electronics

Article

A Deep Learning-Based Action Recommendation Model for Cryptocurrency Profit Maximization

Jaehyun Park and Yeong-Seok Seo *

Department of Computer Engineering, Yeungnam University, Gyeongsan 38541, Korea; kat3160@yu.ac.kr
* Correspondence: ysseo@yu.ac.kr; Tel.: +82-53-810-3534

Abstract: Research on the prediction of cryptocurrency prices has been actively conducted, as cryptocurrencies have attracted considerable attention. Recently, researchers have aimed to improve the performance of price prediction methods by applying deep learning-based models. However, most studies have focused on predicting cryptocurrency prices for the following day. Therefore, clients are inconvenienced by the necessity of rapidly making complex decisions on actions that support maximizing their profit, such as "Sell", "Buy", and "Wait". Furthermore, very few studies have explored the use of deep learning models to make recommendations for these actions, and the performance of such models remains low. Therefore, to solve these problems, we propose a deep learning model and three input features: sellProfit, buyProfit, and maxProfit. Through these concepts, clients are provided with criteria on which action would be most beneficial at a given current time. These criteria can be used as decision-making indices to facilitate profit maximization. To verify the effectiveness of the proposed method, daily price data of six representative cryptocurrencies were used to conduct an experiment. The results confirm that the proposed model showed approximately 13% to 21% improvement over existing methods and is statistically significant.

Keywords: cryptocurrency; Bitcoin; Bitcoin price prediction; deep learning; input feature; decision making; profit

check for updates

Citation: Park, J.; Seo, Y.-S. A Deep Learning-Based Action Recommendation Model for Cryptocurrency Profit Maximization. *Electronics* **2022**, *11*, 1466. https://doi.org/10.3390/electronics11091466

Academic Editor: George A. Papakostas

Received: 6 April 2022
Accepted: 29 April 2022
Published: 3 May 2022

1. Introduction

Since Satoshi Nakamoto first introduced the cryptocurrency known as Bitcoin in 2008, numerous altcoins, such as Ethereum have emerged [1–3]. Presently, cryptocurrency continues to impact global financial markets and has become relatively ubiquitous in everyday life. Cryptocurrencies have become popular speculative investments compared to stocks, and they are also used for routine purchases of everyday products. Furthermore, blockchain technology has attracted considerable attention for the prevention of forgery and falsification via decentralized ledgers. Blockchain technology has been applied in various fields, including the Internet of Things (IoT), owing to its high security and ease of management [4–8].

Moreover, the cryptocurrency market can be integrated with various new technologies such as non-fungible tokens (NFT) or the metaverse, which have recently attracted attention. Hence, these associations have further increased the value of cryptocurrency and their potential for more widespread adoption [9,10]. Moreover, non-face-to-face services, which minimize contact between people, are also increasing in popularity. As a result, NFT auctions of digital goods representing the ownership of tokens referencing links to artworks or digital assets are being actively conducted as non-face-to-face services. After an auction, transactions are conducted in cryptocurrency to acquire ownership of the tokens. Other non-face-to-face service-based transactions in metaverse systems are also conducted using cryptocurrency. In this manner, the existing cryptocurrency market is expanding beyond speculative investment—a limited scope—and is being integrated into the foundation of various emerging technologies. This trend is expected to accelerate along with associated research and development efforts.

Despite considerable controversy regarding the recognition of cryptocurrency assets, cryptocurrencies are actively traded on large exchanges with many investors in roughly the same manner as stocks. Investors analyze the trading market to predict cryptocurrency prices. However, cryptocurrency has a short history compared to conventional stocks, so research on techniques for the analysis and prediction of cryptocurrency prices has thus far remained in its relatively early stages. Therefore, recently, studies have been actively conducted to compensate for this gap in the literature [11–14]. Cryptocurrency price prediction can be helpful from a maintenance aspect, such as maintaining blockchain implementations or expanding blockchain networks. Moreover, cryptocurrency price prediction can be used as auxiliary indices for transactions and market price adjustments in NFT and metaverse markets.

Representative works include analyses of various training data from diverse perspectives by applying deep learning, as well as research on the prediction of cryptocurrency prices using long short-term memory (LSTM) and gated recurrent unit (GRU) models, which have exhibited excellent performance on time series data that solved the slow learning rate and "vanishing gradient" problems of recurrent neural network (RNN) models [11–15]. However, recently, cryptocurrency prices have fluctuated drastically to the extent that it has become impossible or impractical to predict prices over time. For this reason, cryptocurrency prices may be considered to exhibit the characteristics of increasing volatility and fluctuations. Furthermore, it is generally difficult to predict cryptocurrency prices, and the prediction performance may be degraded considerably depending on circumstances [16]. Over the past several years, various ideas have been proposed to solve these problems, and they have demonstrated effectiveness in real predictions. These studies have utilized hybrid models, sentiment analysis to reflect human psychological states, and frequency decomposition to minimize volatility and fluctuation [17–27]. However, these studies involve some limitations. Accurate predictions may be important for cryptocurrencies, but the perspective of ultimately maximizing profit should also be considered in terms of situations involving extreme volatility and fluctuations [28]. To maximize profits, providing clients with the means to make decisions—such as at what particular points they should sell, buy, or hold cryptocurrency assets—is much more useful in terms of client convenience and support than accurate price prediction alone. In this paper, to overcome these limitations, we propose new input features based on existing input features used in previous studies.

Therefore, in this study, we propose a deep learning-based price prediction and action recommendation model designed to recommend actions to support maximizing profits in addition to simply predicting near-term prices for the convenience of cryptocurrency clients and to facilitate their decision-making. To this end, we defined features sellProfit as the profit that can be obtained by selling a cryptocurrency at a given time, buyProfit, as the profit that can be obtained by buying a cryptocurrency at a given time, and maxProfit, which is the maximum profit that can be obtained at a given time. Next, we conducted an experiment to compare performance when sellProfit, buyProfit, maxProfit—the three input features we defined—were used with the performance when these features were not used. A total of six representative cryptocurrencies were used in this study, including Bitcoin (BTC), Cardano (ADA), Dash (DASH), Ethereum (ETH), Litecoin (LTC), and Monero (XMR) [29–34].

BTC is the first and most famous cryptocurrency. Therefore, it is receiving the most attention from the public [29]. ADA was developed using a protocol called Ouroboros and a programming language called Haskell [30]. Therefore, it is characterized by being designed very safely from hacking. DASH focused on speed. Therefore, most transactions can be processed within one second, and immediate payment is possible [31]. ETH can be applied to various fields. Therefore, it is being developed based on the ETH platform in IoT, and other fields are also paying attention to ETH [32]. LTC also focused on speed. It has improved the speed of transactions by adopting a Lightning network that only uploads final results after synthesizing all transactions [33]. XMR applied a CryptoNote protocol to ensure the safest anonymity. Therefore, it is impossible to track receiving addresses and transaction history [34].

For an objective experimental comparison, accuracy and F1 score were used as evaluation metrics. In addition, a *t*-test was also used to perform a systematic verification based on statistics.

Through this proposed model, clients are provided with criteria to determine whether they would benefit from selling, buying, or holding ("waiting") cryptocurrency assets. Hence, they can use these criteria as decision-making indices that support profit maximization. Furthermore, in the context of NFTs and metaverse systems, these criteria can be used as auxiliary indices from a perspective of transactions conducted in internal markets and the adjustment of market prices.

In general, profit in cryptocurrency refers to a long-term profit calculated by comparing the price at the time the cryptocurrency was purchased, the current price, and the future price. However, short-term profits are also considered, which leads to market patterns resembling day trading. Such profits do not reflect the price at the time the cryptocurrency was purchased. Instead, such profits are calculated based on the best choice that can be made at a given time. For example, on the one hand, from a short-term profit perspective, it may be profitable to sell cryptocurrency before the price falls further if it is predicted to decrease. On the other hand, it may be profitable to buy the cryptocurrency before the price goes up further if the cryptocurrency price is predicted to increase. Therefore, we did not focus on long-term profits in this study. Instead, we focused on short-term profits, which do not take into consideration the price at the time the cryptocurrency was purchased. Thus, we propose a deep learning-based action recommendation model that supports maximizing profits.

The contributions of this study are summarized as follows:

- In contrast to previous studies on the prediction of cryptocurrency prices, we proposed an action recommendation technique with new input features (sellProfit, buyProfit, and maxProfit) to improve the profits of cryptocurrency clients. The proposed approach can provide convenience to cryptocurrency clients and help them decide on actions to be taken at a given time to maximize profit;
- The technique proposed in this study was compared with existing methods and analyzed in detail through a statistical verification method to verify the performance of the proposed technique;
- More practical results were provided by analyzing the latest available data collected from real cryptocurrency markets instead of artificial data;
- Cryptocurrency price prediction and action recommendation can be used as auxiliary indices when performing tasks such as cryptocurrency-based transactions and market price adjustments within new technology areas such as NFTs and metaverse systems.

The remainder of this study is organized as follows. Section 2 describes related research on deep learning models designed to recommend actions that support profit maximization and improve price prediction performance using a deep learning model. Section 3 explains our overall approach. Then, Section 4 describes the experimental design along with the results. Lastly, Section 5 summarizes the results of this study and describes our conclusions and future research.

2. Related Work

In this section, we introduce previous studies on deep learning models designed to predict the future prices of cryptocurrencies and models that recommend an action to support profit maximization. Cryptocurrency price prediction is a considerably difficult and challenging field in time series data research. The characteristics of high volatility and price fluctuations increase the difficulty of price prediction [16]. Numerous studies have been proposed to solve these problems [17–27].

2.1. Hybrid Model

Hybrid models are designed to predict prices by using two or more different predictive models together. They are frequently used in various fields owing to their combined advantages to compensate for the shortcomings of each model.

Livieris et al. proposed a hybrid model which combined a convolution neural network (CNN) model and an LSTM architecture, which has shown high performance on time series data. They used 1399 days of Bitcoin, Ethereum, and Ripple price data from 1 January 2017 to 31 October 2020. In addition, they proposed a method to predict prices using the attributes and characteristics of data from CNN's pooling layer as input to an LSTM layer. This method was shown to be not only less expensive but also to exhibit higher predictive performance than existing time series models [17].

Patel et al. proposed a hybrid model based on LSTM and GRU models, which demonstrated high performance on time series data. They used 1279 days of Litecoin price data from 24 August 2016 to 23 February 2020 and 1851 days of Monero data from 30 January 2015 to 23 February 2020. They proposed a method of predicting prices by concatenating predicted prices in two models using the same data. This method exhibited predictive performance higher than that of an existing LSTM model [18].

Koo et al. proposed a hybrid model based on LSTM and Generalized Auto Regressive Conditional Heteroskedasticity (GARCH) model. GARCH model is used for analyzing time-series data where the variance error is believed to be serially autocorrelated. They used 6179 days of S&P 500 price data from 1 January 2004 to 30 November 2020. They proposed a method of predicting prices by concatenating predicted prices in two models using the same data. This method exhibited predictive performance higher than that of an existing LSTM model [19].

2.2. Sentiment Analysis

Sentiment analysis is a method of analyzing the extent to which the psychological state of clients affects future outcomes. Data on psychological states can be extracted, for instance, through Twitter, online communities, and news headlines. It is used in various fields because clients' perspectives on specific targets can affect predictive performance.

Valencia et al. proposed a model that used tweets on Twitter as input features. They used Bitcoin, Ethereum, Litecoin, and Ripple data for 1440 h from 16 February 2018 to 21 April 2018. They collected Twitter tweets every hour and determined which cryptocurrency was associated with the most positive or negative tweets. Next, they proposed a method using the results as input features to predict whether the price of these currencies would rise or fall. This method proved to be more predictive than when Twitter's tweets were not used [20].

Maqsood et al. similarly proposed a model that used tweets on Twitter as input features. They used five years of Apple, Citigroup, Google, and Microsoft stock data from 2012 to 2016. In contrast to previous studies, their proposed method predicted neutral tweets as well as positive and negative tweets as new output. This method exhibited higher predictive performance than when only positive and negative tweets were used [21].

Aasi et al. proposed a model that used news headlines, Google Trends data, and tweets on Twitter as input features. They also proposed Multivariate and Multi-frequency LSTM (MMLSTM). MMLSTM extracts information from time series data. They used 4333 days of Apple stock data from 1 January 2009 to 11 November 2020. This method exhibited higher predictive performance than when only tweets were used [22].

2.3. Frequency Decomposition

Frequency deposition algorithms address the issues of high volatility and fluctuation. These algorithms reduce the volatility of data and the magnitude of fluctuations, resulting in higher predictive performance. Because of these characteristics, they are used in various fields.

Xuan et al. performed preprocessing of price-related data that reduced volatility and fluctuation using the empirical mode deposition (EMD) algorithm. The EMD algorithm extracts information from high-frequency to low-frequency components of the graph data. They used price data on the Chinese stock market recorded over five years, from 2015 to 2019. This method proved that the addition of frequency components to existing input features showed higher stock prices and predictive performance [23].

Hadi et al. used the complementary ensemble empirical mode decomposition (CEEMD) algorithm. This algorithm solved the problem of "Mode Mixing" of the EMD algorithm [24]. They used stock price data from the S&P 500 as well as the DAX, Dow Jones, and Nikkei 225 from January 2010 to September 2019. Their results demonstrated that the use of the CEEMD algorithm showed higher predictive performance than the EMD algorithm [25].

2.4. Action Recommendation

Action recommendation is a method of recommending actions to support profit maximization. However, compared to studies on price prediction, relatively few methods have been developed, and their recommendation performance is low.

Nelson et al. experimented with action recommendation models for two classes, including "Sell" and "Buy". They experimented with four models, including LSTM, a multi-layer perceptron (MLP), a random forest (RF), and a pseudo-random model. They used Brazilian stock data from 2008 to 2015. In this experiment, they demonstrated that LSTM showed the highest recommendation performance and an average accuracy of 0.5590 [26].

Sanboon et al. experimented with action recommendation models for two classes, including "Sell" and "Buy". They experimented with seven models, including LSTM, RF, MLP, Support Vector Machine (SVM), Logistic Regression, Decision tree, and K-Nearest Neighbor (KNN). They used Thailand stocks data from 2015 to 2017. They also used market data excluding trading volume as input features. They demonstrated that LSTM showed the highest performance compared to other models [27].

A summary of related works can be found in Table 1. As may be observed from related works, many studies have considered the development of methods to improve the performance price prediction methods. However, relatively few studies have been conducted on recommending action to support profit maximization. Therefore, in this study, we focus on improving the recommendation performance of a deep learning model designed to recommend action to support profit maximization through daily price-related data on cryptocurrencies.

Table 1. Summary of related studies.

Category	Author	Year	Data Set	Model	Input Features
Hybrid Model	Livieris et al. [17]	2021	BTC, ETH, Ripple	CNN-LSTM model	Market data *
	Patel et al. [18]	2020	LTC, XMR,	LSTM-GRU model	Market data
	Koo et al. [19]	2022	S&P 500	GARCH-LSTM model	Financial indices **
Sentiment Analysis	Valencia et al. [20]	2019	BTC, ETH, LTC, Ripple	MLP, SVM, RF	Market data, tweets on Twitter
	Maqsood et al. [21]	2020	Apple, Citigroup, etc.	Linear Regression, Support Vector Regression, Deep Learning	Market data, tweets on Twitter
	Aasi et al. [22]	2021	Apple	MMLSTM	Market data, tweets on Twitter, news headlines, Google Trends data
Frequency Decomposition	Xuan et al. [23]	2020	Chinese stock	EMD-LSTM-CSI model	Market data, EMD results
	Hadi et al. [25]	2021	S&P 500, DAX, etc.	CNN-LSTM model	Market data, EMD results, CEEMD results

Table 1. *Cont.*

Category	Author	Year	Data Set	Model	Input Features
Action Recommendation	Nelson et al. [26]	2017	Brazilian stock	LSTM, MLP, RF, etc.	Market data
	Sanboon et al. [27]	2019	Thailand stock	LSTM, SVM, Logistic Regression, etc.	Market data excluding trading volume

* Market data: Average price, Open price, High price, Low price, Trading volume, etc. ** Financial indices: Exchange rate, Price, Stock index, Term yield, etc.

3. Approach

Currently, cryptocurrencies are attracting a lot of attention. Cryptocurrencies have become new investments, and numerous platforms based on cryptocurrency are being developed. Therefore, cryptocurrency prediction research papers are important in terms of profitability and platform maintenance. However, most cryptocurrency prediction studies focus only on price prediction. Therefore, clients are inconvenienced by the necessity of rapidly making complex decisions on actions that support maximizing their profit. Clients have to sell cryptocurrency when the price of cryptocurrency rises, and they have to purchase it when the price fall. Furthermore, very few studies have explored the use of deep learning models to make recommendations for these actions, and the performance of such models remains low. In addition, there is a lack of research to newly calculate the input features. Therefore, in this paper, we propose new input features and a deep learning model to solve these problems.

Figure 1 shows the overall approach of the proposed method. The output of our proposed method is calculated based on cryptocurrency price data. First, we calculate new input features such as sellProfit, buyProfit, and maxProfit through the collected data and the proposed equation to train the deep learning model. Classification groups are limited to the most common functions: "Sell", "Buy", and "Wait". We classify groups based on calculated sellProfit, buyProfit, maxProfit, and our proposed criteria. These classes then become the final output in the deep learning model. Next, we trained and tested the deep learning model through collected data, proposed input features, and proposed classes. Through the evaluation criteria used in the experiment, we compared existing methods using the LSTM model with the proposed method [26,27]. Finally, we evaluated the performance of these methods in terms of these evaluation criteria.

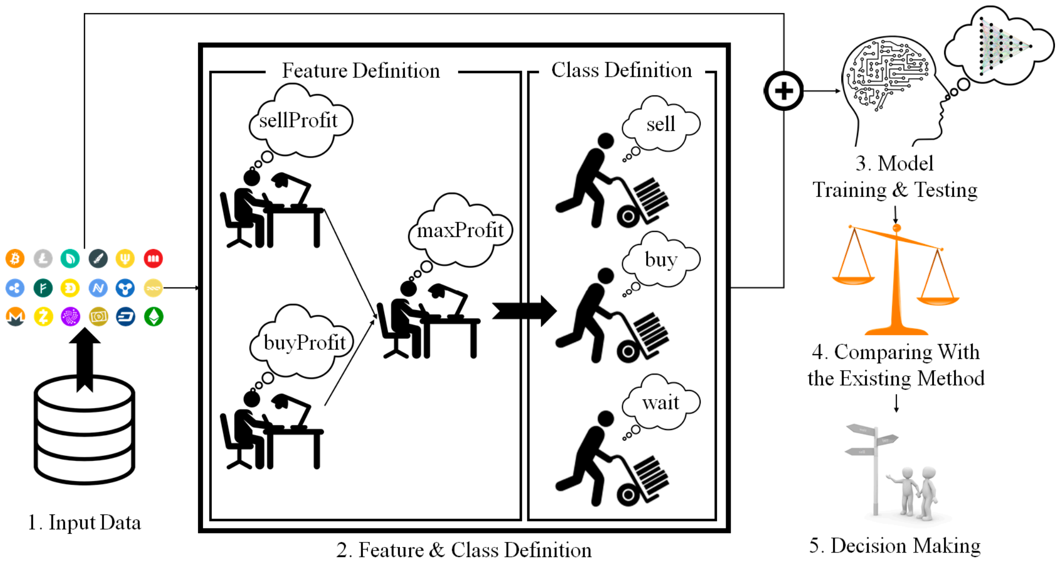


Figure 1. Overall approach of the proposed method.

3.1. Feature Definition

In this section, we describe the three proposed input features: sellProfit, buyProfit, and maxProfit. For easy understanding, we described them based on price graphs, equations, and real prices.

3.1.1. sellProfit

The feature sellProfit represents the profit that can be obtained when selling a specific amount of cryptocurrency. Figure 2 shows the daily price of Bitcoin over 88 days between 1 April 2021 and 30 June 2021. This graph shows that Bitcoin's daily price continues to fall in the long run. According to a study by Ji et al., clients who trade stocks tend to sell if the price is predicted to continue to fall in the long run, as shown in Figure 2 [35]. Based on the results of this study, we propose Equation (1). The result of this equation is defined as sellProfit.

$$\text{sellProfit} = \text{M} \times (\text{todayPrice} - \text{tomorrowPrice}) \tag{1}$$

Figure 2. Graph of falling Bitcoin price per day.

We assume a client sells M units of cryptocurrency today when tomorrow's price (tomorrowPrice) falls compared to today's (todayPrice). Because we sell when the price is high, the system turns a profit by multiplying the price difference (todayPrice − tomorrowPrice) by M units of cryptocurrency sold. On the contrary, if tomorrowPrice rises compared to todayPrice, we sell when the price is low. Thus, we lose profit by multiplying the price difference (todayPrice − tomorrowPrice) by M units of cryptocurrency sold.

For example, in Figure 2, the price of Bitcoin on 17 April 2021 is about $60,042. However, the price of Bitcoin on 18 April 2021 is about $56,207. This is a price down $3805 from the previous day. If we sell one cryptocurrency before the price falls, we can earn about $3805 in profit.

3.1.2. buyProfit

The feature buyProfit is defined as profit that can be obtained when buying a specific amount of cryptocurrency. Figure 3 is a Bitcoin daily price graph for 181 days between 1 September 2020 and 28 February 2021. This graph shows that Bitcoin's daily price continued to fall in the long run. According to a study by Ji et al., clients who traded stocks tended to sell their stocks if the price was predicted to continue to rise in the long run, as shown in Figure 3 [35]. Based on the results of this study, we present Equation (2). The result of this equation is defined as buyProfit.

$$\text{buyProfit} = \text{N} \times (\text{tomorrowPrice} - \text{todayPrice}) \tag{2}$$

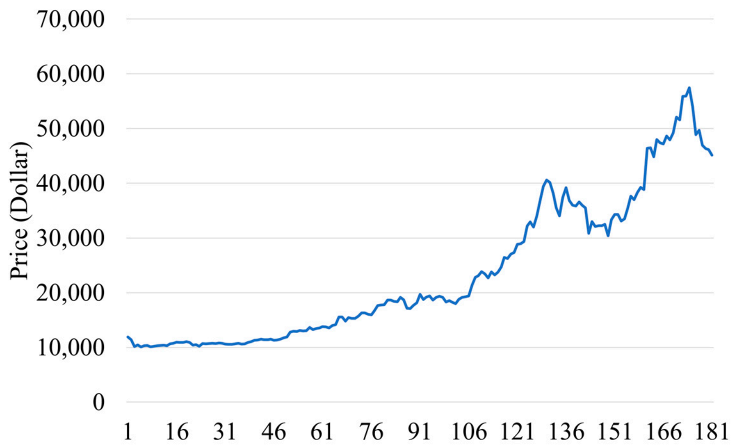


Figure 3. Graph of rising Bitcoin price per day.

We assume that we purchase N units of cryptocurrency today when tomorrow's price (tomorrowPrice) rises compared to today's (todayPrice). Because we buy the units of currency when the price is low, we make a profit by multiplying the price difference (tomorrowPrice − todayPrice) by N quantities of cryptocurrency purchased. On the contrary, if tomorrowPrice falls compared to todayPrice, we purchase units when the price is high. Thus, we lose profit by multiplying the price difference (tomorrowPrice − todayPrice) by N quantities of cryptocurrency purchased.

For example, in Figure 3, the price of Bitcoin on 4 November 2020 is about $14,146. However, the price of Bitcoin on 5 November 2020 is about $15,587. This is a price increase of $1441 from the previous day. If we purchase one cryptocurrency before the price rises, we can earn about $1441 in profit.

3.1.3. maxProfit

The feature maxProfit represents the maximum profit that can be obtained between sellProfit and buyProfit. Figure 4 shows the flow chart for calculating maxProfit. According to a study by Ji et al., clients who trade stocks tend to take specific actions after comparing the profits earned when purchasing stocks with the profits earned from selling [35]. Based on the results of this study, we propose a flowchart to obtain maxProfit. If sellProfit is greater than buyProfit, maxProfit is assigned the same value as sellProfit. This means that profits can be maximized when we sell. On the contrary, if buyProfit is greater than sellProfit, maxProfit is assigned the same value as buyProfit. This implies that profits can be maximized when units of cryptocurrency are purchased. Brief definitions of the three input features defined in this section are shown in Table 2.

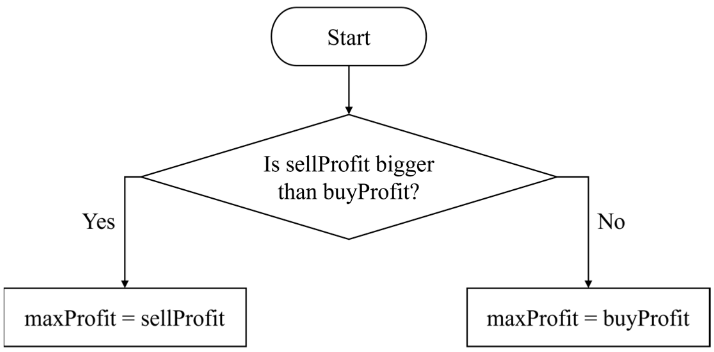


Figure 4. Flowchart of maxProfit definition.

Table 2. Definition of sellProfit, buyProfit, and maxProfit.

Input Feature	Definition
sellProfit	Profit that can be earned when selling a certain amount of cryptocurrency.
buyProfit	Profit that can be earned when buying a certain amount of cryptocurrency.
maxProfit	Maximum profit that can be earned between sellProfit and buyProfit.

For example, in Figure 2, the price of Bitcoin on 17 April 2021 is about $60,042. However, the price of Bitcoin on 18 April 2021 is about $56,207. We assume that we will sell or purchase one cryptocurrency. Assuming that the current date is 17 April 2021, if sellProfit is calculated through Equation (1), the value is 3805. However, if buyProfit is calculated through Equation (2), the value is −3835. Consequently, maxProfit has the same value as sellProfit because sellProfit is larger than buyProfit. Therefore, we can maximize profits when we sell cryptocurrency rather than when we purchase it.

On the contrary, in Figure 3, the price of Bitcoin on 4 November 2020 is about $14,146. However, the price of Bitcoin on 5 November 2020 is about $15,587. We assume that we will sell or purchase one cryptocurrency. Assuming that the current date is 4 November 2020, if sellProfit is calculated through Equation (1), the value is −1441. However, if buyProfit is calculated through Equation (2), the value is 1441. Consequently, maxProfit has the same value as buyProfit because buyProfit is larger than sellProfit. Therefore, we can maximize profits when we purchase cryptocurrency rather than when we sell it.

3.2. Class Definition

Figure 5 shows the class classification criteria as flowchart. Three classes are used to perform classification, including "Sell", "Buy", and "Wait". Classes were defined based on sellProfit, buyProfit, and maxProfit. In the case of the "Wait" class, some of the clients act when they can earn more than a given profit threshold, and if not, they can wait another day. Therefore, the "Wait" class means that maxProfit is less than a specific value and does not earn as much profit as desired.

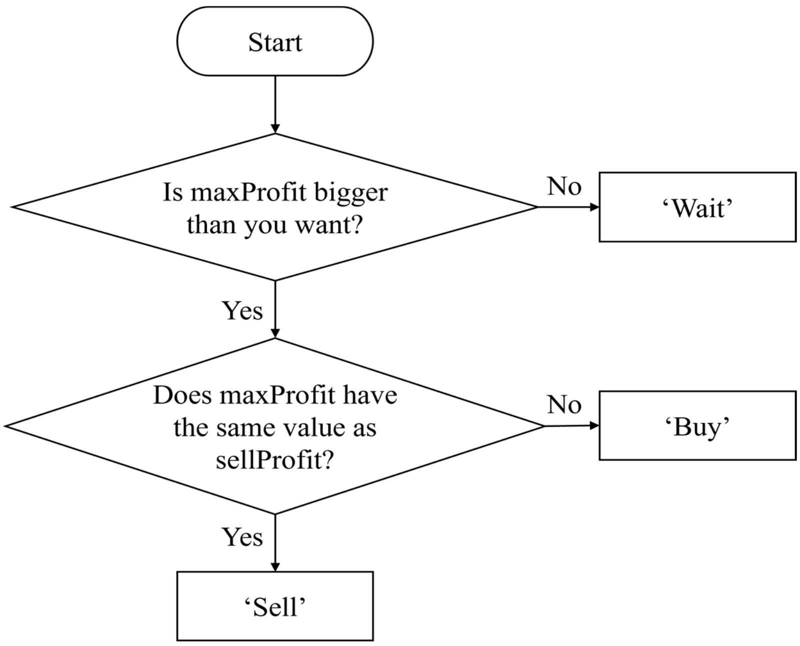


Figure 5. Flowchart of the class definition.

4. Experiment

This section describes the experimental design for conducting our experiment and the results of the experiment performed. Detailed descriptions are given at the head of each subsection.

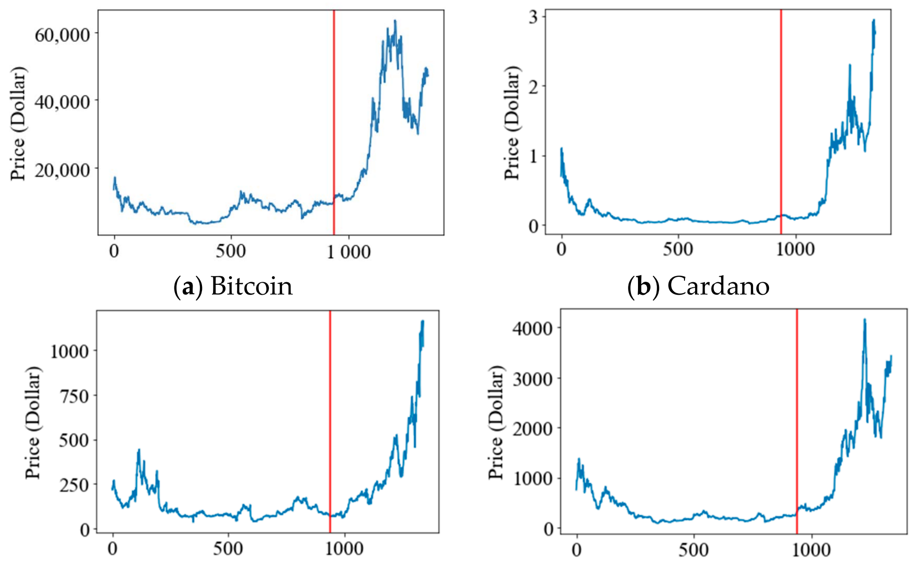
4.1. Experimental Design

The experimental design consists of three subsections: "Data Collection", "Model Architecture", and "Evaluation". First, in the "Data Collection" section, we describe a brief description of the collected data for the experiment. Next, in the "Model Architecture" section, we describe input features, a Tensorflow 2.0-based deep learning model, hyperparameters, and a final output used in the experiment. Last, in the "Evaluation" section, we describe the evaluation criteria used in the experiment with equations and examples.

4.1.1. Data Collection

The cryptocurrencies considered in this study primarily include BTC and ETH, which are currently the most popular cryptocurrencies. Furthermore, based on data since January 2018, as the popularity of cryptocurrencies began to increase, we considered a total of six cryptocurrencies, including ADA, DASH, LTC, and XMR, which have advantages in terms of security, scalability, and speed. Moreover, the input features used in the experiment included average price (P), opening price (O), highest price (H), lowest price (L), and trading volume (V). The input features were obtained from Investing.com (access date: 5 April 2022, website: https://www.investing.com/) [36].

Figure 6 shows a price graph for each cryptocurrency. For each graph, the x-axis represents the date, and the y-axis represents the price of the cryptocurrency on that date. Cryptocurrencies received what may be considered full-fledged attention in the second half of 2017. As a result, the amount of recorded data is insufficient. Thus, 1339 days of data were used from 1 January 2018 to 31 August 2021; as may be observed from Figure 6, all cryptocurrencies used in the experiment increased in volatility and fluctuation after a specific date. To evaluate the recommendation performance in this case, 70% of the data (937 days) were used to perform training and the remaining 30% (402 days) as testing data. In Figure 6, the left side shows the training data, and the right side shows the testing data based on the red vertical line.



(**a**) Bitcoin (**b**) Cardano

(**c**) Dash (**d**) Ethereum

Figure 6. *Cont.*

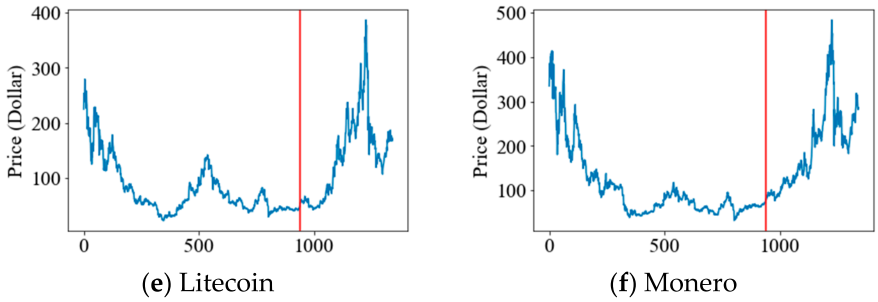


Figure 6. Price fluctuation graph of 6 major cryptocurrencies per each day. (**a**) Bitcoin; (**b**) Cardano; (**c**) Dash; (**d**) Ethereum; (**e**) Litecoin; (**f**) Monero.

4.1.2. Model Architecture

We used a deep learning model based on Tensorflow 2.0 with Python and the Jupyter Notebook to implement the proposed approach. To validate the experimental results, the experiment was conducted using the same deep learning model. The model architecture is shown in Figure 7. In the case of hyperparameters, the value was set to show optimal performance while consuming minimal time. Detailed values can be found in Table 3 [37].

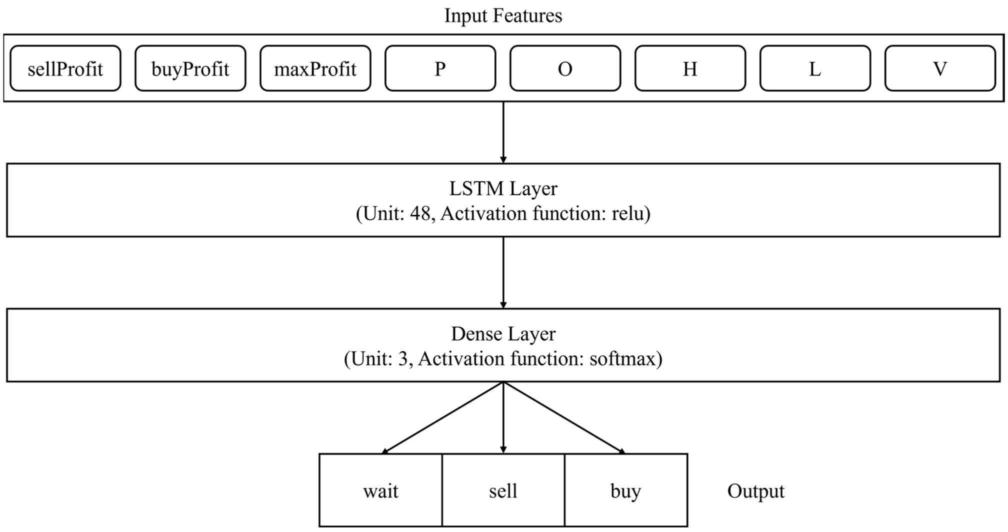


Figure 7. Deep learning model used in the experiment.

Table 3. Value of hyperparameters used in the deep learning model.

Hyperparameter	LSTM Layer	Last Layer
Unit	48	3
Activation Function	ReLU	softmax
Optimizer	Adam	
Loss function	categorical_crossentropy	
Learning rate	0.001	
Batch_size	64	
Epoch	100	

The study by Nelson et al. found that the action recommendation model showed the highest performance when using LSTM among LSTM, MLP, RF, and a pseudo-random model [26]. The study by Sanboon et al. also found that the action recommendation model showed the highest performance when using LSTM among LSTM, SVM, RF, MLP, KNN, Logistic Regression, and a Decision tree model [27]. Therefore, we used LSTM to determine experimentally whether the proposed model affected actual performance. Input features include P, O, H, L, and V by default, and we compare the extent to which performance differences occurred when sellProfit, buyProfit, and maxProfit were added as input features.

4.1.3. Evaluation

In this section, we describe the evaluation criteria used in the experiment. Because the model used in the experiment was a recommendation model, the accuracy and the F1 score were used as two appropriate evaluations [25–27]. In addition, the *t*-test was also used to verify that the proposed method exhibited a significant performance improvement compared to existing methods [26,27]. All evaluation criteria were calculated with Scipy, which is an open-source Python library used for scientific computing and technical computing. These evaluation criteria cannot be absolute if used alone. Thus, a more systematic evaluation was performed using the accuracy, F1 score, and the results of the *t*-test. Table 4 is a simple matrix for understanding the evaluation criteria used in our experiment. When the actual class and the recommended class are the same, and the class is "Positive", the result is a TP. On the contrary, when the actual class and the recommended class were the same, and the class was "Negative", the result is considered a TN. When the actual class and the recommended class differed, and the class was "Positive", the result was considered an FP. On the contrary, when the actual class and the recommended class differed, and the class was "Negative", the result becomes FN.

Table 4. Simple confusion matrix.

Confusion Matrix		Predicted Class	
		Positive	Negative
Actual Class	Positive	True Positive (TP)	False Negative (FN)
	Negative	False Positive (FP)	True Negative (TN)

- Accuracy

Accuracy can be calculated as shown in Equation (3). This equation is a value calculated by dividing the number of accurately predicted samples by the number of total samples and indicates the overall prediction accuracy. It has a value between 0 and 1. This implies higher predictive performance as the value increases [38]. However, accuracy involves a fatal problem as a measure of effectiveness. We assume that 100 samples would include, on average, 10 positive and 90 negative samples. If a predictive model predicts that all 100 samples were negative, the accuracy would be 0.9. However, this result would involve a class imbalance issue because positive classes were not successfully predicted at all. To solve this problem, the concepts of precision and recall emerged. Through the precision and the recall, we can calculate the F1 score.

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \tag{3}$$

- Precision

The precision can be calculated as shown in Equation (4). This equation represents the ratio at which the actual value is also a "Positive" class when the predictive model is predicted to be a "Positive" class. It takes the same value as the accuracy between 0 and 1. This means higher predictive performance with increasing value [39]. If the same example as the accuracy is applied, the precision becomes 0.

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{4}$$

- Recall

The recall can be calculated as shown in Equation (5). This equation represents the ratio at which the predicted value is also a positive class when the actual value is given a positive class. It takes the same range of values as the accuracy, between 0 and 1. This indicates higher predictive performance as the value increases [39]. If the same example as the accuracy is applied, the recall becomes 0.

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{5}$$

- F1 score

The F1 score can be calculated as shown in Equation (6). The precision and recall can be used effectively when class distribution imbalances. The F1 score is an equation that considers both the precision and the recall. It has the same value as the accuracy between 0 and 1. This means higher predictive performance as the value increases. In general, when both the precision and the recall were high, the F1 score was also high [40]. If the same example as the accuracy was applied, the F1 score becomes 0. Considering both the accuracy and the F1 score, the assumed predictive model cannot be said to have exhibited excellent predictive performance.

$$\text{F1 score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{6}$$

- *t*-test

The results of the *t*-test can be calculated as shown in Equation (7). The *t*-test is a statistical method that verifies whether there is a significant difference between the means of the two populations. In general, if the *t*-test's value is less than the *p*-value, a difference between the means of the two populations is interpreted as significant. On the contrary, if it is larger than the *p*-value, the difference between the means of the two populations is interpreted as insignificant [41].

$$t = \frac{\text{Mean of Sample A} - \text{Mean of Sample B}}{\sqrt{\frac{(\text{Standard Deviation of Sample A})^2}{\text{Size of Sample A}}} + \sqrt{\frac{(\text{Standard Deviation of Sample B})^2}{\text{Size of Sample B}}}} \tag{7}$$

4.2. Experimental Results

For the verification of the proposed model, 402 days of data from 25 July 2020 to 31 August 2021, which is the date of increasing price volatility and fluctuation among 1339 days of data from 1 January 2018 to 31 August 2021, were used as testing data.

The confusion matrix for all cryptocurrencies is shown in Table 5. We set "Positive" as "Wait" class and "Negative" as "Sell" and "Buy" class. When the actual class and the recommended class were the same, and the class was "Wait", the result was a TP. On the contrary, when the actual class and the recommended class were the same, and the class was "Sell" or "Buy", the result was considered a TN. When the actual class and the recommended class differed, and the class was "Wait", the result was considered an FP. On the contrary, when the actual class and the recommended class differed, and the class was "Sell" or "Buy", the result became FN. The results of the experiments in terms of the accuracy and the F1 score using test data and confusion matrix are summarized as follows.

Table 5. Confusion matrix of the proposed method.

Confusion Matrix		Recommended Class		
		wait	sell	buy
Actual Class	wait	TP	FN	FN
	sell	FP	TN	FN
	buy	FP	FN	TN

Table 6 shows the accuracy and F1 score of the existing and proposed models for each cryptocurrency. Compared to existing models, it can be seen that both the accuracy and the F1 score improved from about 13% to 21%. In terms of accuracy, the percentage of correct answers in the proposed method improved from about 13% to 21%. In addition, in terms of F1 score, the percentage of correct answers in the proposed method improved from about 13% to 21% when reflecting the class distribution imbalance. We used the *t*-test, a statistical technique, to assign validity to the performance difference for each cryptocurrency. The *t*-test is a technique to verify whether there was a significant difference between the two populations. Before performing the *t*-test, we checked the homogeneity of variance through the *F*-test. We use the homoscedastic *t*-test for the homogeneity of variance and the heteroscedastic *t*-test for the heterogeneity of variance [42]. The results of the *F*-test for each cryptocurrency and evaluation criteria can be found in Table 7. At this time, the significance level (*p*-value) was set to 0.05.

Table 6. Value of accuracy and F1 score for each cryptocurrency.

Cryptocurrency	Nelson et al. [26]		Sanboon et al. [27]		Proposed Method	
	Accuracy	F1 Score	Accuracy	F1 Score	Accuracy	F1 Score
Bitcoin	0.7090	0.7013	0.6865	0.6691	0.8606	0.8581
Cardano	0.7114	0.7007	0.6716	0.6606	0.8333	0.8285
Dash	0.7288	0.7228	0.6915	0.6787	0.8482	0.8268
Ethereum	0.6915	0.6405	0.6667	0.6233	0.7860	0.7371
Litecoin	0.7363	0.7199	0.7139	0.6970	0.8432	0.8348
Monero	0.7338	0.7267	0.7065	0.6785	0.8532	0.8404

Table 7. Value of the *F*-test in each cryptocurrency's accuracy and the F1 score.

Cryptocurrency	Nelson et al. [26]		Sanboon et al. [27]	
	Accuracy	F1 Score	Accuracy	F1 Score
Bitcoin	1.04×10^{-8}	1.13×10^{-8}	1.11×10^{-7}	1.15×10^{-7}
Cardano	0.018	0.024	0.038	0.022
Dash	0.024	0.032	1.28×10^{-9}	1.23×10^{-9}
Ethereum	0.035	4.84×10^{-5}	0.026	0.031
Litecoin	0.037	0.028	0.014	0.034
Monero	0.041	0.044	0.035	0.040

For all cases, when the value calculated through the *F*-test is smaller than the *p*-value, this means there is not the homogeneity of variance but the heterogeneity of variance. Based on this result, we perform the heteroscedastic *t*-test. Table 8 shows the results of the heteroscedastic *t*-test for each cryptocurrency and evaluation criteria. At this time, the *p*-value was set to 0.05, which is the same value as the *F*-test.

Table 8. Value of the *t*-test in each cryptocurrency's accuracy and the F1 score.

Cryptocurrency	Nelson et al. [26]		Sanboon et al. [27]	
	Accuracy	F1 Score	Accuracy	F1 Score
Bitcoin	1.56×10^{-10}	1.74×10^{-10}	1.08×10^{-9}	0.93×10^{-9}
Cardano	0.021	0.029	3.44×10^{-5}	2.86×10^{-7}
Dash	0.022	0.026	0.015	0.037
Ethereum	0.038	5.74×10^{-6}	0.018	0.021
Litecoin	0.031	0.036	1.96×10^{-7}	0.029
Monero	0.027	0.035	0.034	0.033

For all cases, the value calculated through *t*-test is smaller than the *p*-value, which means there was a significant difference between the two populations. Therefore, we concluded through statistical techniques that the proposed model showed significant performance improvements. We provide a confusion matrix showing the accuracy for each class for the systematic and transparency of the experiment. The confusion matrix can be found in Figure 8 [43].

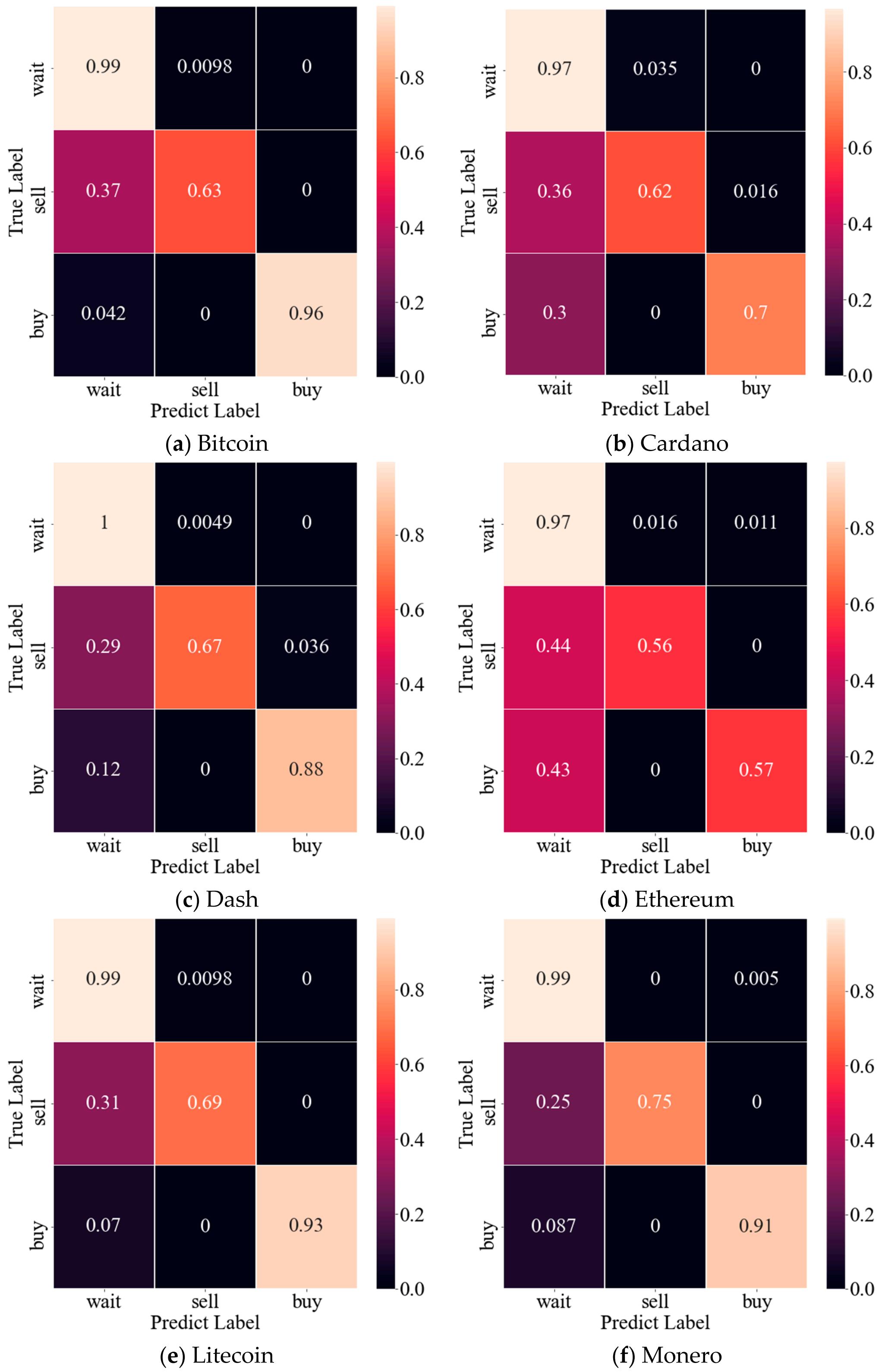


Figure 8. Confusion matrix of each cryptocurrency's class. (**a**) Bitcoin; (**b**) Cardano; (**c**) Dash; (**d**) Ethereum; (**e**) Litecoin; (**f**) Monero.

For all cryptocurrencies, the first row refers to accuracy for the "Wait" class, the second row refers to accuracy for the "Sell" class, and the third row refers to accuracy for the "Buy" class. It may be observed that the accuracy for the "Wait" class was high in all cryptocurrencies, and the accuracy was low for the other two. In particular, Ethereum showed lower performance than other cryptocurrencies. We confirmed a class distribution for each cryptocurrency to identify the reasons for this result, and the result is shown in Table 9.

Table 9. Class distribution for each cryptocurrency.

Class	Cryptocurrency					
	Bitcoin	**Cardano**	**Dash**	**Ethereum**	**Litecoin**	**Monero**
Wait	636	648	670	**822**	673	685
Sell	**307**	285	275	194	275	248
Buy	**396**	406	394	323	391	406

Table 9 presents the class distribution of each cryptocurrency. Bold font represents which cryptocurrency has the largest number of specific classes. In contrast, an italic font represents which cryptocurrency has the least number of specific classes. First of all, it may be observed that all cryptocurrencies have many classes in the order of "Wait"-"Buy"-"Sell". This class distribution is shown in Table 9. A phenomenon may be observed in which accuracy for each class also has a higher value or the value in the order of "Wait"-"Buy"-"Sell". In addition, in the case of ETH, it may be observed that the accuracy for "Buy" and "Sell" classes was lower than that of other cryptocurrencies. Excluding ETH, five cryptocurrencies exhibited between 600 and 700 "Wait" samples out of 1339 data. However, ETH had more than 800 "Wait" classes. Hence, the number of instances of learning "Buy" and "Sell" classes decreased, which caused the model's low performance for these classes. This is expected to be solved if an equal ratio between classes can be created, such as by increasing the amount of learning data.

The observations evident from these results are described as follows. The proposed model showed approximately 13% to 21% improvement in accuracy and F1 score compared to existing methods. In addition, we confirmed that there was no major problem in class distribution imbalance through the F1 score. Performance improvement for all cryptocurrencies was statistically verified by the *t*-test.

Moreover, for all cryptocurrencies, the class distribution was in the order of "Wait"-"Buy"-"Sell". It may also be observed that the class distribution was higher in the order of "Wait"-"Buy"-"Sell" in the Confusion Matrix. In the case of ETH, the accuracy of the "Buy" and "Sell" classes was lower than that of other cryptocurrencies. We confirmed the class distribution to identify the reason for this result and found that there were, in fact, an excessively large number of samples classified as "Wait" compared to other cryptocurrencies. We expect that this problem would be solved by correcting the class imbalance of the training data by creating an equal ratio between classes, such as by increasing the amount of learning data.

Table 10 shows the actual number of accurate prediction results for each cryptocurrency. The existing method [26] correctly predicted between 278 and 296 out of a total of 402 data, and existing method [27] correctly predicted between 268 and 287 out of a total of 402 data. Even though existing methods showed better prediction performance than the proposed method at certain short-term test data, the proposed method showed higher prediction performance generally. The proposed method correctly predicted between 316 and 346 out of a total of 402 data.

Table 10. The number of times each method correctly predicted out of a total of 402 test data.

Cryptocurrency	Nelson et al. [26]	Sanboon et al. [27]	Proposed Method
Bitcoin	285	276	346
Cardano	286	270	335
Dash	293	278	341
Ethereum	278	268	316
Litecoin	296	287	339
Monero	295	284	343

Considering the results, our proposed model showed approximately 13% to 21% improvement over the existing model. Moreover, the *t*-test results indicate that the performance improvement was meaningful. The results of this work provide clients with the recommended action that should be taken at the current time. Hence, the proposed method is expected to provide enhanced convenience and facilitate timely and effective decision-making. Through this paper, we also expect that researchers will be able to propose newly calculated input features based on existing research papers. They may be able to propose brand new methods to improve the performance of prediction models.

5. Conclusions and Future Research

In this study, we proposed a deep learning-based model designed to recommend actions to support maximizing short-term profit-taking by cryptocurrency clients operating roughly in the manner of conventional day traders. We collected data on a total of six representative cryptocurrencies: BTC, ADA, DASH, ETH, LTC, and XMR. Accuracy and F1 score were used as evaluation metrics for objective experimentation and verification. Furthermore, a statistical technique called a *t*-test was also used, in contrast to previous studies. Finally, the meaningful performance improvement of the model was systematically verified and analyzed compared to existing methods.

Compared to existing methods that only used the LSTM model, the accuracy and F1 score of the proposed model increased by approximately 13% to 21%. The experimental results indicate that the proportion of the correct answers was about 13% to 21% higher for the proposed model. The results also indicate that there was no major problem in terms of the class distribution imbalance. Moreover, the results of the *t*-test were lower than the *p*-values for all cryptocurrencies. The *t*-test results indicate that the performance improvement was meaningful.

Among the cryptocurrencies used in the experiment, Ethereum exhibited lower performance in terms of F1 score than the other five cryptocurrencies, so we analyzed this result using the confusion matrix. In particular, the "Sell" and "Buy" classes showed low performance. When the amount of data per class was analyzed, Ethereum contained approximately 800 samples classified as "Wait" out of a total of 1339 data samples. In contrast, other cryptocurrencies contained approximately 600 to 700 "Wait" classes. Hence, there was less opportunity to learn the "Sell" and "Buy" classes, and the results of the analysis showed that this issue caused the low performance of these classes.

The results of this work provide clients with the recommended action that should be taken at the current time. Hence, the proposed method is expected to provide enhanced convenience and facilitate timely and effective decision-making. In addition, cryptocurrency is increasingly being integrated with various new technologies such as NFTs and metaverse systems. In accordance with this trend, it is expected that cryptocurrency price prediction and action recommendations can be used as auxiliary indices from a perspective of transactions made within the market and market price adjustments. Through this paper, we also expect that researchers will be able to propose newly calculated input features based on existing research papers. They could be able to propose brand new methods to improve the performance of prediction models.

Although the proposed method exhibited higher performance than the existing method, it only focused on the maximum profit that could be made in a single day. Therefore, there is a limitation that a period longer than one day was not considered. To make recommendations by considering a wider period of time, research needs to be conducted to improve the performance of interval prediction methods, which reflect information from the distant future. However, the accuracy of interval prediction is very low, and most studies on interval prediction remain extremely inadequate. For example, most existing methods only predict the price for the following day [44]. Therefore, we plan to conduct a study to improve the performance of interval prediction methods, which predict prices in the more distant future, such as three or seven days or one month. Moreover, based on existing research on interval prediction research and the results of the present work, we expect to further advance the state of the art with a deep learning model designed to recommend actions that cryptocurrency clients should take to maximize their profit over a longer period of time.

Author Contributions: Conceptualization, J.P. and Y.-S.S.; Data curation, J.P.; Formal analysis, J.P. and Y.-S.S.; Funding acquisition, Y.-S.S.; Investigation, J.P.; Methodology, J.P. and Y.-S.S.; Project administration, Y.-S.S.; Resources, J.P.; Software, J.P.; Supervision, Y.-S.S.; Validation, J.P. and Y.-S.S.; Visualization, J.P.; Writing—original draft, J.P.; Writing—review and editing, Y.-S.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the Basic Science Research Program through the National Research Foundation of Korea (NRF), funded by the Ministry of Education (NRF-2020R1I1A3073313).

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

References

1. Eyal, I. Blockchain technology: Transforming libertarian cryptocurrency dreams to finance and banking realities. *Computer* **2017**, *50*, 38–49. [CrossRef]
2. Mukhopadhyay, U.; Skjellum, A. A brief survey of cryptocurrency systems. In Proceedings of the 2016 14th Annual Conference on Privacy, Security and Trust, Auckland, New Zealand, 12–14 December 2016; pp. 745–752.
3. Ciaian, P.; Rajcaniova, M. Virtual relationships: Short-and long-run evidence from BitCoin and altcoin markets. *J. Int. Financ. Mark. Inst. Money* **2018**, *52*, 173–195. [CrossRef]
4. Meunier, S. Blockchain 101: What is blockchain and how does this revolutionary technology work? In *Transforming Climate Finance and Green Investment with Blockchains*; Marke, A., Ed.; Elsevier: Amsterdam, The Netherlands, 2018; pp. 23–34.
5. Huh, J.H.; Seo, K. An indoor location-based control system using bluetooth beacons for IoT systems. *Sensors* **2017**, *17*, 2917. [CrossRef] [PubMed]
6. Sharma, P.K.; Park, J.H. Blockchain based hybrid network architecture for the smart city. *Future Gener. Comput. Syst.* **2018**, *86*, 650–655. [CrossRef]
7. Kim, H.W.; Jeong, Y.S. Secure authentication-management human-centric scheme for trusting personal resource information on mobile cloud computing with blockchain. *Hum.-Centric Comput. Inf. Sci.* **2018**, *8*, 11. [CrossRef]
8. Lee, Y.; Rathore, S. A blockchain-based smart home gateway architecture for preventing data forgery. *Hum.-Centric Comput. Inf. Sci.* **2020**, *10*, 9. [CrossRef]
9. Dowling, M. Is non-fungible token pricing driven by cryptocurrencies? *Finance Res. Lett.* **2022**, *44*, 102097. [CrossRef]
10. Duan, H.; Li, J. Metaverse for social good: A university campus prototype. In Proceedings of the 29th ACM International Conference on Multimedia, Chengdu, China, 20–24 October 2021; pp. 153–161.
11. Zexin, H.; Yiqi, Z. A survey of forex and stock price prediction using deep learning. *Appl. Syst. Innov.* **2021**, *4*, 9.
12. Khedr, A.M.; Arif, I. Cryptocurrency price prediction using traditional statistical and machine-learning techniques: A survey. *Intell. Syst. Account. Finance Manag.* **2021**, *28*, 3–34. [CrossRef]
13. Sebastiao, H.; Godinho, P. Forecasting and trading cryptocurrencies with machine learning under changing market conditions. *Financial Innov.* **2021**, *7*, 3. [CrossRef]
14. Park, J.; Seo, Y.S. Understanding the Association Between Cryptocurrency Price Predictive Performance and Input Features. *KIPS Trans. Softw. Data Eng.* **2022**, *11*, 19–28.
15. Hochreiter, S.; Schmidhuber, J. Long short-term memory. *Neural Comput.* **1997**, *9*, 1735–1780. [CrossRef] [PubMed]
16. Hoseinzade, E.; Haratizadeh, S. CNNpred: CNN-based stock market prediction using a diverse set of variables. *Expert Syst. Appl.* **2019**, *219*, 273–285. [CrossRef]
17. Livieris, I.E.; Kiriakidou, N. An advanced CNN-LSTM model for cryptocurrency forecasting. *Electronics* **2021**, *10*, 287. [CrossRef]
18. Patel, M.; Tanwar, S. A deep learning-based cryptocurrency price prediction scheme for financial institutions. *J. Inf. Secur. Appl.* **2020**, *55*, 102583. [CrossRef]
19. Koo, E.; Kim, G. A Hybrid Prediction Model Integrating GARCH Models with a Distribution Manipulation Strategy Based on LSTM Networks for Stock Market Volatility. *IEEE Access.* **2022**, *10*, 34743–34754. [CrossRef]

20. Valencia, F.; Gomez-Espinosa, A. Price movement prediction of cryptocurrencies using sentiment analysis and machine learning. *Entropy* **2019**, *21*, 589. [CrossRef]
21. Maqsood, H.; Mehmood, I. A local and global event sentiment based efficient stock exchange forecasting using deep learning. *J. Inf. Manag.* **2020**, *50*, 432–451. [CrossRef]
22. Aasi, B.; Imtiaz, S.A. Stock Price Prediction Using a Multivariate Multistep LSTM: A Sentiment and Public Engagement Analysis Model. In Proceedings of the 2021 IEEE International IOT, Electronics and Mechatronics Conference, Toronto, ON, Canada, 21–24 April 2021; pp. 1–8.
23. Xuan, Y.; Yu, Y. Prediction of Short-term Stock Prices Based on EMD-LSTM-CSI Neural Network Method. In Proceedings of the 2020 5th IEEE International Conference on Big Data Analytics, Xiamen, China, 8–11 May 2020; pp. 135–139.
24. Huang, N.E.; Shen, Z. The empirical mode decomposition and the Hilbert spectrum for nonlinear and non-stationary time series analysis. *Proc. R. Soc. A Math. Phys. Eng. Sci.* **1998**, *454*, 903–995. [CrossRef]
25. Hadi, R.; Hamidreza, F. Stock price prediction using deep learning and frequency decomposition. *Expert Syst. Appl.* **2021**, *169*, 114332.
26. Nelson, D.M.Q.; Pereira, A.C.M. Stock market's price movement prediction with LSTM neural networks. In Proceedings of the 2017 International Joint Conference on Neural Networks, Anchorage, AK, USA, 14–19 May 2017; pp. 1419–1426.
27. Sanboon, T.; Keatruangkamala, K. A deep learning model for predicting buy and sell recommendations in stock exchange of thailand using long short-term memory. In Proceedings of the 2019 IEEE 4th International Conference on Computer and Communication Systems, Singapore, 23–25 February 2019; pp. 757–760.
28. Salimitari, M.; Chatterjee, M. Profit maximization for bitcoin pool mining: A prospect theoretic approach. In Proceedings of the 2017 IEEE 3rd International Conference on Collaboration and Internet Computing, San Jose, CA, USA, 15–17 October 2017; pp. 267–274.
29. Vranken, H. Sustainability of bitcoin and blockchains. *Curr. Opin. Environ. Sustain.* **2017**, *28*, 114332. [CrossRef]
30. Worley, C.; Skjellum, A. Blockchain tradeoffs and challenges for current and emerging applications: Generalization, fragmentation, sidechains, and scalability. In Proceedings of the 2018 IEEE International Conference on Internet of Things and IEEE Green Computing and Communications and IEEE Cyber, Physical and Social Computing and IEEE Smart Data, Halifax, NS, Canada, 30 July–3 August 2018; pp. 1582–1587.
31. Sovbetov, Y. Factors influencing cryptocurrency prices: Evidence from bitcoin, ethereum, dash, litcoin, and monero. *J. Econ. Finance* **2018**, *2*, 1–27.
32. Mensi, W.; Al-Yahyaee, K.H. Structural breaks and double long memory of cryptocurrency prices: A comparative analysis from Bitcoin and Ethereum. *Finance Res. Lett.* **2019**, *29*, 222–230. [CrossRef]
33. Tanwar, A.K.; Kumar, S. Modelling the dynamics of Bitcoin and Litecoin: GARCH versus stochastic volatility models. *Appl. Econ.* **2019**, *51*, 2073–4082.
34. Li, Y.; Yang, G. Traceable monero: Anonymous cryptocurrency with enhanced accountability. *IEEE Trans. Dependable Secure Comput.* **2019**, *18*, 679–691. [CrossRef]
35. Ji, L.J.; Zhang, Z. To buy or to sell: Cultural differences in stock market decisions based on price trends. *J. Behav. Decis. Mak.* **2008**, *21*, 399–413. [CrossRef]
36. Investing.com. Available online: https://www.investing.com/ (accessed on 5 April 2022).
37. Hutter, F.; Hoos, H. An efficient approach for assessing hyperparameter importance. In Proceedings of the International Conference on Machine Learning, Beijing, China, 21–26 June 2014; pp. 754–762.
38. Zhu, W.; Zeng, N. Sensitivity, specificity, accuracy, associated confidence interval and ROC analysis with practical SAS implementations. In Proceedings of the NESUG Proceedings: Health Care and Life Sciences, Baltimore, MD, USA, 14–17 November 2010; pp. 1–67.
39. Park, S.W.; Huh, J.H. BEGAN v3: Avoiding mode collapse in GANs using variational inference. *Electronics* **2020**, *9*, 688. [CrossRef]
40. Huang, H.; Xu, H. Maximum F1-score discriminative training criterion for automatic mispronunciation detection. *IEEE ACM Trans. Audio Speech Language Process.* **2015**, *23*, 787–797. [CrossRef]
41. Kim, T.K. T test as a parametric statistic. *Korean J. Anesthesiol.* **2015**, *68*, 540. [CrossRef]
42. Sayago, A.; Asuero, A.G. Fitting straight lines with replicated observations by linear regression: Part II. Testing for homogeneity of variances. *Crit. Rev. Anal. Chem.* **2004**, *34*, 133–146. [CrossRef]
43. Legg, P.; Smith, J. Visual analytics for collaborative human-machine confidence in human-centric active learning tasks. *Hum.-Centric Comput. Inf. Sci.* **2019**, *9*, 5. [CrossRef]
44. Lops, Y.; Choi, Y. Real-time 7-day forecast of pollen counts using a deep convolutional neural network. *Neural. Comput. Appl.* **2019**, *32*, 11827–11836. [CrossRef]