A Novel Hybrid Approach Using an Attention-Based Transformer + GRU Model for Predicting Cryptocurrency Prices

Esam Mahdi *, Carlos Martin Barreiro , Xavier Cabezas

Posted Date: 8 April 2025

doi: 10.20944/preprints202504.0484.v1

Keywords: Bitcoin; Cryptocurrencies; Deep learning; Ethereum, Fear, and greed index; Gated recurrent unit; General regression neural network; Long short-term memory; Radial basis function network; Transformer modl

 doi:10.20944/preprints202504.0484.v1

Article

A Novel Hybrid Approach Using an Attention-Based Transformer + GRU Model for Predicting Cryptocurrency Prices

Esam Mahdi [1,*], Carlos Martin-Barreiro [2], Xavier Cabezas [3,4]

[1] School of Mathematics and Statistics, Carleton University, Canada; EsamMahdi@cunet.carleton.ca (E.M.)
[2] Facultad de Ingeniería, Universidad Espíritu Santo, Samborondón 0901952, Ecuador; carlosmartin@uees.edu.ec (C.M.-B.)
[3] Facultad de Ciencias Naturales y Matemáticas, Escuela Superior Politécnica del Litoral ESPOL, Guayaquil 090902, Ecuador; joxacabe@espol.edu.ec (X.C.)
[4] Centro de Estudios e Investigaciones Estadísticas, Escuela Superior Politécnica del Litoral ESPOL, Guayaquil, Ecuador; joxacabe@espol.edu.ec (X.C.)
* Correspondence: EsamMahdi@cunet.carleton.ca (E.M.)

Abstract: In this article, we introduced a novel deep learning hybrid model that combines attention transformer and gated recurrent unit architectures to improve the accuracy of cryptocurrency price predictions. The transformer architecture is used to capture long-range dependencies in the data, while the GRU is used to model sequential patterns and short-term fluctuations. We considered the daily closing prices, trading volume, and Fear and Greed Index of Bitcoin from September 17, 2014 to February 28, 2025, and Ethereum from November 9, 2017 to February 28, 2025. We evaluated the performance of our proposed model by comparing it with four other machine learning models: two are non-sequential feedforward models: Radial Basis Function Network (RBFN) and General Regression Neural Network (GRNN), and two are bidirectional sequential memory-based models: Bidirectional Long Short-Term Memory (BiLSTM) and Bidirectional Gated Recurrent Unit (BiGRU). The performance evaluation is based on some performance metrics, including Mean Squared Error (MSE), Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), and Mean Absolute Percentage Error (MAPE). The results demonstrated that the hybrid Transformer + GRU model outperforms the other models, achieving the lowest values across all performance metrics.

Keywords: Bitcoin, Cryptocurrencies, Deep learning, Ethereum, Fear & greed index, Gated recurrent unit, General regression neural network, Long short-term memory, Radial basis function network, Transformer model.

1. Introduction

The cryptocurrency market has grown exponentially, with Bitcoin and Ethereum standing out as the top two dominant digital assets. Both of these account for the majority of trading activity and significantly impact global financial trends. One of the biggest challenges in working with cryptocurrencies is figuring out how to develop accurate models that can predict their highly volatile prices. Numerous studies have showed that machine learning (ML) models are highly effective in predicting cryptocurrency prices, as they can capture complex nonlinear patterns in financial time series [1–9]. Among these, artificial neural network (ANN)-based machine learning algorithms, such as radial basis function network (RBFN), general regression neural network (GRNN), gated recurrent unit (GRU), and long short-term memory (LSTM), have been widely applied in this field.

Broomhead and Lowe [10,11] introduced the radial basis function networks (RBFNs) to model complex relationships between features and target variables and to make predictions on new data. RBFNs are a type of simple feedforward neural network that utilizes radial basis functions as their activation functions. One of the key benefits of using radial basis functions in RBFNs is their ability to smooth out non-stationary time series data while effectively still modeling the underlying trends.

 doi:10.20944/preprints202504.0484.v1

Alahmari [12] compared using the linear, polynomial, and radial basis function (RFB) kernels in the support vector regression for predicting the prices of Bitcoin, XRP and Ethereum and showed that the RFB outperforms the other kernel methods in terms of accuracy and effectiveness. Recently, Casillo et al. [13] showed that the RRBFn models are effective in predicting Bitcoin prices from the analysis of online discussion sentiments. More recently, Zhang [14] demonstrated that combining a radial basis function (RBF) with a battle royale optimizer (BRO) significantly enhances the accuracy of stock price predictions, including those for cryptocurrencies. The results of his study showed that this approach outperformed other machine learning models like LSTM, BiLSTM-XGBoost, and CatBoost.

The general regression neural network (GRNN) is another type of feedforward neural network was first proposed by Specht [15] designed for efficient use in regression tasks. One of the key strengths of GRNN is its ability to model complex nonlinear relationships without the need for iterative training. This has made it a powerful tool for applications in regression, prediction, and classification. Despite its advantages, GRNN's application in cryptocurrency prediction is still relatively not widely explored [16,17]. In this article, we try to fill in the gap by assessing the performance of GRNN in predicting the prices of Bitcoin and Ethereum.

Recurrent neural networks (RNNs) are designed for handling sequential data because, unlike feedforward neural networks, they have feedback loops that enable them to retain information from previous inputs using a method called backpropagation through time (BPTT). However, when training on long sequences, the residual error gradients that needs to be propagated back diminishes exponentially due to repeated multiplication of small weight across time steps. This makes it difficult for the network to learn long term dependencies. To address such a problem and to improve the modeling of long term dependencies in sequential data, Hochreiter and Schmidhuber [18] proposed the long short-term memory (LSTM) network model. LSTM model has been widely adopted in literature, as it has proven to be highly effective in forecasting cryptocurrency prices, outperforming other machine learning models [19–21]. Lahmiri and Bekiros [22] showed that the Bitcoin, Digital Cash, and Ripple price predictability of LSTM is significantly higher when compared to that of GRNN. Ji et al. [23] conducted a comparative study comparing the LSTM network model with deep learning network, convolutional neural network, and deep residual network models for predicting the Bitcoin prices and showed that the LSTM slightly outperformed the other models for regression problems, whereas for classification (up and down) problems, deep learning network works the best. Uras et al. [24] used LSTM to predict the daily closing price series of Bitcoin, Litecoin and Ethereum cryptocurrencies, based on the prices and volumes of prior days and showed that the LSTM outperformed the traditional linear regression models. Lahmiri and Bekiros [25] conducted a study comparing the performance of LSTM and GRNN models in predicting Bitcoin, Digital Cash, and Ripple prices. Their findings indicated that the LSTM model outperformed GRNN in terms of prediction accuracy.

The gated recurrent unit (GRU) neural network model, introduced by Cho et al. [26], was designed to handle long term dependencies in sequential data, similar to the LSTM model, but with fewer parameters, making GRU computationally less expensive and faster than LSTM [27]. Dutta et al. [28] used GRU to predict the prices of cryptocurrencies and demonstrated that the GRU outperformed the LSTM networks. Tanwar et al. [29] employed GRU and LSTM to predict the price of Litecoin and Zcash cryptocurrencies, taking into account the inter-dependency of the parent coin. Their findings showed that these models forecasted the prices with high accuracy compared to other machine learning models. Ye et al. [30] proposed a method combining LSTM and GRU to predict the prices of Bitcoin using the historical transaction data, sentiment trends of Twitter, and technical indicators, and showed that their method can better assist investors in making the right investment decision. Patra1 and Mohanty [31] developed a multi-layer GRU network model with multiple features to predict the prices of Bitcoin, Ethereum, and Dogecoin, and demonstrated that their model provided a better performance compared with the LSTM and GRU models that used a single feature.

 doi:10.20944/preprints202504.0484.v1

Several studies have shown that bidirectional long short-term memory (BiLSTM) and bidirectional gated recurrent unit (BiGRU) enhance the accuracy of financial time series predictions. Unlike traditional LSTM and GRU, these bidirectional models process data in both forward and backward directions allowing to capture dependencies from past and future time steps. For example, Hansun et al. [32] compared three popular deep learning architectures, LSTM, bidirectional LSTM, and GRU, for predicting the prices of five cryptocurrencies, Bitcoin, Ethereum, Cardano, Tether, and Binance Coin using various prediction models. Their findings indicated that BiLSTM and GRU performed just as well as LSTM, offering robust and accurate predictions of cryptocurrency prices. Ferdiansyah et al. [33] showed that combining GRU and BiLSTM in a hybrid model increased prediction accuracy for Bitcoin, Ethereum, Ripple, and Binance.

In this article, we propose a novel hybrid transformer + GRU model and compare its performance with two non-sequential feedforward models (GRNN and RBFN) and two bidirectional sequential memory-based models (BiGRU and BiLSTM) for forecasting cryptocurrency prices across two distinct modeling scenarios. The first scenario aims to predict Bitcoin prices using historical Bitcoin price data, its trading volume, and crypto fear and greed index (FGI). The second scenario aims to predict Ethereum prices using historical price data for both Bitcoin and Ethereum, along with Ethereum's trading volume and the Fear and Greed Index (FGI). The Transformer model, introduced by Vaswani et al. [34], revolutionized natural language processing (NLP) by using self-attention and feed-forward networks. Since then, transformers have been adapted and extended in numerous ways and became the foundation for AI systems like ChatGPT, DeepSeek, and others [35,36]. The application of transformer neural networks to financial time series data is a promising area of research [37–39]. To the best of our knowledge, this is the first study to introduce a deep learning model that combines a self-attention-based Transformer architecture with a sequential memory-based GRU model.

The rest of this article is organized as follows: Section 2 defines the prediction models used to predict Bitcoin and Ethereum prices. Here, we define four neural network models that we need to compare with our proposed model. Two of these four models are classified as feedforward neural networks and the other two are classified as memory-based models. Section 3 introduces our new deep learning approach for cryptocurrency price prediction. Section 4 presents the data and results, showcasing how all five models perform in this comparative study. Finally, Section 5 summarizes our findings and draws some conclusions and recommendations.

2. Prediction Models

We consider modeling the cryptocurrency prices as a function of past feature values with a one-time-step lag:

$$y_t = f(\mathbf{x}_{t-1}) + \varepsilon_t, \tag{1}$$

where y_t is the target variable represents the price of Bitcoin or Ethereum at time t, $f(\mathbf{x}_{t-1})$ is a function of the features $\mathbf{x}_{t-1}$ at time $t-1$ that need to be approximated, and ε_t is the error term represents the difference between the predicted price and actual price at time t.

In this article, we consider the following two scenarios:

Model 1: $\mathbf{x}_{t-1}$ is a three dimensional vector represents the prices of Bitcoin in US dollars, fear and greed index (FGI), and exchange trading volume (USD) of Bitcoin at time $t-1$. Thus, model (1) can be rewritten as follows:

$$\mathrm{B}_t = f(\mathrm{B}_{t-1}, F_{t-1}, V_{t-1}) + \varepsilon_t, \tag{2}$$

where B_{t-1}, F_{t-1}, and V_{t-1} are the prices (USD) of Bitcoin, fear and greed index, and exchange trading volume of Bitcoin (USD) at time $t-1$, respectively.

 doi:10.20944/preprints202504.0484.v1

Model 2: $\mathbf{x}_{t-1}$ is a four dimensional vector represents the prices of Ethereum, Bitcoin in US dollars, fear and greed index (FGI), and exchange trading volume of Ethereum (USD) at time $t-1$.

$$\mathrm{E}_t = f(\mathrm{E}_{t-1}, \mathrm{B}_{t-1}, F_{t-1}, V_{t-1}) + \varepsilon_t, \tag{3}$$

where E_{t-1} is price of Ethereum at time $t-1$.

We use the following four different network along with our proposed model models to forecast the prices Bitcoin and Ethereum based on Equations 2 and 3.

2.1. Radial Basis Function Network (RBFN)

Radial basis function network (RBFN) is a type of feedforward neural network that uses radial basis functions as activation functions. Typically, it consists of three layers as seen in Figure 1: the Input Layer, which receives the input data; the Hidden Layer, which applies radial basis functions, such as Gaussian activation (kernel) functions, to transform the input data into a higher-dimensional space; and the Output Layer, which produces the final output, often as a linear combination of the hidden layer outputs.

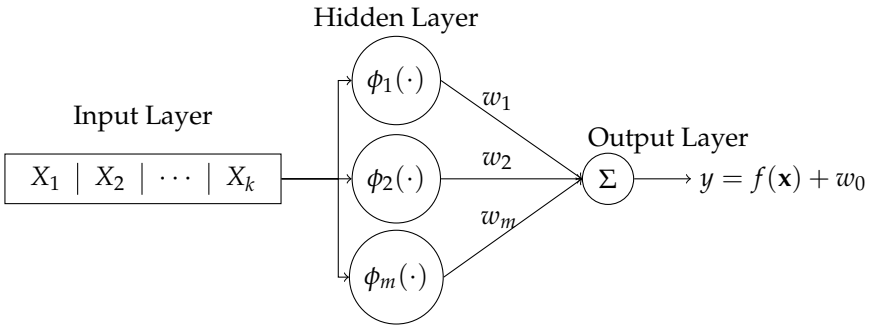

Figure 1. Architecture of a radial basis function network (RBFN).

Mathematically, the input can be considered as a vector of k variables $\mathbf{x} = (X_1, X_2, \cdots, X_k)^\top$, each with n observations. The kernel function is used as the activation function in the hidden layer. Typically, the most common choice is the Gaussian function:

$$\varphi(\mathbf{x}, \mathbf{c}_i) = \exp\left(-\frac{\|\mathbf{x} - \mathbf{c}_i\|^2}{2\sigma_i^2}\right) \tag{4}$$

where $||\mathbf{x} - \mathbf{c}_i||$ is the Euclidean distance between the input vector $\mathbf{x}$ and the center $\mathbf{c}_i = (c_{i1}, c_{i2}, \cdots, c_{ik})$, and σ_i is the bandwidth or spread parameter. The output of the hidden layer is passed to the output layer, where the final output is computed as a linear combination of the hidden layer outputs:

$$y = \sum_{i=1}^{m} w_i \varphi(\mathbf{x}, \mathbf{c}_i) + w_0 \tag{5}$$

where m denotes the number of neurons in the hidden layer, $\mathbf{c}_i$ is the center vector for neuron i, w_i is the weight of neuron i in the linear output neuron, w_0 is the bias term, and φ is the activation function (radial basis function) for the i^{th} RBF neuron.

2.2. General Regression Neural Network (GRNN) Model

The general regression neural network (GRNN), introduced by Specht [15], is another type of feedforward neural network that is designed for regression tasks and closely related to the radial basis function (RBF) network. Unlike traditional feedforward and deep neural networks, GRNN does not rely on backpropagation or gradient-based optimization. Instead, it operates by minimizing the

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 8 April 2025 doi:10.20944/preprints202504.0484.v1

difference between predicted and actual target values. The model directly memorizes the training data and leverages it for predictions.

GRNN comprises three layers: the input layer, hidden layer, and output layer. The set of n training data $\{\mathbf{x}_i, y_i\}_{i=1}^{n}$ enters the input layer as a set of k features $\mathbf{x}$ and their associated target y. Then the hidden layer uses a kernel function to compute the similarity between a new input feature $\mathbf{x}$ and each training sample $\mathbf{x}_i$. The Gaussian kernel is commonly used:

$$K(\mathbf{x}, \mathbf{x}_i) = \exp\left(-\frac{\|\mathbf{x} - \mathbf{x}_i\|^2}{2\sigma_i^2}\right), \tag{6}$$

where $\mathbf{x}, \mathbf{x}_i$, and σ_i are the new input vector, the training sample, and the smoothing parameter respectively. Then it gives a set of weights associated with the closeness distance:

$$w_i = \frac{\exp\left(-\frac{\|\mathbf{x}-\mathbf{x}_i\|^2}{2\sigma_i^2}\right)}{\sum_{i=1}^{n} \exp\left(-\frac{\|\mathbf{x}-\mathbf{x}_i\|^2}{2\sigma_i^2}\right)}. \tag{7}$$

Finally, the output layer computes the prediction value of y as a weighted average of nearby observations:

$$\hat{y} = \sum_{i=1}^{n} w_i y_i \tag{8}$$

2.3. Long Short-Term Memory (LSTM) Model

Long short-term memory (LSTM) is a specialized type of recurrent neural network (RNN) developed to solve the vanishing gradient problem that standard RNNs often faced. This problem makes it challenging for the model to capture long term dependencies when dealing with long sequences. To overcome this limitation, LSTM uses a memory cell designed to store information over extended periods while discarding irrelevant details. Figure 2 illustrates the structure of the LSTM.

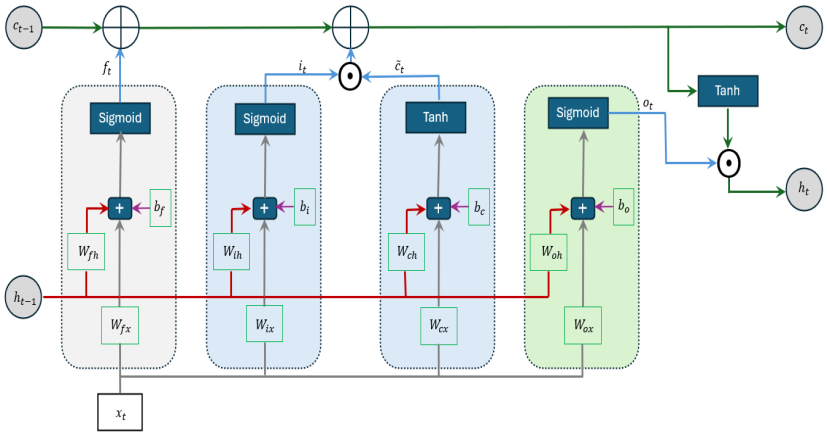

Figure 2. Architecture of long short-term memory (LSTM) network contains four interacting layers.

The first step of the LSTM network is to decide what information will be discarded from the memory cell state c_{t-1}. The sigmoid activation function in the forget gate is applied to the current input, X_t, and the output from the previous hidden state, h_{t-1}, and produces values between 0 and 1 for each element in the cell state. A value of 1 indicates "keep this information", while a value of 0 means "omit this information":

$$\text{Forget gate:} \quad f_t = \sigma\left(W_{fx}X_t + W_{fh}h_{t-1} + b_f\right) \tag{9}$$

 doi:10.20944/preprints202504.0484.v1

where W_{fx} and W_{fh} represent the weight matrices for the input and recurrent connections, respectively, and b_f denotes the bias vector parameters.

The next step decides what new information will be stored in the cell state. The input gate first applies the sigmoid layer to the input from the previous hidden state and the input from the current state to determine which parts of the information should be updated. Then, a tanh layer generates a new candidate value, $\tilde{c}_t$, that could potentially be added to the cell state:

$$\text{Input gate:} \quad i_t = \sigma(W_{ix}x_t + W_{ih}h_{t-1} + b_i) \tag{10}$$

$$\text{Intermediate cell state:} \quad \widetilde{c_t} = \tanh(W_{cx}x_t + W_{ch}h_{t-1} + b_c) \tag{11}$$

where W and b are weight matrices and bias vector parameters. Following this, the previous cell state c_{t-1} is updated by a new cell state c_t:

$$\text{New cell state:} \quad c_t = f_t \odot c_{t-1} + i_t \odot \tilde{c}_t \tag{12}$$

where $\odot$ denotes the Hadamard product (element-wise product).

Lastly, the model determines the final output which is derived from the updated cell state after applying a filtering process: First, a sigmoid layer decides which portions of the cell state will be included in the output. Next, the cell state is passed through a tanh activation function to scale its values between -1 and 1, and the result is multiplied by the output gate to produce the final output.

$$\text{Output gate:} \quad o_t = \sigma(W_{ox}x_t + W_{oh}h_{t-1} + b_o) \tag{13}$$

$$\text{New state:} \quad h_t = o_t \odot \tanh(c_t) \tag{14}$$

2.4. Gated Recurrent Unit (GRU) Model

The Gated Recurrent Unit (GRU) has a simpler design than the LSTM network while still being effective at capturing long term dependencies and handling the vanishing gradient problem we face in using RNN Cho et al. [26]. It uses two gates: the reset gate and update gate. The reset gate determines how much of the previous hidden state h_{t-1} should be discarded when computing the new candidate hidden state $\tilde{h}_{t-1}$. The update gate controls how much of the new candidate hidden state, $\tilde{h}_{t-1}$, should be used to update the current hidden state.

Figure 3 illustrates the structure of the GRU, while Equations 15−18 explain its functionality mathematically as follows:

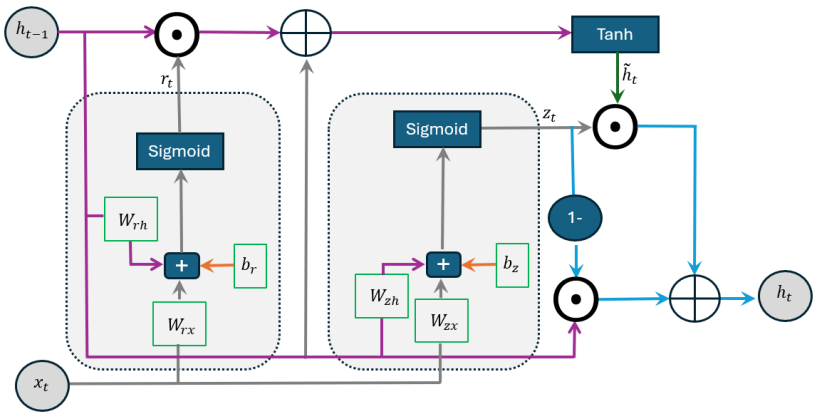

Figure 3. Architecture of gated recurrent unit (GRU) network.

 doi:10.20944/preprints202504.0484.v1

$$\text{Reset gate}: \quad r_t = \sigma(W_{rx}x_t + W_{rh}h_{t-1} + b_r) \tag{15}$$

$$\text{Update gate}: \quad z_t = \sigma(W_{zx}x_t + W_{zh}h_{t-1} + b_z) \tag{16}$$

$$\text{Cell state}: \quad \tilde{h}_t = \tanh(W_x x_t + W_h(r_t \odot h_{t-1}) + b) \tag{17}$$

$$\text{New state}: \quad h_t = z_t \odot \tilde{h}_t + (1 - z_t) \odot h_{t-1} \tag{18}$$

where, x_t denotes the input at the current state, h_{t-1} represents the hidden state at the previous state, r_t signifies the output of the reset gate, and z_t corresponds to the output of the update gate. The symbol $\odot$ stands for the Hadamard product, which is an element-wise multiplication operation and $\tilde{h}_t$ denotes the candidate hidden state. Additionally, W_x and W_h refer to the feedforward and recurrent weight matrices, respectively, while b represents the bias parameters.

As shown in Figure 3 and Equations 15−18, the reset gate applies the sigmoid activation function to a linear combination of the current state, the output from the previous state, and a bias term. This helps determine how much information should be discarded. Similarly, the update gate uses the sigmoid function on a different linear combination of the current state, the previous output, and a bias term to decide how much information should be updated. The tanh activation function is then used to generate a new candidate value, $\tilde{h}_t$. This candidate is multiplied (using the Hadamard product) by the output from the update gate. Meanwhile, the difference value of the update state from 1 is multiplied by the previous output (again using the Hadamard product), and this result is added to the first product to produce the final output.

Research has demonstrated that processing the input sequences in both forward and backward directions can improved the accuracy of time series predictions. In our analysis, we will consider the two bidirectional models: the bidirectional long short-term memory (BiLSTM) and the bidirectional gated recurrent unit (BiGRU). Figure 4 shows the architecture of BiGRU. The BiLSTM follows the same

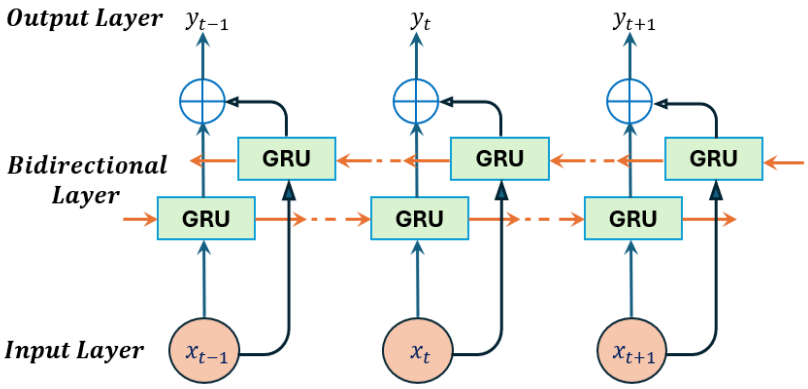

Figure 4. Architecture of bidirectional gated recurrent unit (BiGRU).

architecture as the BiGRU shown in Figure 4, replacing the word "GRU" by "LSTM". The forward pass in BiLSTM and BiGRU processes the sequence from $t = 1$ to $t = n$, while the backward pass processes the sequence from $t = n$ to $t = 1$. This bidirectional architecture allow these models to capture dependencies from past and future time steps, leading to improved prediction performance.

3. Hybrid Transformer + GRU Architecture

Transformer neural networks, introduced by Vaswani et al. [34] in their seminal 2017 paper "Attention is All You Need", have revolutionized the field of natural language processing (NLP). Unlike traditional sequential models like LSTM and GRU, transformers use self-attention to model relationships between all elements in a parallel sequence simultaneously, rather than processing them

 doi:10.20944/preprints202504.0484.v1

step by step. This architecture allows transformers to handle both local and global dependencies without the need for recurrent connections more efficiently. The transformers follow an encoder-decoder architecture, where both consist of multiple identical layers. The encoder processes the input, and the decoder produces the output. By leveraging the strengths of transformer and gated recurrent unit deep learning models, we propose a new hybrid Transformer + GRU model for cryptocurrency price prediction. Figure 5 illustrates the structure of this model. The main idea of our proposed model is to treat the historical cryptocurrency prices, trading volumes, and the Fear and Greed Index as a sequence of tokens, leveraging the self-attention mechanism to capture long-range dependencies while using GRU to detect sequential patterns and short-term fluctuations across different time steps. As

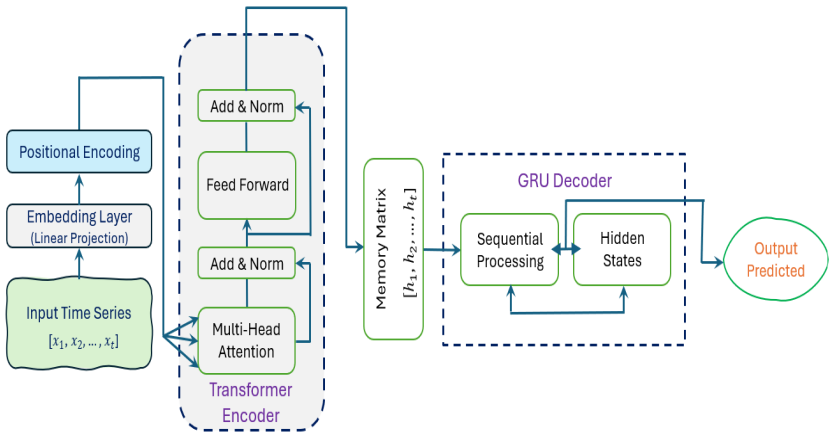

Figure 5. Architecture of hybrid Transformer + GRU model.

shown in Figure 5, the historical data is first encoded as input tokens, then it goes through embedding and positional encoding before entering the Transformer layers. The encoder outputs are fed into the GRU decoder, which applies attention over the encoder's memory to make predictions.

4. Exploratory Data Analysis

In our analysis, we consider the top two prominent cryptocurrencies with the highest cryptocurrency market capitalization: Bitcoin and Ethereum. The daily data was downloaded from the website https://coinmarketcap.com (last accessed on 4 April 2025), where we consider the daily closing prices and the trading exchange volume in USD for both digital assets. Table 1 provides a detailed summary of the respective cryptocurrency price datasets analyzed in this study. The table includes information on the start date, end date, and the total number of records for each dataset, offering a comprehensive overview of the temporal coverage and dataset size for Bitcoin and Ethereum. These details are crucial for understanding the scope and reliability of the data utilized in subsequent analyses.

Table 1. Historical data of Bitcoin and Ethereum prices.

Cryptocurrency	Start Date	End Date	Number of Records
Bitcoin	September 17, 2014	February 28, 2025	3818
Ethereum	November 9, 2017	February 28, 2025	2669

Several studies have utilized the fear and greed index (FGI), Google search index (GSI), and Twitter data to explore how sentiment influences cryptocurrency prices [40–42]. The crypto fear and greed index (FGI) was introduced by `Alternative.me`, https://alternative.me/crypto/fear-and-greed-index (last accessed on 4 April 2025), on February 1, 2018, so there is no official FGI data available before this date. In order to study the impact of the FGI on cryptocurrency prices prior to February 2018,

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 8 April 2025 doi:10.20944/preprints202504.0484.v1

we propose the following proxy measures using the social media sentiment and Google trends data related to cryptocurrency, specifically for Bitcoin and Ethereum:

$$\text{FGI} = w_1 \text{ Social Media Sentiment} + w_2 \text{ Google Trends}$$

In our research we assign equal weights ($w_1 = w_2 = 0.5$) to both indicators.

We utilized the Twitter Intelligence Tool (Twint) and Twitter's API to collect historical social media data from Twitter (now X) before February 1, 2018 using the following keywords with hashtags, #, and Dollar signs, $: `Crypto`, `Cryptocurrency`, `digital currency`, `Bitcoin`, `Ethereum`, `BTC`, `ETH`. Then we used the open source `Python` library Valence Aware Dictionary and Sentiment Reasoner (VADER) to classify the input statement social media sentiment according to a score ranged from -1 to 1, where a score from 1 to -0.05 stands for Negative sentiment, -0.05 to 0.05 stands for Neutral sentiment, and 0.05 to 1 stands for Positive sentiment. To collect the historical Google Trends data before February 2018, we enter the aforementioned keywords in the search bar of https://trends.google.com/trends (last accessed on 4 April 2025) and calculate the average score to get the daily search interest values, which range from 0 to 100.

After that, we calculate the FGI score using the formula:

$$\text{FGI} = \frac{1}{2}\left[\frac{\text{Score from VADER} + 1}{2} \times 100 + \text{Score Google Trends}\right] \tag{19}$$

The FGI ranges from 0 to 100, where $0-24$ represents extreme market fear, $25-49$ means market fear, $50-74$ indicates market greed, and $75-100$ represents extreme market greed.

Figures 6 shows the daily prices of Bitcoin from September 17, 2014 to February 28, 2025 and Ethereum from November 9, 2017 to February 28, 2025. It can be seen that the prices of both currencies have increased exponentially over time, showing a strong cyclic pattern in relation to FGI. Although the long term trend remains upward, the prices of Bitcoin and Ethereum often decline during fear/extreme fear periods followed by recoveries during greed/extreme greed times. The boxplots in Figure 7 show the distribution of cryptocurrency prices for categories of FGI sentiments. When fear sentiment dominates, prices tend to cluster at lower values (right-skewed), suggesting most trades happen at depressed levels. Extreme fear shows a more balanced distribution (symmetric) but with slightly higher prices, hinting at potential stabilization or cautious buying. Neutral sentiment shifts prices upward (left-skewed), reflecting modest optimism. Greed pushes prices higher (left-skewed), with more trades concentrated at premium values. Finally, extreme greed results in a right-skewed pattern indicating speculative spikes and increase volatility, likely from FOMO-driven buying.

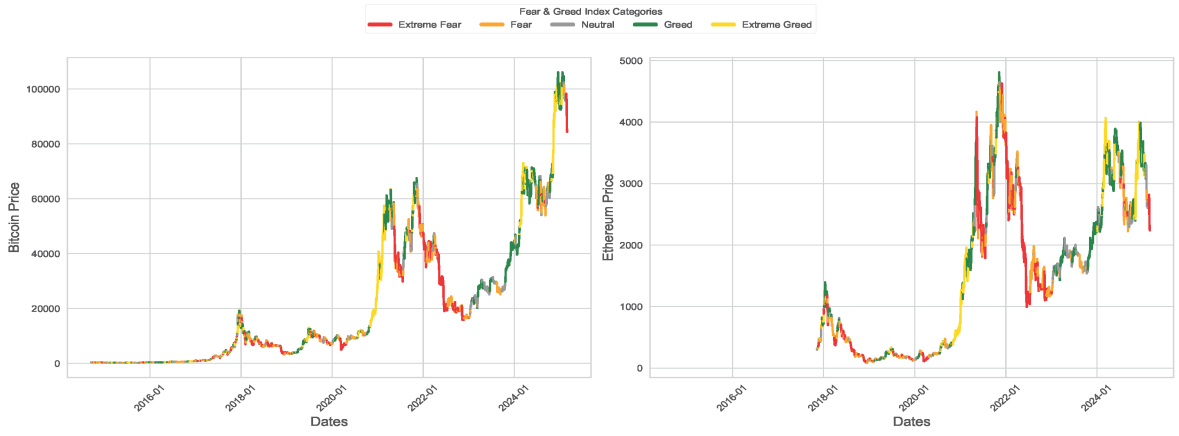

Figure 6. Bitcoin daily prices (left) and Ethereum prices (right), with trends color-coded according to fear and greed index (FGI) categories.

 doi:10.20944/preprints202504.0484.v1

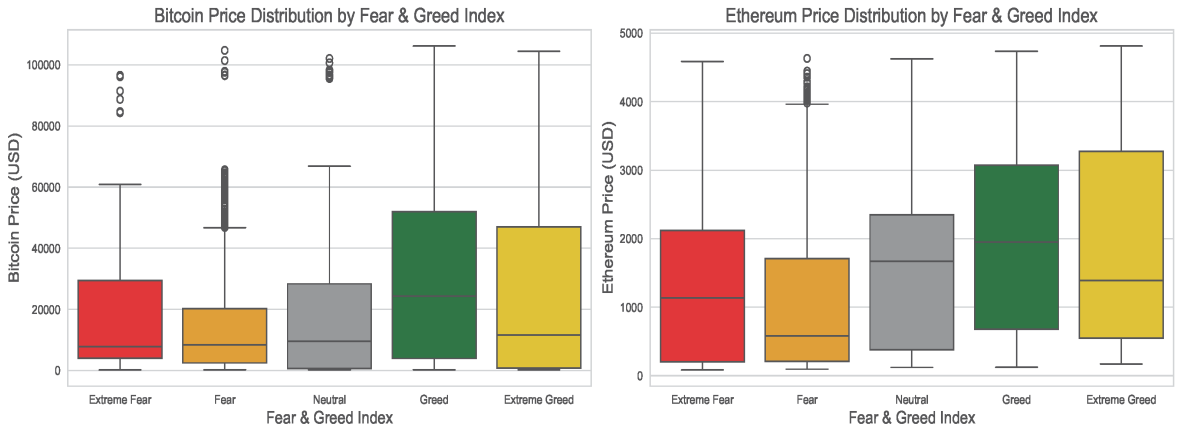

Figure 7. Box-plots of Bitcoin daily prices (left) and Ethereum prices (right) categorized by the fear and greed index (FGI).

We began our analysis by splitting the dataset into an 80-20 train-test split. Then, we normalize both datasets using the Min-Max normalizing:

$$x^{\star} = \frac{x - x_{\min}}{x_{\max} - x_{\min}} \tag{20}$$

After that, we fit the models 2 and 3 using the five neural networks (RBFN, GRNN, BiGRU, BiLSTM, Hybrid transformer + GRU) on the training data and predict the prices of Bitcoin and Ethereum using the testing data. Figures 8 and 9 show the prediction results along with their corresponding 95% prediction interval compared with the actual prices of Bitcoin and Ethereum, respectively. As evident from these figures, we observe that our proposed model outperforms the other approaches in capturing price movements.

 doi:10.20944/preprints202504.0484.v1

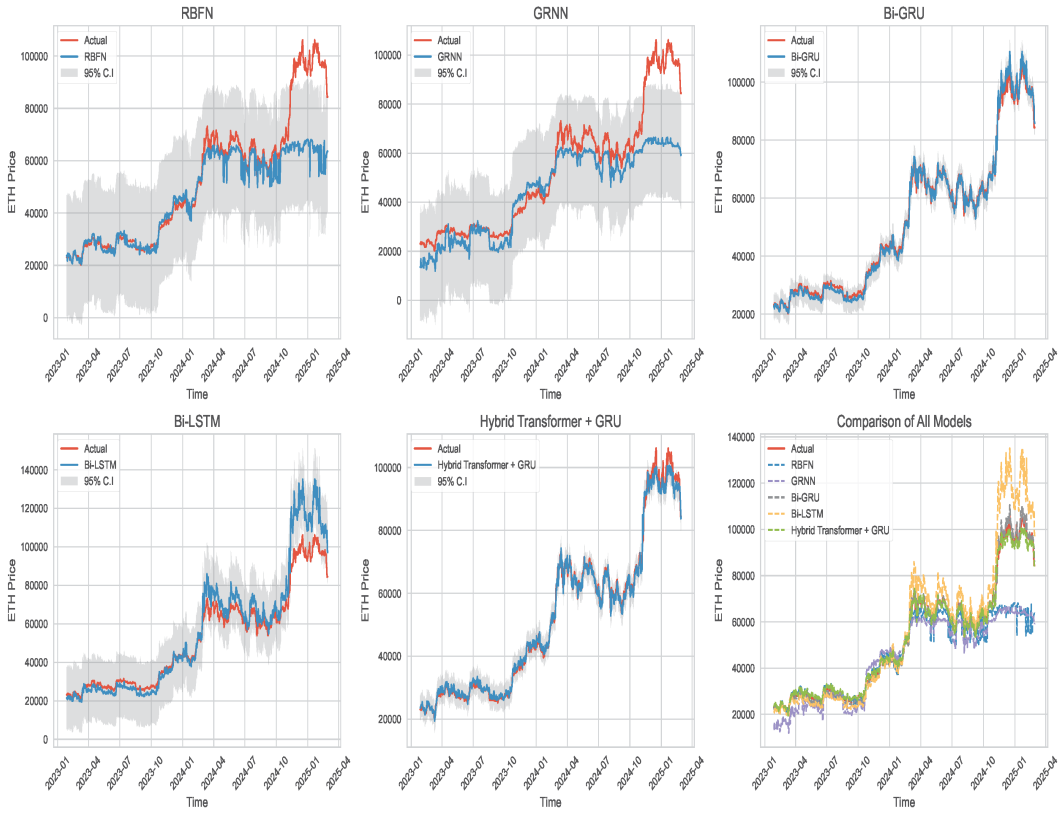

Figure 8. Comparison of the proposed hybrid Transformer-GRU model with four competing deep learning networks for Bitcoin price prediction.

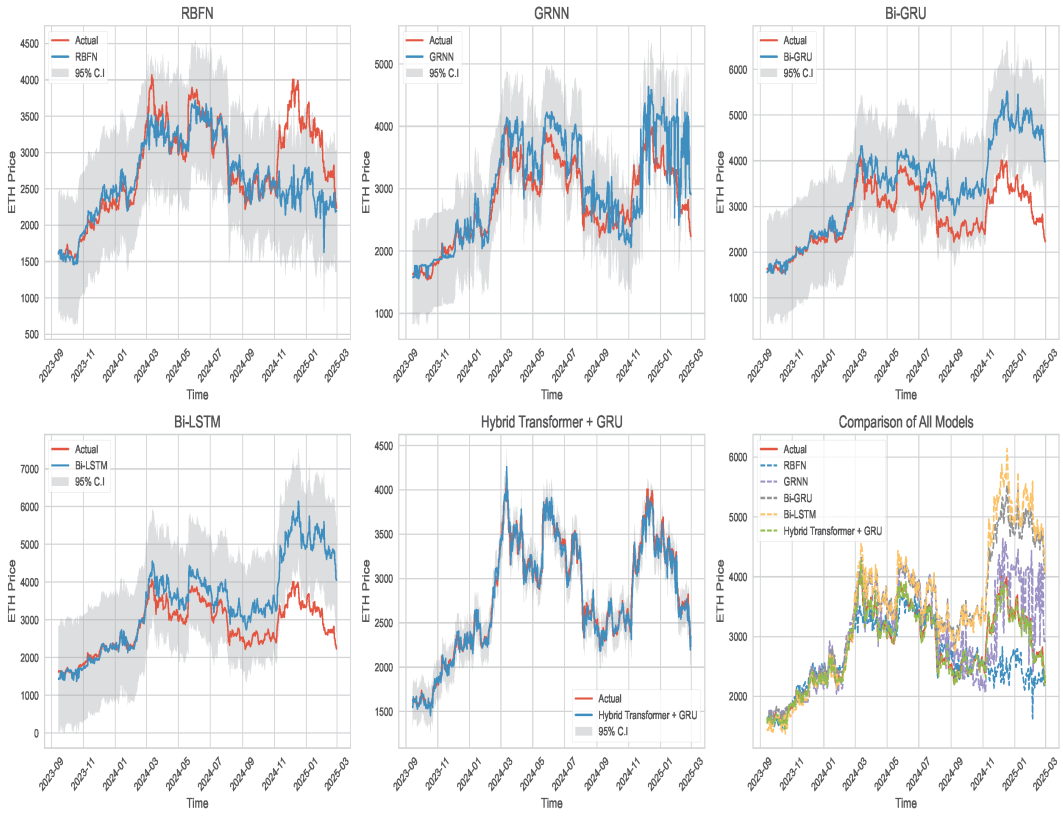

Figure 9. Comparison of the proposed hybrid Transformer-GRU model with four competing deep learning networks for Ethereum price prediction.

 doi:10.20944/preprints202504.0484.v1

In order to measure the prediction accuracy of the five models using the testing data, we calculate the mean squared error (MSE), root mean squared error (RMSE), mean absolute error (MAE), and mean absolute percentage error (MAPE). The smaller the value the metric, the better the prediction neural network model. These metrics can be expressed mathematically as follows:

$$\text{MSE} = \frac{1}{n}\sum_{t=1}^{n}(y_t - \hat{y}_t)^2 \tag{21}$$

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{t=1}^{n}(y_t - \hat{y}_t)^2} \tag{22}$$

$$\text{MAE} = \frac{1}{n}\sum_{t=1}^{n}(\mid y_t - \hat{y}_t \mid) \tag{23}$$

$$\text{MAPE} = \frac{1}{n}\sum_{t=1}^{n}\left(\frac{\mid y_t - \hat{y}_t \mid}{y_t}\right) \times 100\% \tag{24}$$

where y_t and $\hat{y}_t$ are the actual price and predicted price of cryptocurrency at time t. Tables 2−3 present the comparison of performance metrics results for the proposed hybrid transformer + GRU network and the other four neural networks for predicting the prices of Bitcoin and Ethereum respectively. From these two tables, it is seen that the proposed model substantially superior its compositors achieving more precise predictions as it always gives small predictions errors. For example, as shown in Table 2, when comparing the four compositors machine learning models, the BiGRU network achieves the smallest MSE value of 4,358,457 for Bitcoin price prediction. Whereas, the MSE value for the proposed hybrid transformer + GRU model, 3,818,128. For Ethereum price prediction, as shown in Table 3, the proposed model demonstrates excellent performance with an MSE of 11,344.686. This is approximately 18 times smaller than the RBNF model's MSE (203,541.169), 17 times smaller than GRNN's (194,985.943), 60 times smaller than BiGRU's (687,799.984), and 80 times smaller than BiLSTM's (907,844.180). From the results in both tables, we can see that the feedforward neural networks (RBFN and GRNN) struggled to accurately predict Bitcoin prices. On the other hand, both bidirectional models (BiGRU and BiLSTM) didn't perform as well when it came to predicting Ethereum prices.

Table 2. Performance metrics (MSE, RMSE, MAE, and MAPE) for predicting the prices of Bitcoin using different models.

Model	MSE	RMSE	MAE	MAPE
RBFN	1.731258e+08	13157.727	6928.640	9.479
GRNN	1.875502e+08	13694.897	9179.342	15.857
BiGRU	4.358457e+06	2087.692	1559.954	3.271
BiLSTM	8.184621e+07	9046.889	5877.042	9.600
Hybrid Transformer + GRU	3.818128e+06	1954.003	1419.972	2.825

 doi:10.20944/preprints202504.0484.v1

Table 3. Performance metrics (MSE, RMSE, MAE, and MAPE) for predicting the prices of Ethereum using different models.

Model	MSE	RMSE	MAE	MAPE
RBFN	203541.169	451.155	288.453	9.362
GRNN	194985.943	441.572	345.963	11.901
BiGRU	687799.984	829.337	608.416	20.735
BiLSTM	907844.180	952.809	675.427	22.640
Hybrid Transformer + GRU	11344.686	106.511	78.809	2.755

5. Conclusions and Recommendation

The results obtained in this study show better performance of a novel attention-based hybrid Transformer model in predicting cryptocurrency prices compared with two feedforward neural networks (radial basis function, RBFN, and general regression neural network, GRNN) and two deep learning models (bidirectional gated recurrent unit, BiGRU, and bidirectional long short-term memory, BiLSTM). Our proposed model builds on the same Transformer architecture, the foundation behind AI systems like ChatGPT and DeepSeek. We combined the Transformer's ability to capture long-range patterns with the GRU's strength in learning temporal relationships to improve the accuracy of cryptocurrency price predictions. We conducted the comparison study using the two leading digital assets, Bitcoin and Ethereum based on historical data with a one-time-step lag. When predicting Bitcoin prices, the proposed model achieves $6-7$ times lower RMSE, $4-6$ times lower MAE, and $3-6$ times lower MAPE compared to the RBFN and GRNN models. For Ethereum, our model reduces RMSE by about 4 times, MAE by $3-4$ times, and MAPE by $3-4$ times compared to RBFN and GRNN. On the other hand, when we predict the Bitcoin prices, the proposed model achieves $1-5$ times lower RMSE, $1-4$ times lower MAE, and $1-3$ times lower MAPE compared to the BiGRU and BiLSTM models. Moreover for Ethereum, our model reduces RMSE, MAE, and MAPE by approximately $8-9$ times compared to BiGRU and BiLSTM models. This paper opens up several promising directions for future work. For example, our model could be extended to predict the prices of univariate and multivariate other digital assets based on historical data with different time lags. Another potential research topic is to adopt a hybrid Transformer + LSTM model comparing its performance with our proposed Transformer + GRU approach for predicting time series data. Furthermore, future research could explore whether a hybrid Transformer + BiLSTM or Transformer + BiGRU model could enhance time series prediction compared to other deep learning models, including the one proposed in this study. Finally, a crucial future endeavor involves developing a software package in `R` and a `Python` library that embed our models, enabling other researchers to utilize them in their studies. These implementations will undoubtedly facilitate the expansion of research in cryptocurrency price prediction.

Author Contributions: Data curation, E.M. and C.M.-B.; investigation, E.M. and X.C.; formal analysis and methodology, E.M.; writing—original draft, E.M.; writing—review and editing, C.M.-B., and X.C.; All authors have read and agreed to the submitted version of the manuscript.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The analyzed data and used codes are available under request.

Acknowledgments: The authors would also like to thank the editor and reviewers for their constructive comments which led to improving the presentation of the manuscript.

Conflicts of Interest: The authors declare no conflict of interest.

 doi:10.20944/preprints202504.0484.v1

Funding: The project has no fund.

References

1. Razi, M.A., Athappilly, K. (2005). A comparative predictive analysis of neural networks (NNS), nonlinear regression and classification and regression tree (CART) models. *Expert Systems with Applications*, 29, 65–74.
2. Ślepaczuk, R., Zenkova, M. (2018). Robustness of support vector machines in algorithmic trading on cryptocurrency market. *Central European Economic Journal*, 5, 186–205.
3. Chen, Z., Li, C., Sun, W. (2020). Bitcoin price prediction using machine learning: An approach to sample dimension engineering. *Journal of Computational and Applied Mathematics*, 365, 112395.
4. Mahdi, E., Leiva, V., Mara'Beh, S., Martin-Barreiro, C. (2021). A New Approach to Predicting Cryptocurrency Returns Based on the Gold Prices with Support Vector Machines during the COVID-19 Pandemic Using Sensor-Related Data. *Sensors*, 21, 6319. https://doi.org/10.3390/s21186319.
5. Akyildirim, E., Goncu, A., Sensoy, A. (2021). Prediction of cryptocurrency returns using machine learning. *Annals of Operations Research*, 297, 3–36. https://doi.org/10.1007/s10479-020-03575-y.
6. Makala, D., Li, Z. (2021). Prediction of gold price with ARIMA and SVM. *Journal of Physics: Conference Series*, 1767, 012022.
7. Jaquart, P., K¨opke, S., Weinhardt, C. (2022). Machine learning for cryptocurrency market prediction and trading. *The Journal of Finance and Data Science*, 8: 331–352. https://doi.org/10.1016/j.jfds.2022.12.001.
8. Mahdi, E., Al-Abdulla, A. (2022). Impact of COVID-19 Pandemic News on the Cryptocurrency Market and Gold Returns: A Quantile-on-Quantile Regression Analysis. *Econometrics*, 10, 26. https://doi.org/10.3390/econometrics10020026.
9. Qureshi, M., Iftikhar, H., Rodrigues, P. C., Rehman, M. Z., Salar, S. A. A. (2024). Statistical Modeling to Improve Time Series Forecasting Using Machine Learning, Time Series, and Hybrid Models: A Case Study of Bitcoin Price Forecasting. *Mathematics*, 12(23), 3666. https://doi.org/10.3390/math12233666.
10. Broomhead, D. S., Lowe, D. (1988). Radial basis functions, multi-variable functional interpolation and adaptive networks (Technical report). *Royal Signals and Radar Establishment (RSRE)*, Memorandum 4148.
11. Broomhead, D. S.; Lowe, D. (1988). Multivariable functional interpolation and adaptive networks. *Complex Systems*, 2: 321–355.
12. Alahmari, S., A., (2020). Predicting the Price of Cryptocurrency Using Support Vector Regression Methods. *Journal of Mechanics of Continua and Mathematical Sciences*, 15(4): 313–322. https://doi.org/10.26782/jmcms.2020.04.00023.
13. Casillo, M., Lombardi, M., Lorusso, A., Marongiu, F., Santaniello, D., Valentino, C. (2022). Sentiment Analysis and Recurrent Radial Basis Function Network for Bitcoin Price Prediction. *IEEE 21st Mediterranean Electrotechnical Conference (MELECON), Palermo, Italy*, pp. 1189–1193, 10.1109/MELECON53508.2022.9842889.
14. Zhang, Y. (2025). Stock price behavior determination using an optimized radial basis function. *Intelligent Decision Technologies*, 1–18. doi:10.1177/18724981251315846
15. Specht, D. F. (1991). A general regression neural network. *IEEE Transactions on Neural Networks*, 2(6): 568–576. doi:10.1109/72.97934.
16. Martínez, F., Charte, F., Rivera, A.J., Frías, M.P. (2019). Automatic Time Series Forecasting with GRNN: A Comparison with Other Models. *In: Rojas, I., Joya, G., Catala, A. (eds) Advances in Computational Intelligence. IWANN 2019, Lecture Notes in Computer Science()*, 11506. Springer, Cham. https://doi.org/10.1007/978-3-030-20521-8_17.
17. Martínez, F., Charte, F., Frías, M., P., Martínez-Rodríguez, A. M. (2022). Strategies for time series forecasting with generalized regression neural networks. *Neurocomputing*, 49: 509-521 https://doi.org/10.1016/j.neucom.2021.12.028.
18. Hochreiter, S., Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, 9(8): 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735.
19. McNally, S., Roche, J., Caton, S. (2018). Predicting the Price of Bitcoin Using Machine Learning. *26th Euromicro International Conference on Parallel, Distributed and Network-based Processing (PDP)*, 339–343. https://api.semanticscholar.org/CorpusID:206505441
20. Liu, Y., Gong, C., Yang, L., Chen, Y. (2020). DSTP-RNN: A dual-stage two-phase attention-based recurrent neural network for long-term and multivariate time series prediction. *Expert Systems with Applications*, 143, 113082

 doi:10.20944/preprints202504.0484.v1

21. Zoumpekas, T. Houstis, E., Vavalis, M. (2020). ETH analysis and predictions utilizing deep learning. *Expert Systems with Applications*, 162: 113866. https://doi.org/10.1016/j.eswa.2020.113866.
22. Lahmiri, S., Bekiros, S. (2019). Cryptocurrency forecasting with deep learning chaotic neural networks. *Chaos, Solitons & Fractals*, 118, 35–40. https://doi.org/10.1016/j.chaos.2018.11.014.
23. Ji, S., Kim, J., Im, H. (2019). A Comparative Study of Bitcoin Price Prediction Using Deep Learning. *Mathematics*, 7(10), 898. https://doi.org/10.3390/math7100898.
24. Uras, N., Marchesi, L., Marchesi, M., Tonelli, R. (2020). Forecasting Bitcoin closing price series using linear regression and neural networks models. *PeerJ Computer Science*, 6: e279. https://doi.org/10.7717/peerj-cs.279.
25. Lahmiri, S., and Bekiros, S. (2021). Deep learning forecasting in cryptocurrency high frequency trading. *Cognitive Computation*, 13: 485–487.
26. Cho, K., Merrienboer, B., Gulcehre, C., Bahdanau, D., Fethi, B., Holger, S., Bengio, Y. (2014). Learning phrase representations using RNN encoder- decoder for statistical machine translation. https://arxiv.org/abs/1406.1078.
27. Jianwei, E., Ye, J., Jin, H. (2019). A novel hybrid model on the prediction of time series and its application for the gold price analysis and forecasting. *Physica A: Statistical Mechanics and its Applications*, 527, 121454. https://doi.org/10.1016/j.physa.2019.121454.
28. Dutta, A., Kumar, S., Basu, M. (2020). A gated recurrent unit approach to bitcoin price prediction. *Journal of Risk and Financial Management*, 13(2): 23. https://doi.org/10.3390/jrfm13020023.
29. Tanwar, S., Patel, N. P. , Patel, S. N., Patel, J. R., Sharma, G., Davidson, I.E. Deep Learning-Based Cryptocurrency Price Prediction Scheme With Inter-Dependent Relations. *IEEE Access*, 9: 138633–138646. 10.1109/ACCESS.2021.3117848.
30. Ye, Z., Wu, Y., Chen, H., Pan, Y., Jiang, Q. (2022). A Stacking Ensemble Deep Learning Model for Bitcoin Price Prediction Using Twitter Comments on Bitcoin. *Mathematics*, 10(8), 1307. https://doi.org/10.3390/math10081307.
31. Patra1, G., R., Mohanty, M., N., (2023). Price Prediction of Cryptocurrency Using a Multi-Layer Gated Recurrent Unit Network with Multi Features. *Computational Economics*, 62: 1525–1544. https://doi.org/10.1007/s10614-022-10310-1.
32. Hansun, S., Wicaksana, A ., Khaliq, A.Q.M. (2022). Multivariate cryptocurrency prediction: comparative analysis of three recurrent neural networks approaches. *Journal of Big Data*, 9, 50.
33. Ferdiansyah, F., Othman, S. H., Radzi, R. Z., M., Stiawan, D., Sutikno T., (2023). Hybrid gated recurrent unit bidirectional-long short-term memory model to improve cryptocurrency prediction accuracy. *IAES International Journal of Artificial Intelligence (IJ-AI)*, 12 (1). http://doi.org/10.11591/ijai.v12.i1.pp251-261.
34. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Polosukhin, I. (2017). Attention is all you need. *Advances in Neural Information Processing Systems*, 30.
35. Devlin, J., Chang, M. W., Lee, K., Toutanova, K. (2018). BERT: Pre-training of deep bidirectional transformers for language understanding. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics. 10.18653/v1/N19-1423.
36. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., Houlsby, N. (2020). An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. ArXiv, abs/2010.11929.
37. Zhou, H., Zhang, S., Peng, J., Zhang, S., Li, J., Xiong, H., Zhang, W. (2021). Informer: Beyond Efficient Transformer for Long Sequence Time-Series Forecasting. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(12), 11106–11115. https://doi.org/10.1609/aaai.v35i12.17325.
38. Grigsby, J., Wang, Z., Qi, Y. (2021). Long-Range Transformers for Dynamic Spatiotemporal Forecasting. *computer science bibliography*. ArXiv, https://arxiv.org/abs/2109.12218.
39. Lezmi, E., Xu, J. (2023). Time Series Forecasting with Transformer Models and Application to Asset Management. Available at SSRN: https://ssrn.com/abstract=4375798 or http://dx.doi.org/10.2139/ssrn.4375798.
40. Kristoufek, L. (2013). BitCoin meets Google Trends and Wikipedia: Quantifying the relationship between phenomena of the Internet era. *Scientific Reports*, 3, 3415. https://doi.org/10.1038/srep03415.

 doi:10.20944/preprints202504.0484.v1

41. Urquhart, A. (2018). What causes the attention of Bitcoin?. *Economics Letters*, 166, 40–44, https://doi.org/10.1016/j.econlet.2018.02.017.
42. Kao, Y.S., Day, M.Y., Chou, K.H. (2024). A comparison of bitcoin futures return and return volatility based on news sentiment contemporaneously or lead-lag. *The North American Journal of Economics and Finance*, 72, 102159. https://doi.org/10.1016/j.najef.2024.102159.
43. Colah Understanding LSTM Networks. (2015). Available online: http://colah.github.io/posts/2015-08-Understanding-LSTMs/ (accessed on March 01, 2024).