điểm thời gian trước đó (q).

Một mô hình tự hồi quy AR bậc p – AR(p) là một quá trình tuyến tính được xác định bởi phương trình:

$$y_t = c + \sum_{i=1}^{p} \emptyset_i y_{t-i} + \varepsilon_t$$

với y_t là giá trị tại thời điểm t, c là hằng số, $\emptyset_i$ là hệ số tự tương quan tại các thời điểm t-1, t-2, ..., t-p trước đó và ε_t là một số hạng sai số ngẫu nhiên không tương quan, có giá trị trung bình bằng 0 và phương sai không đổi σ_ε^2.

Một mô hình trung bình trượt MA bậc q – MA(q) được xác định bởi phương trình:

$$y_t = \mu + \sum_{i=0}^{q} \theta_i \, \varepsilon_{t-i}$$

với μ là giá trị trung bình của chuỗi thời gian, θ_i là trọng số của các số hạng sai số ngẫu nhiên tại các thời điểm t, t-1, …, t-q, ε_t là một số hạng sai số ngẫu nhiên không tương quan, có giá trị trung bình bằng 0 và phương sai không đổi σ_ε^2.

Giả sử một chuỗi thời gian tuân theo cả quá trình tự hồi quy và trung bình trượt, chúng ta có thể kết hợp hai mô hình lại với nhau để biểu diễn chuỗi thời gian bởi phương trình:

$$y_t = c + \sum_{i=1}^{p} \emptyset_i y_{t-i} + \varepsilon_t + \sum_{i=0}^{q} \theta_i \, \varepsilon_{t-i}$$

với $\emptyset_i \neq 0$, $\theta_i \neq 0$ và $\sigma_\varepsilon^2 > 0$. Trên thực tế, mô hình ARIMA có khả năng áp dụng trên các chuỗi thời gian không có tính dừng nếu chuỗi thời gian này được lấy sai phân bậc d để chuyển đổi thành chuỗi thời gian có tính dừng.

Để xây dựng mô hình ARIMA cho dự báo chuỗi thời gian, Box và Jenkins đề xuất một phương pháp luận gồm bốn bước sau:

- Bước 1. Nhận dạng mô hình ARIMA (p, d, q): tìm các giá trị thích hợp của p và q thông qua hàm tự tương quan riêng phần (PACF) và hàm tự tương quan (ACF). Giá trị d là số lần lấy sai phân để chuỗi thời gian có tính dừng.
- Bước 2. Ước lượng mô hình: dựa vào dữ liệu chuỗi thời gian để tìm các tham số của mô hình tự hồi quy và trung bình trượt.
- Bước 3. Kiểm định mô hình: xem xét mô hình đã xây dựng có phù hợp với dữ liệu hay không. Các tiêu chuẩn AIC (Akaike information criterion), BIC (Bayesian information criterion), HQIC (Hannan–Quinn information criterion) hỗ trợ việc kiểm định này. Ngoài ra, một cách kiểm định đơn giản khác là xem xét phần dư ước lượng từ mô hình có tính ngẫu nhiên thuần túy hay không. Nếu có thì mô hình có thể được chấp nhận, còn không thì phải thực hiện lại bước nhận dạng mô hình và ước lượng mô hình đến khi nào tìm được mô hình có thể chấp nhận được.
- Bước 4. Dự báo chuỗi thời gian: sử dụng mô hình vừa tìm được để dự báo giá trị tại các thời điểm t trong tương lai của chuỗi thời gian.

B. Mô hình FFNN (Feedforward Neural Network)

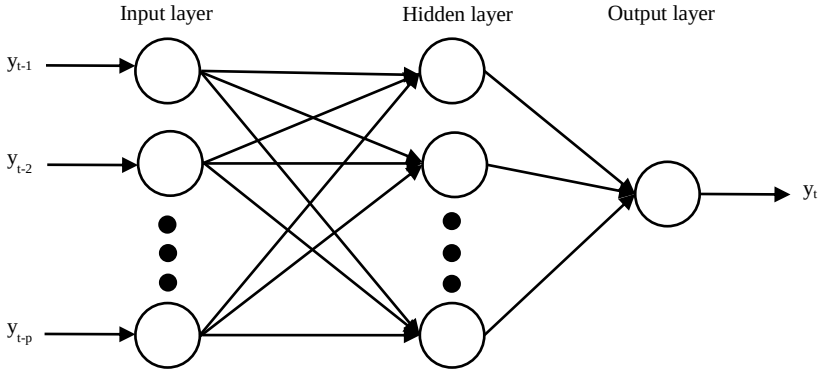
Trong những năm gần đây, nhiều nghiên cứu đã áp dụng các mô hình mạng nơron nhân tạo trong dự báo chuỗi thời gian và đạt được một số kết quả nhất định. Bogdan Oancea [6] đã cài đặt mô hình mạng nơron với hai kiến trúc mạng là FFNN và RNN (Recurrent Neural Network) cho việc dự báo chuỗi thời gian. Tác giả chạy thực nghiệm trên tập dữ liệu tỷ giá hối đoái giữa đồng EUR/RON và USD/RON. Đầu tiên, tác giả chuẩn hóa dữ liệu bằng công thức logarit tự nhiên để nâng cao độ chính xác dự báo. Sau đó, mô hình FFNN được xây dựng với 20 nơron ở lớp đầu vào (input layer), 40 nơron ở lớp ẩn (hidden layer) và 1 nơron ở lớp đầu ra (output layer) là giá trị dự báo cho thời gian tiếp theo t+1. Tác giả chia tập dữ liệu với 80% cho huấn luyện (training) và 20% cho thử nghiệm (testing). Kế tiếp, mạng RNN được cài đặt với 20 nơron ở lớp đầu vào, 10 nơron trong lớp ẩn hồi quy và 1 nơron ở lớp đầu ra. Sau khi thực nghiệm, tác giả khẳng định mô hình RNN cho kết quả dự báo tốt hơn FFNN trên tập dữ liệu tỷ giá hối đoái.

M. Raeesi [7] sử dụng mạng nơron FFNN để dự báo dữ liệu giao thông ở thành phố Monroe, bang Louisiana, Hoa Kỳ. Nghiên cứu này đề xuất một mạng nơron sử dụng dữ liệu giao thông của ngày hôm nay, ngày hôm qua, tuần trước, hai tuần trước, ba tuần trước và một tháng trước để làm đầu vào cho dự báo lưu lượng giao thông của ngày mai. Kết quả thực nghiệm cho thấy mô hình mạng nơron đã xây dựng có thể được sử dụng cho dự báo giao thông tại thành phố Monroe. Tuy nhiên, một vài trường hợp có kết quả dự báo với sai số lớn do những yếu tố bất thường tác động như tai nạn, thời tiết xấu,… Kumar Abhishek [8] cũng sử dụng mạng nơron FFNN với giải thuật lan truyền ngược (back-

propagation) trong dự báo chứng khoán trên tập dữ liệu của tập đoàn Microsoft từ 1/1/2011 đến 31/12/2011 gồm 2 lớp đơn giản trong mạng (10 nơron lớp đầu vào, 1 nơron lớp đầu ra), độ chính xác dự báo lên đến 99%.

Mạng nơron nhân tạo lấy ý tưởng từ việc mô phỏng hoạt động của não bộ con người. Mạng nơron nhân tạo có nhiều kiến trúc khác nhau như mạng nơron truyền thẳng, mạng nơron tích chập, mạng nơron hồi quy. Trong đó, mạng nơron truyền thẳng (FFNN) bao gồm một lớp đầu vào (input layer), một hoặc nhiều lớp ẩn (hidden layer), lớp đầu ra (output layer). Số đặc trưng của tập dữ liệu sẽ tương ứng với số nơron trong lớp đầu vào. Tất cả các nơron này được kết nối với mỗi nơron trong lớp ẩn thông qua các đường liên kết gọi là "khớp thần kinh". Mỗi "khớp thần kinh" sẽ được gán một trọng số (weight). Các trọng số này sẽ được điều chỉnh trong quá trình học của mạng nơron nhân tạo để mô hình hóa mối liên hệ giữa lớp đầu vào và đầu ra.

Trong lớp ẩn, mỗi nơron sẽ thực hiện một hàm kích hoạt (thường là hàm sigmoid, tanh hoặc relu) có chức năng tính toán các giá trị đầu vào kết hợp với trọng số để gửi đến lớp đầu ra. Lớp đầu ra sẽ nhận các giá trị từ lớp ẩn và tính giá trị đầu ra của mô hình. Các giá trị thực tế trong tập huấn luyện và giá trị được tính ra bởi mô hình mạng nơron sẽ có sự sai số (độ lỗi). Để mạng nơron có thể mô hình mối liên hệ giữa lớp đầu vào và đầu ra một cách tốt nhất thì mạng nơron sẽ trải qua một quá trình học hỏi. Bản chất của quá trình này là mô phỏng độ lỗi dự báo bởi một phương trình và tìm giá trị cực tiểu của phương trình này thông qua việc cập nhật các trọng số giữa các nơron trong mạng. Quá trình học hỏi này thường được thực hiện với giải thuật lan truyền ngược và kỹ thuật gradient descent.



Hình 1. Kiến trúc mạng nơron truyền thẳng cho dự báo chuỗi thời gian [7]

Trong dự báo chuỗi thời gian, mô hình mạng nơron FFNN sử dụng đặc trưng cho lớp đầu vào là các giá trị ở những điểm thời gian trước điểm thời gian dự báo. Mối liên hệ giữa giá trị đầu ra (y_t) và các giá trị đầu vào (y_{t-1}, y_{t-2}, …, y_{t-p}) được mô hình bởi phương trình [10]:

$$y_t = \alpha_0 + \sum_{j=1}^{q} \alpha_j g(\beta_{0j} + \sum_{i=1}^{p} \beta_{ij}\, y_{t-i}) + \varepsilon_t$$

trong đó, α_j (j=0, 1, 2, ..., q) và β_{ij} (i=0, 1, 2, ..., p, j=1, 2, ..., q) là các tham số của mô hình, p là số nơron lớp đầu vào và q là số nơron lớp ẩn, ε_t là sai số. Hàm kích hoạt được sử dụng trong các nơron lớp ẩn như hàm sigmoid [10]:

$$g(x) = \frac{1}{1 + \exp(-x)}$$

Do đó, mô hình mạng nơron FFNN trong dự báo chuỗi thời gian là một mô hình phi tuyến mô tả mối quan hệ giữa các giá trị trong quá khứ (y_{t-1}, y_{t-2}, ..., y_{t-p}) và giá trị tương lai (y_t) [10]:

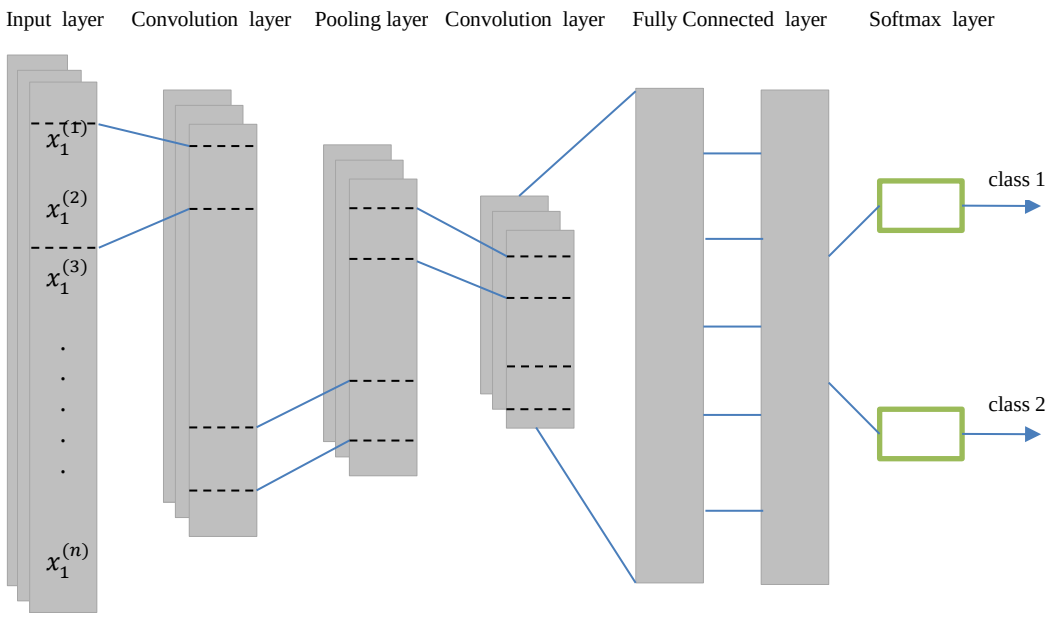
$$y_t = f(y_{t-1}, y_{t-2}, ..., y_{t-p}, w) + \varepsilon_t$$

với w là một vector chứa tất cả các tham số của mô hình FFNN, f là một hàm số được xác định bởi cấu trúc mạng và các tham số.

Việc chọn số nơron đầu vào p và số nơron của lớp ẩn q phụ thuộc vào tập dữ liệu huấn luyện. Mỗi tập dữ liệu chuỗi thời gian khi huấn luyện với mô hình mạng nơron sẽ có p, q khác nhau. Chọn p, q để tìm được mô hình dự báo chuỗi thời gian tốt nhất phải qua thực nghiệm và so sánh giữa các mô hình.

C. Mô hình CNN (Convolutional Neural Network)

Mô hình mạng nơron tích chập (CNN) là một mô hình mạng nơron truyền thẳng (FFNN). Ngoài lớp đầu vào, lớp đầu ra, mạng nơron tích chập còn có các lớp đặc trưng như lớp tích chập (convolutional), lớp lấy mẫu (pooling), lớp kết nối đầy đủ (fully connected) giống như lớp ẩn trong mạng FFNN. Lớp đầu vào của mô hình CNN là một ma trận có số chiều [rộng x cao x sâu]. Tùy vào các bài toán cụ thể, ma trận đầu vào có thể bị giảm một số chiều. Với mạng CNN, ma trận đầu vào khi qua lớp tích chập sẽ thực hiện phép tích chập (convolutional) với các bộ lọc (filters) để tạo ra một ma trận có số chiều nhỏ hơn ma trận đầu vào. Ma trận vừa được tạo ra tiếp tục thực hiện phép lấy mẫu (pooling) để giúp rút trích đặc trưng quan trọng từ lớp đầu vào. Ma trận kết quả của phép lấy mẫu sẽ được làm phẳng (flatten) để làm đầu vào cho lớp kết nối đầy đủ thực hiện như một mạng nơron truyền thẳng và đưa ra kết quả dự báo.



Hình 2. Kiến trúc mạng nơron tích chập trong thực nghiệm của Sheng Chen [9]

Mô hình CNN được đề xuất với mục tiêu ban đầu nhằm phục vụ cho công việc xử lý, nhận dạng hình ảnh. Thông thường, đầu vào của mạng CNN là một ma trận hai chiều hoặc ba chiều nhưng khi áp dụng cho dự báo chuỗi thời gian thì mạng CNN phải xử lý trên dữ liệu mảng một chiều. Tuy nhiên, với những kết quả tốt mạng lại trong xử lý ảnh, một số nghiên cứu đã thử nghiệm áp dụng mạng nơron tích chập trong dự báo chuỗi thời gian. Chẳng hạn, trong nghiên cứu của Sheng Chen [9], mô hình CNN được áp dụng cho việc dự báo sự tăng giảm của chứng khoán. Mạng CNN được huấn luyện trên tập dữ liệu chứng khoán với đầu vào gồm các yếu tố được cho là sẽ ảnh hưởng đến sự tăng giảm của giá chứng khoán là giá mở cửa, giá đóng cửa, giá cao nhất, giá thấp nhất và khối lượng giao dịch trong ngày. Với thực nghiệm của mình, tác giả cho rằng mô hình CNN cũng đáng tin cậy trong dự báo giá chứng khoán, là một phương pháp rất đáng để thử nghiệm.

III. MÔ HÌNH DỰ BÁO KẾT HỢP ARIMA VÀ MẠNG NƠRON

Mô hình ARIMA và các mô hình mạng nơron đều đã đạt được những thành công nhất định trong dự báo chuỗi thời gian. Tuy nhiên, mỗi mô hình thường chỉ phù hợp với một số tập dữ liệu nhất định. Mô hình ARIMA phù hợp với dự báo dữ liệu chuỗi thời gian dạng tuyến tính, còn mô hình mạng nơron lại phù hợp với dự báo dữ liệu chuỗi thời gian dạng phi tuyến tính. Do đó, mô hình kết hợp giữa ARIMA và mạng nơron được đề xuất với kỳ vọng có thể giúp tăng độ chính xác của dự báo trong các ứng dụng thực tế. Ý tưởng này được G. Peter Zhang giới thiệu trong nghiên cứu [10]. Mô hình kết hợp được tác giả thực nghiệm trên ba tập dữ liệu là Wolf's sunspot, Canadian lynx và tỷ giá hối đoái giữa British pound/US dollar. Kết quả thực nghiệm cho thấy các mô hình kết hợp có độ lỗi dự báo ít hơn đáng kể so với từng mô hình ARIMA và mạng nơron riêng lẻ.

Ý tưởng của mô hình này dựa trên việc xem xét dữ liệu chuỗi thời gian là sự kết hợp giữa thành phần tuyến tính và phi tuyến tính. Hai thành phần này được biểu diễn qua phương trình:

$$y_t = L_t + N_t$$

với y_t là giá trị của chuỗi thời gian, L_t là thành phần tuyến tính, N_t là thành phần phi tuyến tính.

Để dự báo giá trị của chuỗi thời gian, mô hình ARIMA được sử dụng để dự báo cho thành phần tuyến tính. Những giá trị dự báo lỗi từ mô hình ARIMA sẽ được dự báo bằng mạng nơron. Gọi e_t là giá trị còn lại sau khi sử dụng mô hình ARIMA để dự báo, e_t được xác định bởi phương trình:

$$e_t = y_t - \widehat{L_t}$$

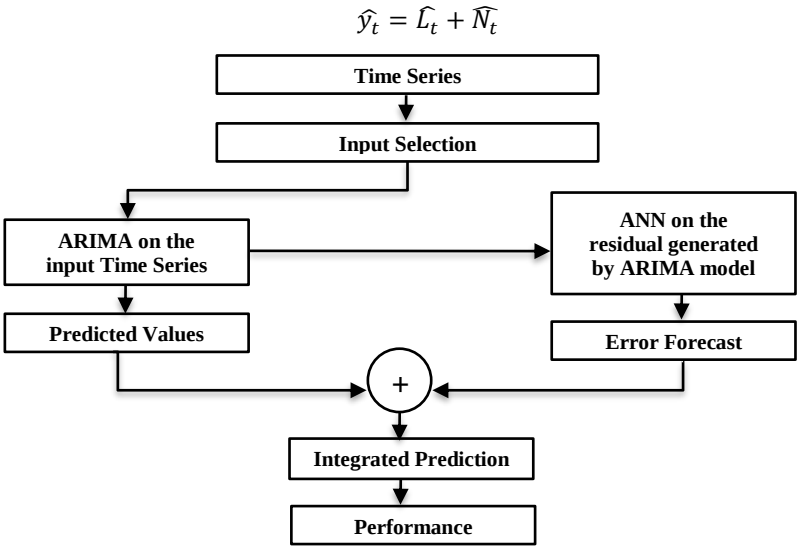
với $\widehat{L_t}$ là giá trị dự báo cho thành phần tuyến tính tại thời điểm t.

Mô hình mạng nơron được dùng để dự báo giá trị còn lại e_t sau khi dự báo bằng mô hình ARIMA sẽ được mô hình hóa bởi một hàm số:

$$e_t = f(e_{t-1}, e_{t-2}, \dots, e_{t-n}) + \varepsilon_t$$

với f là một hàm phi tuyến được xác định bằng mạng nơron, ε_t là giá trị ngẫu nhiên tại thời điểm t.

Ký hiệu $\widehat{N_t}$ là giá trị dự báo cho thành phần phi tuyến tính. Kết quả giá trị dự báo tại thời điểm t ($\widehat{y_t}$) được tính bởi phương trình:



Hình 3. Mô hình kết hợp ARIMA và mạng nơron [12]

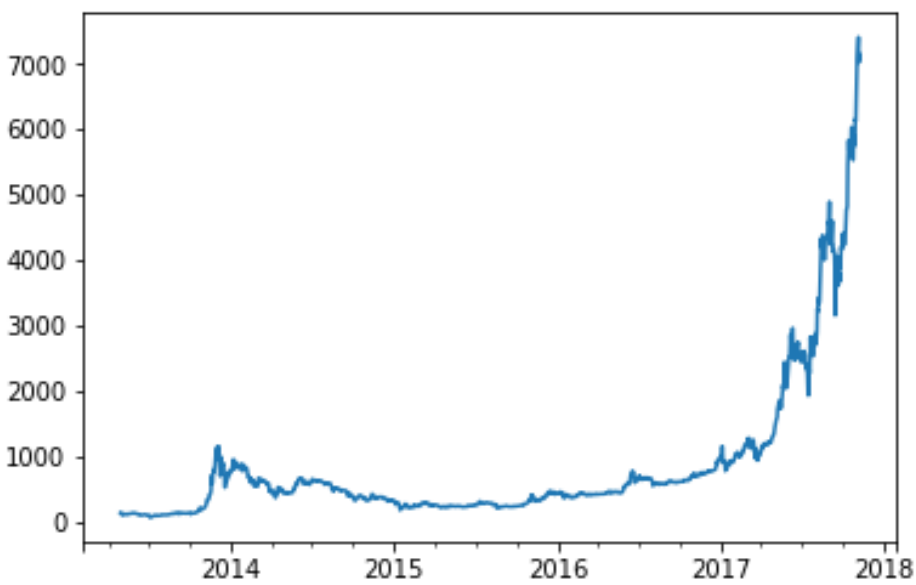
Xuất phát từ ý tưởng của G. Peter Zhang [10], nhiều nghiên cứu đã áp dụng mô hình kết hợp trong dự báo chuỗi thời gian. Durdu O¨mer Faruk [11] đề xuất hướng tiếp cận kết hợp mô hình ARIMA và mạng nơron truyền thẳng để dự báo chất lượng nước tại sông Büyük Menderes thuộc miền tây nam Thổ Nhĩ Kỳ trên tập dữ liệu được thu thập từ năm 1996-2004. Kết quả thực nghiệm cho thấy mô hình kết hợp cho kết quả dự báo tốt hơn từng mô hình riêng lẻ.

Ngoài ra, khi nghiên cứu dự báo xu hướng của giá chứng khoán, Nitin Merh [12] đã áp dụng mô hình kết hợp ARIMA và mạng nơron theo hai hướng tiếp cận. Hướng tiếp cận thứ nhất là xây dựng mô hình kết hợp ARIMA_ANN. Mô hình kết hợp này giống với ý tưởng của G. Peter Zhang [10] đề xuất, tức là áp dụng mô hình ARIMA để dự báo thành phần tuyến tính, sau đó sử dụng mô hình mạng nơron dự báo lỗi của mô hình (thành phần phi tuyến). Hướng tiếp cận thứ hai là xây dựng mô hình kết hợp ANN_ARIMA. Nitin Merh [12] thực hiện kết hợp ngược lại, xây dựng mô hình mạng nơron cho dự báo rồi sau đó áp dụng mô hình ARIMA để dự báo lỗi của mô hình mạng nơron. Kết quả thực nghiệm trên các tập dữ liệu chứng khoán như SENSEX, BSE IT, BSE Oil & Gas, BSE 100 and S& P CNX Nifty cho kết quả khá thú vị. Hầu hết các mô hình dự báo ANN_ARIMA đều có độ lỗi dự báo thấp hơn mô hình ARIMA_ANN.

IV. THỰC NGHIỆM VÀ KẾT QUẢ

A. Tập dữ liệu thực nghiệm

Trong bài báo này, các mô hình dự báo chuỗi thời gian sẽ được cài đặt và thực nghiệm trên tập dữ liệu giá đóng cửa của đồng Bitcoin được thu thập từ ngày 28/04/2013 đến ngày 7/11/2017 gồm 1655 ngày trên website của cộng đồng Kaggle. Dựa vào biểu đồ giá đóng cửa của đồng Bitcoin trong hình 4, có thể thấy giá của đồng Bitcoin có những biến động lớn, đặc biệt vào năm 2017. Tập dữ liệu thực nghiệm sẽ được chia thành hai phần: 80% các điểm thời gian được sử dụng để huấn luyện, 20% các điểm thời gian còn lại sẽ được sử dụng cho thử nghiệm mô hình.



Hình 4. Dữ liệu giá đóng cửa của Bitcoin

B. Đánh giá các mô hình dự báo

Để đánh giá chất lượng dự báo của các mô hình, bài báo này sử dụng các độ đo lỗi RMSE (Root Mean Square Error) và MAPE (Mean Absolute Percentage Error):

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{t=1}^{n}(y_t - \hat{y}_t)^2}$$

$$\text{MAPE} = \frac{1}{n}\sum_{t=1}^{n}\left|\frac{y_t - \hat{y}_t}{y_t}\right| * 100$$

với n là số điểm thời gian thử nghiệm, y_t là giá trị thực tế, $\hat{y}_t$ là giá trị dự báo từ mô hình.

C. Mô hình ARIMA cho dự báo giá Bitcoin

Để tăng độ chính xác dự báo của mô hình ARIMA, chuỗi dữ liệu thời gian được chuẩn hóa bằng cách lấy logarit tự nhiên. Dựa vào kiểm định Augmented Dickey Fuller (ADF), chuỗi dữ liệu giá đóng cửa Bitcoin không phải là một chuỗi thời gian dừng. Vì vậy, chuỗi dữ liệu cần được biến đổi thành chuỗi thời gian dừng bằng cách lấy sai phân bậc 1 (d=1). Sai phân bậc 1 được tính bởi công thức:

$$w_t = y_t - y_{t-1}$$

với w_t là chuỗi thời gian sau khi lấy sai phân bậc 1, y_t và y_{t-1} là giá trị chuỗi thời gian tại thời điểm t và t-1.

Sau khi kiểm định chuỗi thời gian w_t thì chuỗi dữ liệu giá đóng cửa của Bitcoin đã dừng. Sử dụng hàm ACF, PACF để xác định các giá trị q, p thích hợp cho mô hình. Kết quả từ chuỗi dữ liệu cho thấy q có thể nhận các giá trị 0, 4, 6, 11 và p có thể nhận các giá trị 0, 4, 6, 11. Chúng ta cần tìm ra một mô hình ARIMA phù hợp để dự báo giá Bitcoin. Các tiêu chuẩn AIC, BIC, HQIC được dùng để lựa chọn mô hình phù hợp với tập dữ liệu. Mô hình ARIMA nào có các giá trị này nhỏ nhất sẽ được lựa chọn.

Bảng 1. Các tiêu chuẩn kiểm định của mộ số mô hình ARIMA

Mô hình	AIC	BIC	HQIC
ARIMA (6,1,0)	-4604.38	**-4562.88**	**-4588.82**
ARIMA (0,1,6)	-4601.13	-4559.63	-4585.58
ARIMA (11,1,0)	**-4608.42**	-4540.98	-4583.14

Sau khi kiểm định các mô hình, mô hình ARIMA (6,1,0) được chọn làm mô hình dự báo giá đóng cửa của Bitcoin. Mô hình ARIMA (6,1,0) có dạng:

$$y_t = c + \emptyset_1 y_{t-1} + \emptyset_2 y_{t-2} + \emptyset_3 y_{t-3} + \emptyset_4 y_{t-4} + \emptyset_5 y_{t-5} + \emptyset_6 y_{t-6} + \varepsilon_t$$

Các tham số c, $\emptyset_i$, ε_t được ước lượng dựa vào tập dữ liệu huấn luyện. Sau khi mô hình được xây dựng, các giá trị y_t trong tương lai sẽ được dự báo. Giá trị dự báo và giá trị thực tế sẽ có sự sai số, đó là độ lỗi (phần dư) e_t:

$$e_t = y_t - \widehat{y_t}$$

Trong trường hợp, độ lỗi e_t là một số hạng sai số ngẫu nhiên thì mô hình phù hợp với chuỗi dữ liệu và được dùng trong dự báo các điểm thời gian trong tương lai. Nếu ngược lại, một mô hình ARIMA khác sẽ được thử nghiệm đến khi nào tìm được mô hình phù hợp với tập dữ liệu.

D. Mô hình FFNN cho dự báo giá Bitcoin

Chuỗi dữ liệu giá Bitcoin được chuẩn hóa min-max để tăng độ chính xác dự báo và hạn chế trường hợp quá khớp dữ liệu (overfitting). Việc chuẩn hóa min-max được áp dụng bởi công thức:

$$y' = \frac{y - y_{min}}{y_{max} - y_{min}}$$

với y' là giá trị sau khi chuẩn hóa, y là giá trị cần chuẩn hóa.

Để xây dựng mô hình FFNN cho dự báo giá Bitcoin, việc đầu tiên là cần xác định các biến đầu vào và các biến đầu ra cho mô hình. Biến đầu vào là các giá trị dữ liệu giá đóng cửa của Bitcoin ở những ngày trước thời điểm dự báo. Số lượng biến đầu vào được xác định dựa trên thực nghiệm để tìm ra giá trị phù hợp. Biến đầu ra là một giá trị dữ liệu giá Bitcoin được dự báo từ mô hình. Việc kế tiếp là xác định số lớp ẩn và số nơron trong từng lớp ẩn. Thông qua quá trình thực nghiệm để xác định được các thông số này. Với bài toán dự báo giá Bitcoin, chúng tôi xây dựng mô hình FFNN gồm có 3 lớp: 1 lớp đầu vào, 1 lớp ẩn, 1 lớp đầu ra. Số biến của lớp đầu vào (n) được thử nghiệm từ 1 đến 30 (các điểm thời gian trong một tháng trước dự báo). Số nơron lớp ẩn được thử nghiệm từ 1 đến 2n cho mỗi mô hình FFNN. Chúng tôi chọn số lần học trong quá trình huấn luyện lần lượt là 100, 200, 500, 1000. Các mô hình FFNN gồm 2 lớp ẩn cũng đã được thực nghiệm nhưng kết quả không khả quan bởi nhiều lớp ẩn sẽ dẫn đến tính toán phức tạp và xảy ra trường hợp quá khớp dữ liệu. Sau những kết quả thử nghiệm, mô hình FFNN gồm 14 nơron trong lớp input (dựa vào giá Bitcoin của 14 ngày trước để dự báo giá của ngày tiếp theo), 28 nơron trong lớp ẩn và 1 nơron trong lớp đầu ra được lựa chọn cho dự báo giá Bitcoin.

Bảng 2. Độ lỗi dự báo của một số mô hình FFNN

Số nơron lớp đầu vào	Số nơron lớp ẩn thứ nhất	Số nơron lớp ẩn thứ hai	Số nơron lớp đầu ra	RMSE	MAPE (%)
3	6	-	1	291.72	6.69
6	12	-	1	541.09	10.15
7	14	-	1	221.27	6.37
14	28	-	1	**147.20**	**3.41**
21	42	-	1	213.94	7.08
30	60	-	1	161.97	3.68
3	6	4	1	785.24	23.01
14	24	14	1	286.85	8.19
21	42	30	1	488.41	14.29

E. Mô hình CNN cho dự báo giá Bitcoin

Cũng giống như mô hình FFNN, việc xác định số lượng biến đầu vào cũng phải được xác định qua thực nghiệm. Ngoài ra, mô hình CNN còn cần xác định số lớp tích chập (convolutional), số lớp lấy mẫu (pooling), số lớp kết nối đầy đủ (fully-connected). Trong dự báo chuỗi thời gian, những lớp này thường được chọn với số lượng ít để hạn chế trường hợp quá khớp dữ liệu. Trong thực nghiệm này, chúng tôi xây dựng mô hình CNN cho dự báo giá Bitcoin với 1 lớp tích chập gồm 64 bộ lọc, hàm relu được sử dụng để phi tuyến tính giá trị đầu ra từ lớp tích chập, 1 lớp lấy mẫu với ma trận lấy mẫu là 2x1, thêm vào đó là 1 lớp kết nối đầy đủ. Sau đó, lớp đầu ra sẽ có 1 nơron để tính giá trị dự báo cho mô hình.

Bảng 3. Độ lỗi dự báo của một số mô hình CNN

Số nơron lớp đầu vào	Số lớp convolution	Số lớp pooling	Số nơron lớp fully connnected	Số nơron lớp đầu ra	RMSE	MAPE (%)
3	1	1	30	1	276.99	4.22
3	1	1	50	1	**252.89**	**4.09**
7	1	1	70	1	262.47	4.20
7	1	1	100	1	378.71	6.48

F. Mô hình kết hợp ARIMA và mạng nơron cho dự báo giá Bitcoin

Mô hình kết hợp ARIMA và mạng nơron sẽ được thực nghiệm giống với mô hình do G. Peter Zhang [10] đề xuất. Với thực nghiệm kết hợp ARIMA-FFNN và ARIMA-CNN, tập dữ liệu huấn luyện được chia thành 10 khoảng dữ liệu thời gian. Trong đó, những dữ liệu thuộc 9 khoảng thời gian đầu được dùng làm dữ liệu huấn luyện cho mô hình ARIMA. Sau đó, dữ liệu thuộc khoảng thời gian còn lại sẽ được huấn luyện với mô hình FFNN và CNN để dự báo lỗi cho mô hình ARIMA vừa xây dựng. Sử dụng mô hình ARIMA đã huấn luyện để dự báo giá trị cho các điểm thời gian trong tập dữ liệu thử nghiệm. Song song đó, các mô hình mạng nơron cũng được sử dụng để dự báo lỗi cho mô hình ARIMA trong tập dữ liệu thử nghiệm. Kết quả dự báo sẽ là giá trị tổng hợp của cả hai mô hình.

Trong thực nghiệm với dữ liệu giá Bitcoin, mô hình ARIMA (6,1,0) tiếp tục được sử dụng cho dự báo. Các mô hình FFNN và CNN cũng được chạy thử nghiệm nhiều lần với các tham số gồm số biến đầu vào, số lớp ẩn, số nơron trong lớp ẩn. Kết quả thực nghiệm cho thấy mô hình ARIMA (6,1,0) kết hợp với mô hình FFNN-3-8-1 gồm 3 biến đầu vào, 8 nơron của 1 lớp ẩn và 1 nơron đầu ra cho độ lỗi dự báo thấp nhất dựa trên độ đo RMSE và MAPE. Mô hình kết hợp ARIMA-CNN tốt nhất cho dự báo Bitcoin gồm mô hình ARIMA (6,1,0) và mô hình CNN-4-1-1-6-1 với 4 nơron lớp đầu vào, 1 lớp tích chập gồm 64 bộ lọc, 1 lớp lấy mẫu với ma trận lấy mẫu là 2x1, thêm vào đó là 1 lớp kết nối đầy đủ với 6 nơron và 1 nơron lớp đầu ra.

Bảng 4. Độ lỗi dự báo của một số mô hình kết hợp ARIMA và FFNN

Mô hình kết hợp		RMSE	MAPE (%)
ARIMA (6, 1, 0)	FFNN-1-2-1	128.54	2.94
	FFNN-2-4-1	128.37	3.01
	FFNN-2-3-1	130.26	2.98
	FFNN-3-8-1	**126.66**	**2.94**
	FFNN-3-7-1	131.24	2.97

Bảng 5. Độ lỗi dự báo của một số mô hình kết hợp ARIMA và CNN

Mô hình kết hợp		RMSE	MAPE (%)
ARIMA (6, 1, 0)	CNN-3-1-1-10-1	130.96	3.06
	CNN-3-1-1-8-1	130.66	3.08
	CNN-4-1-1-6-1	**128.30**	**2.94**
	CNN-5-1-1-7-1	135.85	3.19

G. Kết quả thực nghiệm các mô hình dự báo giá Bitcoin

Kết quả thực nghiệm của các mô hình dự báo giá đóng cửa (USD) của Bitcoin được trình bày trong bảng 6.

Bảng 6. Bảng so sánh độ lỗi dự báo của các mô hình

Mô hình	RMSE	MAPE (%)
ARIMA	131.49	3.03
FFNN	147.20	3.41
CNN	252.89	4.09
ARIMA-FFNN	**126.66**	**2.94**
ARIMA-CNN	128.30	2.94

Kết quả thực nghiệm cho thấy mô hình kết hợp ARIMA-FFNN cho kết quả dự báo tốt hơn các mô hình khác dựa vào độ đo RMSE và MAPE. Ngoài ra, mô hình ARIMA cũng cho thấy kết quả dự báo tốt trên tập dữ liệu Bitcoin. Trong khi đó, các mô hình mạng nơron FFNN và CNN tỏ ra không thật sự tốt để dự báo giá Bitcoin.

V. KẾT LUẬN VÀ HƯỚNG NGHIÊN CỨU TRONG TƯƠNG LAI

Dự báo chuỗi thời gian là một bài toán khó nhưng rất quan trọng trong các lĩnh vực kinh tế và tài chính. Gần đây, dự báo Bitcoin nhận được nhiều quan tâm của các nhà nghiên cứu trong kinh tế và khoa học máy tính. Với các đặc trưng của mình, giá Bitcoin biến động thường xuyên dẫn đến việc dự báo gặp nhiều khó khăn. Bài báo này thực nghiệm các mô hình dự báo ARIMA, FFNN, CNN, ARIMA-FFNN, ARIMA-CNN để dự báo giá đóng cửa của Bitcoin trong ngày tiếp theo. Kết quả thực nghiệm cho thấy, các mô hình kết hợp ARIMA và mạng nơron đều có độ lỗi dự báo thấp hơn các mô hình riêng lẻ, chứng tỏ mô hình kết hợp sẽ cho dự báo tốt hơn. Trong đó, mô hình kết hợp ARIMA-FFNN cho độ lỗi dự báo thấp nhất với các độ đo RMSE và MAPE.

Nhìn chung, với chuỗi dữ liệu biến động lớn như giá Bitcoin, các mô hình ARIMA và mạng nơron còn gặp nhiều khó khăn trong dự báo trên tập dữ liệu này. Trong tương lai, các mô hình Deep Learning sẽ được nghiên cứu thử nghiệm trên dữ liệu giá Bitcoin với kỳ vọng mang lại kết quả dự báo tốt hơn.

VI. TÀI LIỆU THAM KHẢO

[1] Satoshi Nakamoto, “Bitcoin: A Peer-to-Peer Electronic Cash System”, 2008.
[2] Devavrat Shah, Kang Zhang, “Bayesian regression and Bitcoin”, arXiv preprint arXiv:1410.1231, 2014.
[3] Siddhi Velankar, Sakshi Valecha, Shreya Maji, “Bitcoin price prediction using machine learning”, In Proceedings of the 2018 20th International Conference on Advanced Communication Technology (ICACT), Chuncheon-si, Korea, pp. 144–147, 2018.
[4] João Almeida, Shravan Tata, Andreas Moser, Vikko Smit, “Bitcoin prediction using ANN”, 2015.
[5] Huisu Jang, Jaewook Lee, “An empirical study on modeling and prediction of Bitcoin prices with bayesian neural networks based on blockchain information”, IEEE Access, vol. 6, pp. 5427–5437, 2018.
[6] Bogdan Oancea, ŞTefan Cristian Ciucu, “Time series forecasting using neural networks”, In Proceedings of the “Challenges of the Knowledge Society” Conference, eprint arXiv:1401.1333, pp. 1402–1408, 2014.
[7] M. Raeesi, M. S. Mesgari, P. Mahmoudi, “Traffic Time Series Forecasting by Feedforward Neural Network: a Case Study Based on Traffic Data of Monroe”, The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences, 40(2), 219, 2014.
[8] Kumar Abhishek, Anshul Khairwa, Tej Pratap, Surya Prkash, “A stock market prediction model using Artificial Neural Network”, Computing Communication & Networking Technologies (ICCCNT), 2012.
[9] Sheng Chen, Hongxiang He, “Stock Prediction Using Convolutional Neural Network”, IOP Conference Series Materials Science and Engineering, 2018.
[10] G. Peter Zhang, “Times series forecasting using a hybrid ARIMA and neural network model”, Neurocomputing, vol. 50, pp. 159–75, 2003.
[11] Durdu O¨ mer Faruk, “A hybrid neural network and ARIMA model for water quality time series prediction”, Engineering Applications of Artificial Intelligence, 23(4), 586-594, 2010.
[12] Nitin Merh, Vinod P. Saxena, Kamal Raj Pardasani, “A comparison between hybrid approaches of ANN and ARIMA for Indian stock trend forecasting”, Journal of Business Intelligence, vol. 3, no. 2, pp. 23–43, 2010.

FORECASTING BITCOIN PRICE BY COMBINING ARIMA MODEL AND ARTIFICIAL NEURAL NETWORK

Le Huu Vinh, Nguyen Dinh Thuan

ABSTRACT*— In the recent years, Bitcoin has emerged as the most widely used cryptocurrency in the world. Bitcoin can be used for online payment or simply a profitable investment channel. Bitcoin is not dependent on events in business or by intervening governments. Bitcoin prices change frequently so the forecasting problem is challenging. In this paper, we experiment several models such as ARIMA, neural network, hybrid ARIMA and neural network to predict the closing price (USD) of the Bitcoin in the next day. The closing price data of Bitcoin is collected from April 28, 2013 to November 7, 2017, including 1655 days on the Kaggle community website. The forecasted results of the models will be compared to consider which model is more suitable for forecasting Bitcoin prices.*