*Cryptocurrency price prediction:*

*Using Statistical*

*&*

*Machine Learning algorithms*

1st Ho Thi Hang   
*IS403.N22.HTCL-EN  
University of Information Technology*Ho Chi Minh City  
[20521285@gm.uit.edu.vn](mailto:20521285@gm.uit.edu.vn)

2st Nguyen Hoang Ngoc Hanh   
*IS403.N22.HTCL-EN  
University of Information Technology*Ho Chi Minh City  
[20521287@gm.uit.edu.vn](mailto:20521287@gm.uit.edu.vn)

3st Duong Nhat Minh   
*IS403.N22.HTCL-EN*  
*University of Information Technology*  
Ho Chi Minh City  
[20520242@gm.uit.edu.vn](mailto:20521242@gm.uit.edu.vn)

*Abstract*— Cryptocurrency has always been an area that attracts a lot of research. To further research on crypto price prediction, this article focuses on the top three cryptocurrencies: Ethereum (ETH), Binance Coin (BNB) and Ripple (XRP) by applying the methods and Predictive model, we conduct analysis of factors affecting prices and make future price forecasts. This study will use models such as Linear Regression (LN), Autoregressive Integrated Moving Average (ARIMA), Gated Recurrent Unit (GRU), Holt-Winters Exponential Smoothing(HW), Fuzzy Time Series (FTS), Recurrent Neural Network (RNN), Gaussian Process Regression (GPR), Bayesian structural time series (BSTS), Fully Convolutional Network (FCN) to forecast the closing prices of three cryptocurrencies ETH, BNB, XRP. The comparison results will be based on 3 evaluation parameters: RMSE, MAE and MAPE

Keywords— Linear regression, ARIMA, GRU, RNN, GPR, FCN, BSTS, FTS, Holt-Winters, Exponential Smoothing, Statistical, ETH, BNB, XRP, MAPE, MAE, RMSE

# Introduction

Cryptocurrencies have become an integral part of the global financial system, attracting the interest and participation of millions of people around the world. Among the popular cryptocurrencies, Ethereum (ETH), Binance Coin (BNB) and Ripple (XRP) have emerged as luminaries in the blockchain technology industry. The growth and potential of these cryptocurrencies has led investors and users to expect growth and profitability from holding them.

This paper uses statistical models such as LN, ARIMA, Fuzzy Time Series, Holt-Winters, and machine learning such as GRU, RNN, FCN, BSTS, GPR to predict closing prices closing of cryptocurrencies (ETH, BNB and XRP).

Based on detailed information and analysis, this article hopes to provide an overview of the price outlook of ETH, BNB and XRP, and help readers better understand the risks and opportunities. invest in this potential cryptocurrency sector.

# Related Work

Poongodi M. et al. [1] conducted Prediction of the price of Ethereum blockchain cryptocurrency in an industrial finance system, accepted 25 November 2019 at International journal Computers & Electrical Engineering of academic publishing company Elsevier BV. In this study, price prediction is performed with two machine learning methods, namely linear regression (LN) and support vector machine (SVM), by using a time series consisting of daily ether cryptocurrency closing prices. When using the proposed model, the SVM method has a higher accuracy (96.06%) than the LN method (85.46%).

Saadah et al. [2]applied several machine learning and deep learning methods for predicting the price of Bitcoin, Ethereum, and Ripple. The methods include k-nearest neighbors, support vector machine, and LSTM. The experimental results demonstrated that the LSTM achieved the optimal RMSE for all three cryptocurrencies, with RMSE of 928.62 on Bitcoin, 11.69 on Ethereum, and 0.16 on Ripple

V. Derbentsev et al. [3] conducted Bitcoin (BTC), Ethereum (ETH) and Ripple (XRP) forecasts from the 19th of August 2013 until the 19th of July 2016 throught RNN and LSTM. This study adjusted the units and epochs and used the MAPE. The result shows that the out of sample accuracy of short-term forecasting daily prices obtained by RNN and LSTM in terms of MAPE for three of the most capitalized cryptocurrencies (BTC, ETH, and XRP) was within 0.92-2.61%.

Kiki Ramadani, Dodi Devianto [4] present a forecasting model that utilizes fuzzy logic to estimate the price of Bitcoin in NTERNATIONAL CONFERENCE ON SCIENCE AND APPLIED SCIENCE (ICSAS2020) with the FTS Markov Chain. In this research article, the result of FTS Markov Chain has the best a MAPE with 8.8%, but the FTS Segmented Chen Logical Method is the the best among because of a MAE and a RMSE are better (355.51, 678.01).

Mohammad J. Hamayel and Amani Yousef Owda [5] present an innovative approach to predicting cryptocurrency prices by utilizing machine learning algorithms such as GRU, LSTM, and bi-LSTM, demonstrating promising results in the field of price forecasting. A RMSE, MAPE of GRU model for BTC: 174,129 and 0.2454%; ETH: 26.59 and 0.8267%, LTC: 0.825 and 0.2116%, so GRU can predict cryptocurrency prices better than LSTM and bi-LSTM but overall all algorithms represent excellent predictive results.

Dian Utami Sutiksno et al. [6] had a notable study in the field of forecasting the historical data of Bitcoin. This paper utilizes the ARIMA method and the α-Sutte indicator to predict the trends and fluctuations of Bitcoin price in the future. In this study, ARIMA is not the method has the best accuracy to predict historical data of bitcoin with a MSE, MAE are 295797,315 and 497,657.

# Modeling

## Linear Regression

***Linear Regression*** algorithm has been around for a long time and has been developed since the 1800s. However, the formal formation and development of Linear Regression as it is today started from the famous work of mathematicians and statisticians. British scholar Francis Galton in the late 19th century.

Francis Galton applied linear regression to study the relationship between parent and child height in predicting human height. This work opened a new direction in applying linear regression analysis and laid the foundation for the development of Linear Regression.

There are two types of linear regression: simple linear regression and multivariable linear regression.

***Simple linear regression*** is a method of establishing a relationship between a dependent variable Y and an independent variable X. Represented by the equation:

Y = β0 + β1X + e

***Multivariable linear regression*** is a method to establish a relationship between the dependent variable Y and many independent variables X. Represented by the equation:

Y = β0 + β1X1 + β2X2 +…+ βnXn+ e

Where:

* Y: is the dependent variable
* X1, X2,..Xn : are independent variables
* β0: is the original coefficient
* β1, β2,… βn: are regression coefficients
* e : is the random error

Linear regression can be used to predict the value of the dependent variable based on the independent variables. With a linear relationship defined, a regression model can be used to generate predictions and estimate future values.

## ARIMA

ARIMA or An auto regressive integrated moving average, is a statistical analysis model that uses time series data to either better understand the data set or to predict future trends.[13]

ARIMA integrates two processes: the p-order autoregressive process – AR(p) and the q-order moving average process – MA(q). Otherwise, it is necessary to use false integration analysis I(d) (aka the automatic operator) to make the time series stationary.

*Auto Regressive (AR) model (p):*

Where:

: is a constant.

: is the lag of the series.

: are the parameters.

: is white noise

*Moving Average (MA) model (q):*

Where:

: is a constant.

: is the lag of the series.

: is white noise

*Auto regressive integrated moving average(ARIMA) model (p,q,d):*

The ARIMA model is not a perfect forecasting model for any time series and only works best if the data is time-dependent and the forecast is time-type, random data are often little work with the ARIMA model.

## Holt-Winters Exponential Smoothing

**Holt-Winters Exponential Smoothing** was first suggested by Holt's student, Peter Winters, in 1960 after reading a signal processing book from the 1940s on exponential smoothing.[11]

Holt-Winters Exponential Smoothing is used to handle the time series data containing a seasonal component. This method is based on three smoothing equations: stationary component, trend, and seasonal. Both seasonal and trend can be additive or multiplicative.[12]

Holt-Winter's Additive Method is given by the below formula:

Xt = αYt / St – m + (1 – α)(Xt-1 + Tt-1)

Tt = β(Xt – Xt-1) + (1 – β)Tt-1

St = γYt / Xt + (1 – γ)St – m

Ft+ h = (Xt + h\*Tt) \* St+h-m

Where:

* α : Data smoothing factor. The range is 0 < α <1.
* β : Trend smoothing factor. The range is 0 < β <1.
* γ : Seasonal change smoothing factor. The range is 0 < γ <1.
* Xt is the estimated value at time t
* Xt-1 is the estimated value at time t-1
* Yt is the actual value at time t
* St – m is the estimated value for the seasonality index
* Tt-1 is the trend component at time t-1
* Tt is the trend component at time t
* St is the seasonal component at time t
* m is the period in the season

*\*Note:* α, β, and γ are the constants that must be estimated in such a way that the MSE of the error is minimized.

Holt-Winters Exponential Smoothing is a forecasting method specifically used for time series data with seasonal components. It has the ability to model and forecast variations with seasonal cycles and long-term trends. This method is flexible and stable, allowing for easy adjustment and ensuring the continuity of forecasts. It is also computationally simple and efficient, enabling fast processing of time series data.

## Fuzzy Time Series

In 1965, Zadeh proposed the concept of **fuzzy sets** as a tool to test the unknown degree of membership. **The fuzzy time series** is an analysis method derived from the concept of fuzzy sets.

In 1993, Song and Chissom successfully combined the concept of fuzzy sets with the time series model and began studies on fuzzy time series.

**Definition 1.** Let Y(t)eit' (t=0,1,2, ) be a time series. If f1 (t) is a fuzzy set in Y(t) and F(t)={f1 (t), f2 (t), }, then F(t) is called a fuzzy time series in Y(t).

**Definition 2.** Suppose F(t) is caused by F(t-1) only, i.e.,F(t-1)àF(t). Then this relation can be expressed as F(t)=F(t-1)0R(t,t-1) where R(t,t-1) is a fuzzy relationship and is called the first-order model of F(t).

**Definition 3.** Suppose R(t,t-1) is a first-order model of F(t). If for any t R(t,t-1) is independent on t, i.e., for any t, R(t,t-1) = R(t-1,t-2), then F(t) is called a timeinvariant fuzzy time series or else it is called a time-variant fuzzy time series.

**The fuzzy time series** can be used as prediction tools in a real-life problem or cases where historical data are formed in linguistic value.

***Order of forecasting:***

+ Determine the background space U

U = [Dmin - D1,Dmax + D2]

+ Divide U into corresponding intervals.

+ Identify fuzzy sets Aj

+ Blur existing time series.

+ Build and group fuzzy relationships.

+ Identify groups of fuzzy relationships using forecasting.

+ Forecasting and defuzzification.

## Bayesian Structural Time Series

The report of West and Harrison (1997) used a linear model method based on Bayesian probability theory for time series, and this can be considered a first step in the development of Bayesian Structural Time Series BSTS.

One of the most important works is a paper by Scott and Varian titled "Predicting the Present with Bayesian Structural Time Series" (2014). In this paper, they introduce BSTS and describe the approach and application of BSTS in time series modeling.

Bayesian structural time series (BSTS) model is a statistical technique used for feature selection, time series forecasting, nowcasting, inferring causal impact and other applications. The model is designed to work with time series data.

For the value data sets of virtual currencies, in general, the data sets do not have a certain uptrend or downtrend, nor are they seasonal, so the team that selects the components for the model includes:

+ Autoregressive components

+ Dynamic

From there we have the general formula:

**Y(t) = dynamic\_component(t) + autoreg\_component(t)**

Bayesian structural time series is an amalgamation of

time series models using the state-space representation

and Bayesian statistics.

Bayesian structural time series is an amalgamation of

time series models using the state-space representation

and Bayesian statistics.

## Recurrent Neural Network

Wilhelm Lenz and Ernst Ising created and analyzed the Ising model (1925) which is essentially a non-learning artificial **recurrent neural network** (RNN) consisting of neuron-like threshold elements. In 1972, Shun'ichi Amari made this architecture adaptive. His learning RNN was popularised by John Hopfield in 1982.

**Recurrent Neural Network** is a type of Neural Network where the output from the previous step is fed as input to the current step. In traditional neural networks, all the inputs and outputs are independent of each other, but in cases when it is required to predict the next word of a sentence, the previous words are required and hence there is a need to remember the previous words. Thus RNN came into existence, which solved this issue with the help of a Hidden Layer.

The main and most important feature of RNN is its Hidden state, which remembers some information about a sequence. The state is also referred to as Memory State since it remembers the previous input to the network. It uses the same parameters for each input as it performs the same task on all the inputs or hidden layers to produce the output. This reduces the complexity of parameters, unlike other neural networks.

A diagram of a diagram

Description automatically generated with low confidence

1. **Recurrent Neural Network**

A picture containing text, screenshot, diagram, circle

Description automatically generated

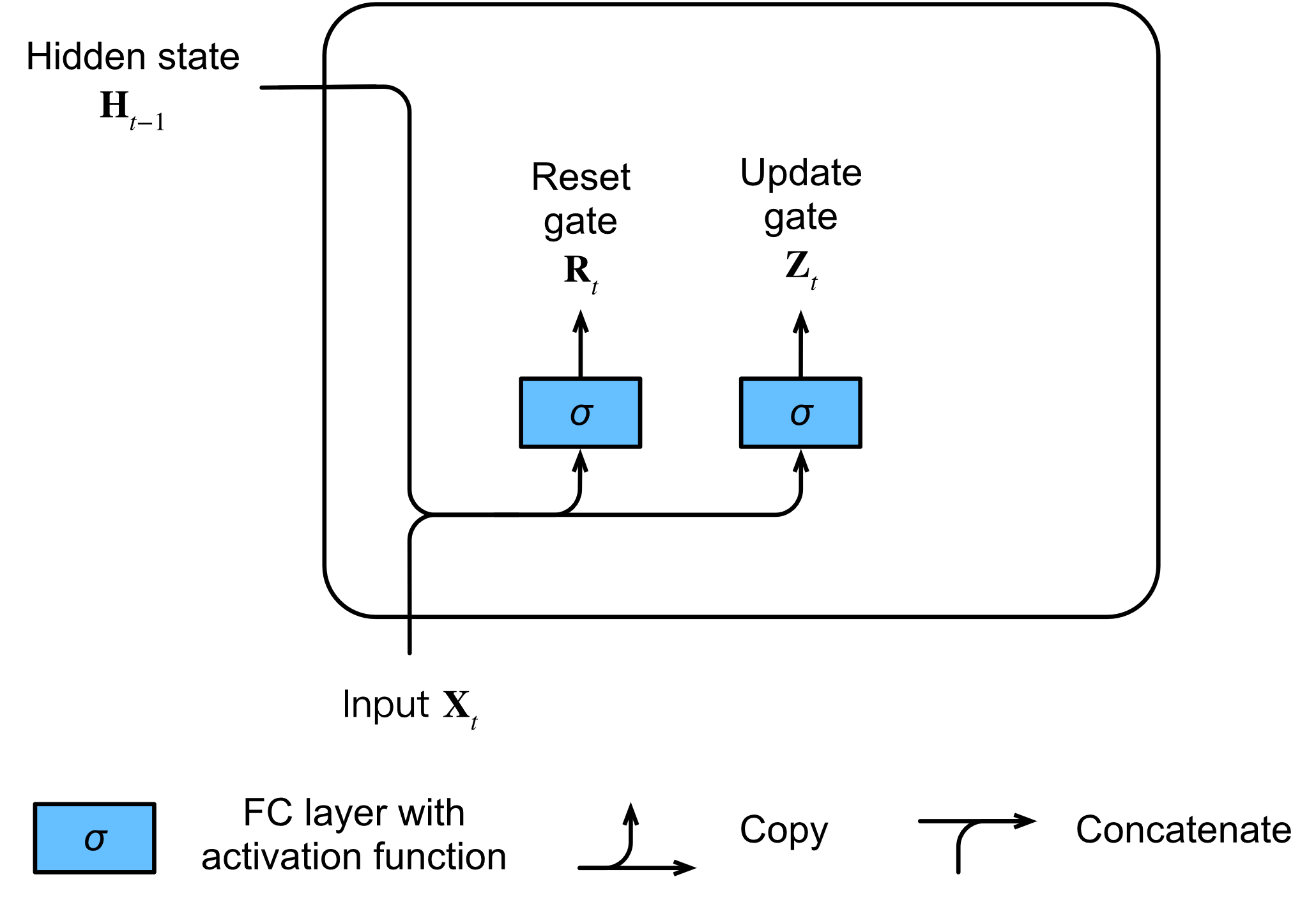
1. **Fully connected Recurrent Neural Network**

## GRU

Gated recurrent units (GRUs) are a gating mechanism in recurrent neural networks, introduced in 2014 by Kyunghyun Cho et al. The GRU is like a long short-term memory (LSTM) with a forget gate, but has fewer parameters than LSTM, as it lacks an output gate. GRU's performance on certain tasks of polyphonic music modeling, speech signal modeling and natural language processing was found to be similar to that of LSTM.[14]

**Reset gate and update gate model:**

The reset gate controls how much of the previous state we might still want to remember. Likewise, an update gate would allow us to control how much of the new state is just a copy of the old state.



1. Computing the reset gate and the update gate in a GRU model.

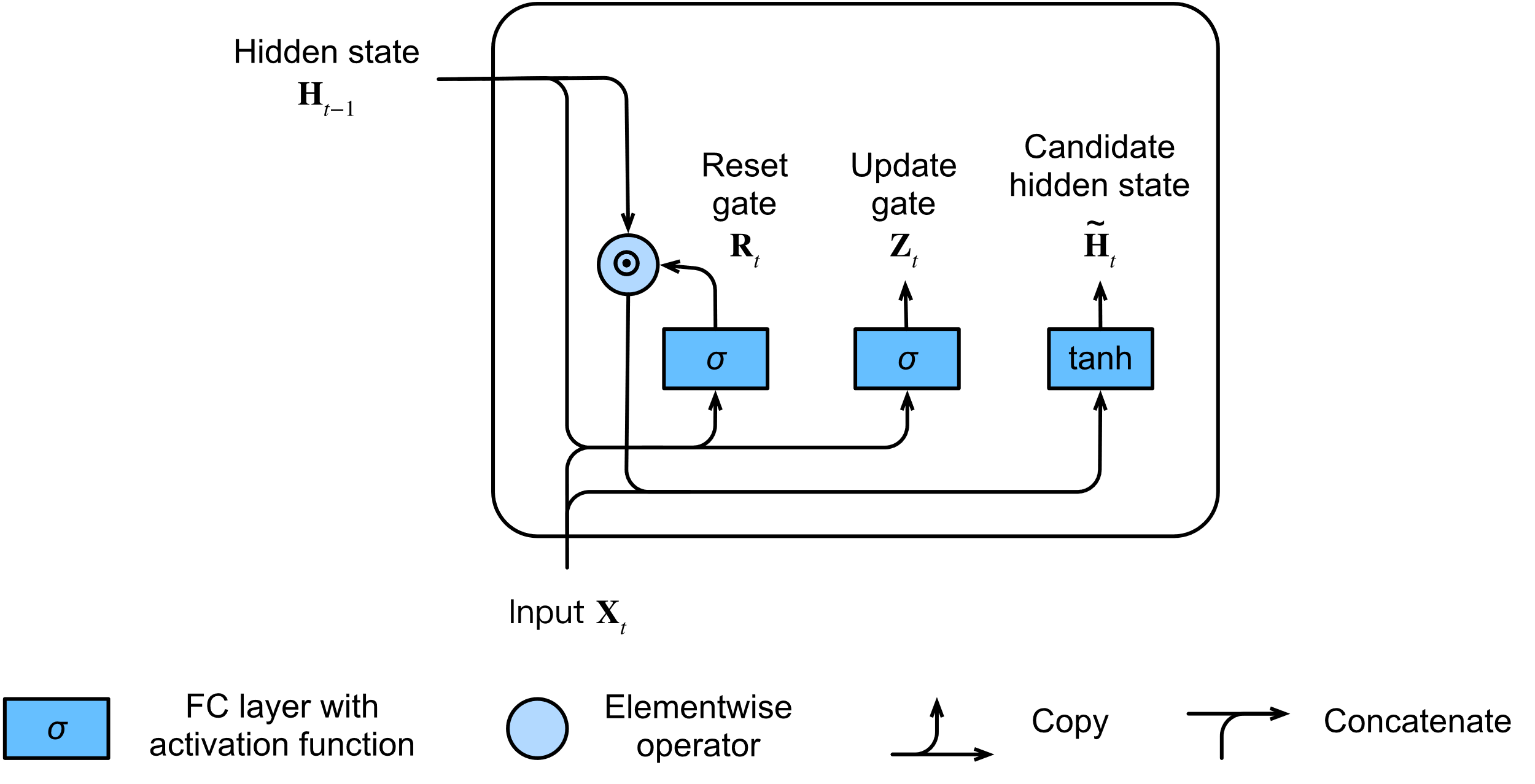
Mathematically, for a given time step ***t,*** suppose that the input is a minibatch (number of examples: n, number of inputs: d) and the hidden state of the previous time step is (number of hidden units: h).Then reset gate and update gate are computed as follows:

,

,

Where and , are weight parameters and  are bias parameters.

**Candidate Hidden State model:**

****

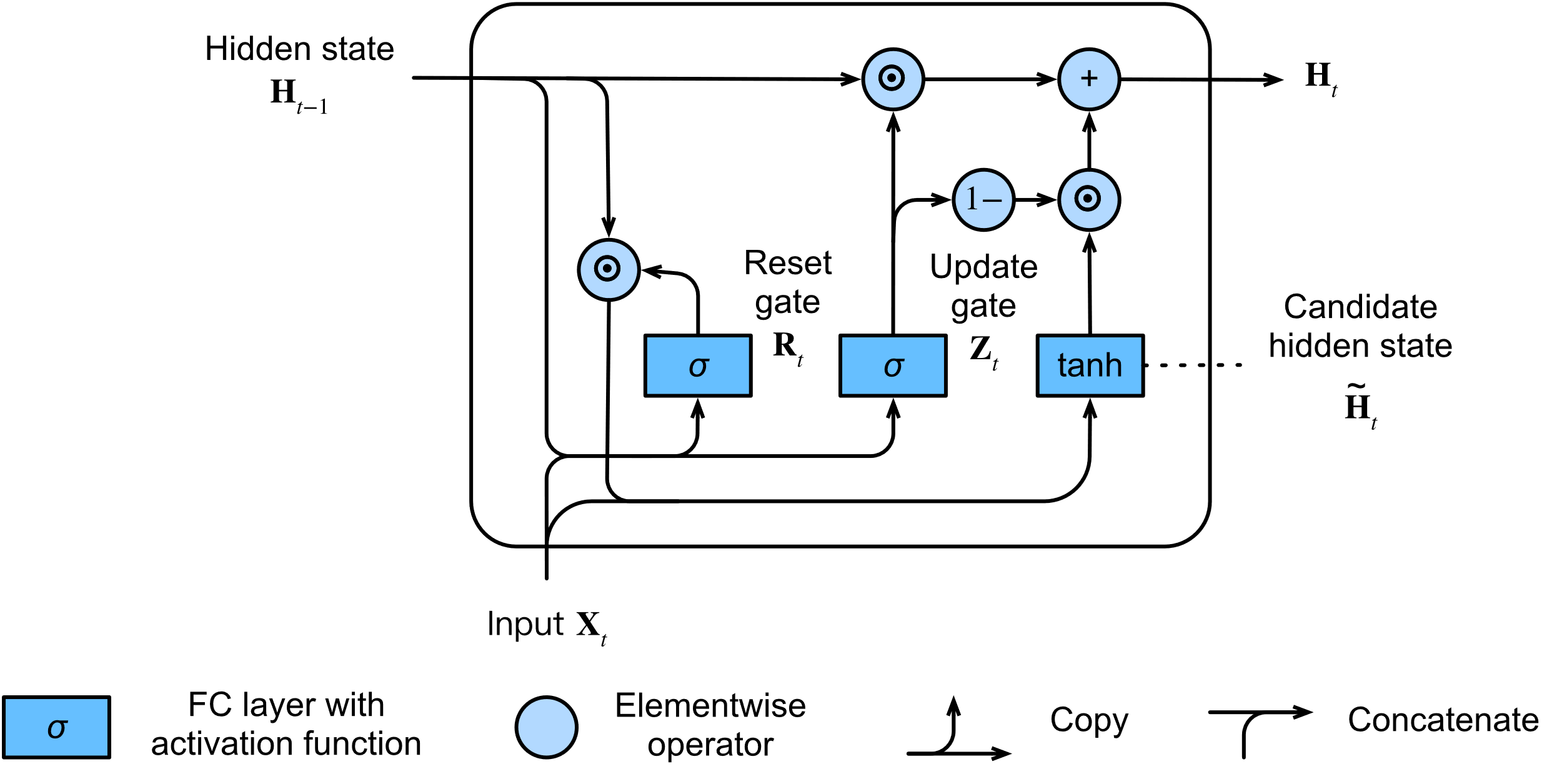
1. **Computing the candidate hidden state in a GRU model.**

integrate the reset gate, leading to the following candidate hidden state at time step ***t:***

Where and are weight parameters,

is the bias, and the symbol is the Hadamard (elementwise) product operator. Here we use a tanh activation function.

**Hidden state model:**

****

1. Computing the hidden state in a GRU model.

incorporate the effect of the update gate . This determines the extent to which the new hidden state matches the old state versus how much it resembles the new candidate state . The update gate can be used for this purpose, simply by taking elementwise convex combinations of This leads to the final update equation for the GRU:

)

In summary, GRUs have the following two distinguishing features:

* Reset gates help capture short-term dependencies in sequences.
* Update gates help capture long-term dependencies in sequences.[15]

## GPR

The distribution in the Gaussian process is represented by an expectation function m(x) and a covariance function k(x,).

In fact, we often consider random variables with expectation m(x) = 0 and are only interested in to the covariance function [16]

f(x) ~ GP (0, k(x,))

where k(x,)=E[f(x)f()] represents the correlation between the outputs f(x) and f) with the input variables x and , respectively.

The matrix K that represents the correlation between all input variables is called the covariance matrix of size n × n. Covariance function formula:

k(x,) =

This covariance function has two hyperparameters θ=(,l). To evaluate the hyperparameters, we use the Bayesian probability formula as follows:

p(f |Y,X)=

where p(f|Y,X) is called the posterior probability, p(Y|X,f) is called the probability probability (likelihood), p(f|X) is called the a priori probability, and p(Y|X) is called the marginal likelihood.[17]

## Fully convolutional network

***Fully convolutional network (FCN)*** is deep learning approaches that take advantage of convolutional neural networks (CNN) for end-to-end classification of univariate time series.

The core component of FCN is the convolutional layer, the formula for calculating the convolutional layers is as follows:

Ảnh có chứa văn bản, Phông chữ, biểu đồ, màu trắng

Mô tả được tạo tự động[9]

Where:

* 𝑐𝑜𝑛𝑣(𝑖,𝑗) is the convolution result, also known as the feature map;
* M indicates the size of the convolution kernel (M × M);
* 𝑤𝑢,𝑣 is the weight of the convolution kernel in line 𝑢 and column 𝑣 ;
* 𝑥𝑖+𝑢,𝑗+𝑣 is the input;
* 𝑏 is the bias;

Ảnh có chứa văn bản, ảnh chụp màn hình, hàng, biểu đồ

Mô tả được tạo tự động

1. Typical architecture of FCN.

A convolutional block contains a continuous M convolutional layers and p pooling layers. In a convolutional network, N continuous convolutional blocks can be stacked, followed by q fully convolutional layers and s deconvolutional layers.[9]

# Method

***Introduce:*** Input is time series historical data of cryptocurrencies. After the data goes through preprocess splits the data according to the training, validation, and testing dataset accordingly, then will use Linear Regression, ARIMA, GRUs, RNN, Fuzzy Time Series, BSTS, FCN, Holt-Winters Exponential Smoothing, GPR models to train. The selected model is trained using the prepared data set.

During training, the model learns the underlying patterns, relationships, and structures in the data. The training process involves tuning the model’s parameters to minimize the difference between the predicted output and the real output from the training data. After the model is trained, it needs to be evaluated to evaluate its performance and generalizability.

If after evaluating the model performance is not effective then go back to adjust the hyperparameters and then do the training again.

***Order of process:***

- Step 1: Collect ime series data: data have a date attribute and values.

- Step 2: Slit data into 3 part for training implement: train data, test data, validate data with different ratio 7:2:1, 6:2:2, 5:3:2

- Step 3: Choose one model, build, tune the hyperparameters to ready for training.

- Step 4: Train model.

- Step 5: Predict base on model.

- Step 6: If the the result is error values, go back to step 3, else, performance estimations and save it.

- Step 7: Choose another model and go back step 2.

A picture containing diagram, sketch, line, plan

Description automatically generated

1. Order of prediction

# experiment

## Dataset

### Dataset description

We are get data from the website Finance.yahoo.com.

### We using dataset of three cryptocurrency include: ETH, BNB and XRP from 20/6/2018 to 20/6/2023 (during 5 years).

Ảnh có chứa văn bản, ảnh chụp màn hình, Phông chữ, số

Mô tả được tạo tự động

1. ETH data representation

Ảnh có chứa văn bản, Phông chữ, số, hàng

Mô tả được tạo tự động

1. BNB data representation

Ảnh có chứa văn bản, Phông chữ, số, ảnh chụp màn hình

Mô tả được tạo tự động

1. XRP data representation

### Description statistics

|  |  |  |  |
| --- | --- | --- | --- |
| Measure | ETH | BNB | XRP |
| Min | 84.31 | 4.5287 | 0.1396 |
| Max | 4812.09 | 675.6841 | 1.8392 |
| Mean | 1238.63 | 178.1599 | 0.475 |
| Count | 1827 | 1827 | 1827 |
| Median | 626.41 | 35.2872 | 0.381 |
| Range | 4727.78 | 671.1554 | 1.67 |
| Variance | 1444095 | 33500.7816 | 0.081 |
| Standard Deviation | 1201.71 | 183.0322 | 0.285 |
| Coefficent of Variation | 0.97 | 1.027348 | 0.601 |
| Skewness | 0.94 | 0.62957 | 1.687 |
| Kurtosis | -0.12 | -0.86894 | 2.723 |

### Histogram Chart

#### ETH

Ảnh có chứa ảnh chụp màn hình, văn bản, biểu đồ, Hình chữ nhật

Mô tả được tạo tự động

#### BNB

A picture containing text, screenshot, diagram, plot

Description automatically generated

#### XRP

Ảnh có chứa văn bản, ảnh chụp màn hình, biểu đồ, Sơ đồ

Mô tả được tạo tự động

### Box Plot Char

#### ETH

Ảnh có chứa biểu đồ, ảnh chụp màn hình, hàng, Hình chữ nhật

Mô tả được tạo tự động

#### BNB

A diagram of a box plot

Description automatically generated with low confidence

#### XRP

Ảnh có chứa văn bản, ảnh chụp màn hình, biểu đồ, thiết kế

Mô tả được tạo tự động

## Tools used

In this test, we use Python language and support tool is Google Colab. Besides, we use Python built-in libraries like Pandas to process data in the form of data frames. Matplotlib plots to visualize data. Fig below illustrate cryptocurrence closeing price data. Numpy helps with math and matrix operations in this experiment. Combine using Keras and Tensorflow library support build, train, evaluate machine learning models .Add finally, the sklearn library supports tools and functions to perform themachine learning and statistics model.

## Split the data

#### ETH

|  |  |  |  |
| --- | --- | --- | --- |
| Model | 7 : 2 : 1 | | |
| RMSE | MAE | MAPE |
| LN | 2084.42 | 2079.87 | 0.5543 |
| ARIMA | 2285.98 | 2275.48 | 0.5753 |
| HW | 67.968 | 4619.615 | 0.734 |
| FTS | 167.1525 | 133.1906 | 0.0774 |
| RNN | 1875.8413 | 1873.6346 | 0.9998 |
| BSTS | 1447.6712 | 1411.0915 | 7.2837 |
| GRU | 1855.403 | 1853.437 | 0.9997 |
| GPR | 1137.587 | 987.089 | 3.3778 |
| FCN | 1650.429 | 1639.302 | 0.9998 |

|  |  |  |  |
| --- | --- | --- | --- |
| Model | 6 : 2 : 2 | | |
| RMSE | MAE | MAPE |
| LN | 620.71 | 584.92 | 0.276 |
| ARIMA | 677.18 | 626.213 | 0.2886 |
| HW | 39.503 | 1560.464 | 0.504 |
| FTS | 143.7591 | 143.7591 | 0.0749 |
| RNN | 1582.6393 | 1561.7394 | 0.9998 |
| BSTS | 1357.3675 | 1325.6962 | 7.6214 |
| GRU | 1597.392 | 1576.7099 | 0.9998 |
| GPR | 1171.07 | 1010.006 | 3.3013 |
| FCN | 1522.477 | 1504.679 | 0.9998 |

|  |  |  |  |
| --- | --- | --- | --- |
| Model | 5 : 3 : 2 | | |
| RMSE | MAE | MAPE |
| LN | 1167.06 | 1140.04 | 2.8163 |
| ARIMA | 2270.49 | 1901.25 | 0.5056 |
| HW | 28.481 | 811.153 | 0.344 |
| FTS | 862.7721 | 823.3577 | 1.1431 |
| RNN | 1571.667 | 1552.0189 | 0.9998 |
| BSTS | 1357.3675 | 1325.7962 | 7.6124 |
| GRU | 1611.92 | 1590.79 | 0.9998 |
| GPR | 1171.0703 | 1010.006 | 3.3013 |
| FCN | 1304.354 | 1292.194 | 0.9998 |

#### BNB

|  |  |  |  |
| --- | --- | --- | --- |
| Model | 7 : 2 : 1 | | |
| RMSE | MAE | MAPE |
| LN | 216.89 | 214.57 | 0.42 |
| ARIMA | 229.425 | 227.596 | 0.4317 |
| HW | 21.092 | 444.89 | 0.597 |
| FTS | 31.8791 | 25.8383 | 0.07 |
| RNN | 318.591 | 317.3136 | 0.9986 |
| BSTS | 291.007 | 289.5353 | 31.803 |
| GRU | 305.948 | 304.69 | 0.9985 |
| GPR | 214.73 | 183.37 | 9.6403 |
| FCN | 301.55 | 300.58 | 0.999 |

|  |  |  |  |
| --- | --- | --- | --- |
| Model | 6 : 2 : 2 | | |
| RMSE | MAE | MAPE |
| LN | 33.61 | 24.96 | 0.08 |
| ARIMA | 56.656 | 48.011 | 0.1426 |
| HW | 18.093 | 327.35 | 0.53 |
| FTS | 29.3082 | 23.8926 | 0.07778 |
| RNN | 313.1188 | 311.827 | 0.9986 |
| BSTS | 276.9725 | 275.2622 | 26.7112 |
| GRU | 305.824 | 304.618 | 0.9986 |
| GPR | 204.929 | 175.233 | 9.2754 |
| FCN | 285.845 | 284.628 | 0.999 |

|  |  |  |  |
| --- | --- | --- | --- |
| Model | 5 : 3 : 2 | | |
| RMSE | MAE | MAPE |
| LN | 253.6 | 251.83 | 6.8 |
| ARIMA | 259.757 | 257.945 | 8.337 |
| HW | 15.685 | 246.027 | 5.738 |
| FTS | 248.0835 | 246.18 | 5.7658 |
| RNN | 287.5423 | 286.4359 | 0.9985 |
| BSTS | 276.9725 | 275.2622 | 26.7112 |
| GRU | 293.095 | 291.885 | 0.99851 |
| GPR | 204.929 | 175.233 | 9.2754 |
| FCN | 148.336 | 147.876 | 0.997 |

#### XRP

|  |  |  |  |
| --- | --- | --- | --- |
| Model | 7 : 2 : 1 | | |
| RMSE | MAE | MAPE |
| LN | 0.571 | 0.569 | 0.57 |
| ARIMA | 0.36 | 0.355 | 0.4534 |
| HW | 0.705 | 0.497 | 0.536 |
| FTS | 0.116 | 0.0976 | 0.1877 |
| RNN | 0.2791 | 0.2782 | 0.5776 |
| BSTS | 0.0982 | 0.0808 | 0.2267 |
| GRU | 0.30834 | 0.30577 | 0.60127 |
| GPR | 0.26122 | 0.17577 | 0.37435 |
| FCN | 0.255 | 0.254 | 0.596 |

|  |  |  |  |
| --- | --- | --- | --- |
| Model | 6 : 2 : 2 | | |
| RMSE | MAE | MAPE |
| LN | 0.249 | 0.244 | 0.373 |
| ARIMA | 0.4904 | 0.4866 | 0.5435 |
| HW | 0.682 | 0.465 | 0.531 |
| FTS | 0.1061 | 0.0822 | 0.1603 |
| RNN | 0.2611 | 0.2594 | 0.605 |
| BSTS | 0.0995 | 0.0842 | 0.2453 |
| GRU | 0.28041 | 0.2723 | 0.6171 |
| GPR | 0.31416 | 0.21463 | 0.4586 |
| FCN | 0.242 | 0.24 | 0.6 |

|  |  |  |  |
| --- | --- | --- | --- |
| Model | 5 : 3 : 2 | | |
| RMSE | MAE | MAPE |
| LN | 0.336 | 0.327 | 4.344 |
| ARIMA | 0.1198 | 0.1047 | 0.2048 |
| HW | 0.459 | 0.211 | 0.34 |
| FTS | 0.0319 | 0.0275 | 0.0627 |
| RNN | 0.252 | 0.2506 | 0.597 |
| BSTS | 0.0995 | 0.0842 | 0.2453 |
| GRU | 0.265 | 0.257 | 0.6035 |
| GPR | 0.3142 | 0.2146 | 0.4586 |
| FCN | 0.25 | 0.248 | 0.61 |

## Evaluation models

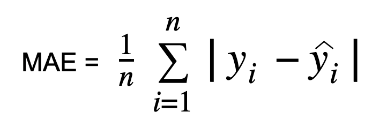
In this research, predictive models are evaluated according to three criteria: MAE, MAPE, and RMSE.

### Mean Absolute Percentage Error (MAPE)

Ảnh có chứa Phông chữ, màu trắng, biểu đồ, thuật in máy

Mô tả được tạo tự động

### Mean absolute errorn (MAE)



### Root Mean Squared Error

Ảnh có chứa Phông chữ, văn bản, màu trắng, biểu đồ

Mô tả được tạo tự động

Where:

* n: is sample size of dataset.
* yi : is the actual value at time t.
* yi : is the mean value at time t.



* yi : is the predicted value of time t.



## Result

Based on the above evaluation results, we can see that for each dataset there will be its own model and appropriate split data ratio. For the best predictive performance, we predict the "close" price of each data set over the next 30 days based on the selected models and split. The selected models will have the MAE, MAPE, RMSE min

### ETH

For ETH dataset: two best model is FTS and FCN.

* FTS with the splitting ratio of Train/Test/Validate as 7/2/1, MAE value as 133.1906, RMSE as 167.1525 and MAPE as 0.0774.
* FCN with the splitting ratio of Train/Test/Validate as 6/2/2, RMSE value as 1522.477, MAE as 1504.679 and MAPE as 0.9998.

So, we will use these 2 models to forecast “Close” price the next 30 days.

### BNB

For BNB dataset: two best model is .

* FTS with the splitting ratio of Train/Test/Validate as 7/2/1, MAE value as 28.8333, RMSE as 318791 and MAPE as 0.07.
* FCN with the splitting ratio of Train/Test/Validate as 6/2/2, RMSE value as 301.55, MAE as 300.58 and MAPE as 0.999.

So, we will use these 2 models to forecast “Close” price the next 30 days.

### XRP

For XRP dataset: two best model is FTS and GRU .

* FTS with the splitting ratio of Train/Test/Validate as 6/2/2, MAE value as 0.0822, RMSE as 0.1061 and MAPE as 0.1603
* GRU with the splitting ratio of Train/Test/Validate as 5/3/2, RMSE value as 0.265, MAE as 0.257 and MAPE as 0.6035

So, we will use these 2 models to forecast “Close” price the next 30 days.

## Visualization

Ảnh có chứa văn bản, ảnh chụp màn hình, biểu đồ, Sơ đồ

Mô tả được tạo tự động

1. The result of FTS model with data split into the ratio of 7/2/1

Ảnh có chứa ảnh chụp màn hình, văn bản, Sơ đồ, biểu đồ

Mô tả được tạo tự động

1. The result of FCN model with data split into the ratio of 6/2/2

A picture containing text, screenshot, plot, diagram

Description automatically generated

1. The result of FTS model with data split into the ratio of 6/2/2

A picture containing text, screenshot, plot, font

Description automatically generated

1. The result of GRU model with data split into the ratio of 5/3/2

# Conclusion

During the implementation of the research project, besides the obtained results, we encountered some difficulties and challenges, which are:

* ***Difficulty in building predictive models***: Cryptocurrency prediction models are often complex and require in-depth knowledge of the field. We examined a wide range of literature including scientific articles and made important decisions in the selection and construction of predictive models.
* ***Difficulty in evaluating model effectiveness:*** To evaluate the effectiveness of prediction models, we used several different evaluation metrics, but the results showed that the accuracy of the models was still not high.

In the future, we will try to address the above challenges and provide better solutions for predicting cryptocurrentcy prices by:

* Improving skills in selecting and processing data: We will continue to research and apply the most advanced methods in selecting and processing data to ensure the feasibility and accuracy of prediction models.
* Combine models: We will continue to research and combine models together to enhance performance.
* Enhancing cooperation and sharing experience: We will continue to search document, research works in academic forums and scientific journals.

With the above solutions, we believe that we can improve the effectiveness and accuracy of cryptocurrentcy price prediction models in the future

# Acknowledgements

First of all, we would like to express our sincere gratitude to Assoc. GS. Dr. Nguyen Dinh Thuan and TA Nguyen Minh Nhut, Nguyen Thi Viet Huong provided us with the necessary expertise to complete this project. With a spirit of dedication and high responsibility, the teachers helped us overcome difficulties and challenges and successfully completed our project research.

In addition, we would also like to express our gratitude to the teachers for their important support in the editing, evaluation and approval of our project map.

However, with limited time, knowledge and experience, shortcomings are inevitable, so the team is looking forward to receiving valuable suggestions from you to help. Your comments, corrections and suggestions for improvement have helped us to understand our mistakes and perform our projects better.

We believe that the knowledge and experience that you have imparted will help us develop our careers and become useful people for society. Once again, we would like to sincerely thank you for all the good things you have brought to us during the past time.

# References

|  |  |
| --- | --- |
| [1] | M. Poongodi, Ashutosh Sharma, V. Vijayakumar, Vaibhav Bhardwaj, Abhinav Parkash Sharma, Razi Iqbal and Rajiv Kumar,, "Prediction of the price of Ethereum blockchain cryptocurrency in an industrial finance system," in *International journal Computers & Electrical Engineering*, 2020. |
| [2] | Siti Saadah, A.A Ahmad Whafa, "Monitoring Financial Stability Based on Prediction of Cryptocurrencies Price Using Intelligent Algorithm," in *2020 International Conference on Data Science and Its Applications (ICoDSA)*, Bandung, Indonesia, 2020. |
| [3] | V. Derbentseva, V. Babenko, K. Khrustalev, H. Obruch, S. Khrustalova, "Comparative Performance of Machine Learning Ensemble Algorithms for Forecasting Cryptocurrency Prices," *International Journal of Engineering,* vol. 34, no. 01, pp. 140-148, 2021. |
| [4] | Kiki Ramadani, Dodi Devianto, "The forecasting model of Bitcoin price with fuzzy time series Markov chain and chen logical method," *NTERNATIONAL CONFERENCE ON SCIENCE AND APPLIED SCIENCE (ICSAS2020),* vol. 2296, no. 1, 2020. |
| [5] | Mohammad J. Hamayel, Amani Yousef Owda, "A Novel Cryptocurrency Price Prediction Model Using GRU, LSTM and bi-LSTM Machine Learning Algorithms," vol. 2, no. 4, pp. 477-496, 2021. |
| [6] | Dian Utami Sutiksno, Ansari Saleh Ahmar, Nuning Kurniasih, Eko Susanto, Audrey Leiwakabessy, "Forecasting Historical Data of Bitcoin using ARIMA and α-Sutte Indicator," in *2nd International Conference on Statistics, Mathematics, Teaching, and Research*, 2018. |
| [7] | Qiang Song, Brad S. Chissom, "Forecasting Enrollments with Fuzzy Time Series part 1," in *The Anuual Meeting of the Mid-South Educational Reseach Association,*, Lexington, KY, 1991. |
| [8] | S.-M. Chen, "“Forecasting enrollments based on fuzzy time series,”," *Fuzzy Sets and System,* no. 81, pp. 311-319, 1996. |
| [9] | Jizhong Wu, Bo Liu, Hao Zhang ,Shumei He, Qianqian Yang, "Fault Detection Based on Fully Convolutional Networks (FCN)," *Journal of Marine Science and Engineering,* vol. 9, no. 3, p. 259, 2021. |
| [10] | Simplilearn, "Recurrent Neural Network(RNN) Tutorial: Types, Examples, LSTM and More," [Online]. Available: https://www.simplilearn.com/tutorials/deep-learning-tutorial/rnn. [Accessed 10 06 2023]. |
| [11] | Wikipedia, "Exponent Smoothing," [Online]. Available: https://en.wikipedia.org/wiki/Exponential\_smoothing. [Accessed 10 06 2023]. |
| [12] | SAP, "Triple Exponent Smoothing," [Online]. Available: https://bom.so/2zt4EL. [Accessed 10 06 2023]. |

[13] “Autoregressive Integrated Moving Average (ARIMA) Prediction Model,” *Investopedia*. https://www.investopedia.com/terms/a/autoregressive-integrated-moving-average-arima.asp (accessed Jun. 21, 2023).

[14] “Gated recurrent unit,” *Wikipedia*. Jun. 04, 2023. Accessed: Jun. 21, 2023. [Online]. Available: https://en.wikipedia.org/w/index.php?title=Gated\_recurrent\_unit&oldid=1158552114#cite\_note-Ravalli2018-4

[15] “10.2. Gated Recurrent Units (GRU) — Dive into Deep Learning 1.0.0-beta0 documentation.” https://d2l.ai/chapter\_recurrent-modern/gru.html (accessed Jun. 21, 2023).

[16] C. E. Rasmussen and C. K. Williams, “Gaussian processesfor machine learning. 2006,” The MIT Press, Cambridge,MA, USA, vol. 38, pp. 715–719, 2006.

[17] “(PDF) Dự đoán xu thế chỉ số chứng khoán Việt Nam VN-Index sử dụng phân tích hồi quy Gaussian Process và mô hình tự hồi quy trung bình động ARMA.” https://www.researchgate.net/publication/329123511\_Du\_doan\_xu\_the\_chi\_so\_chung\_khoan\_Viet\_Nam\_VN-Index\_su\_dung\_phan\_tich\_hoi\_quy\_Gaussian\_Process\_va\_mo\_hinh\_tu\_hoi\_quy\_trung\_binh\_dong\_ARMA (accessed Jun. 21, 2023).