Journal of Mathematics and Modeling in Finance (JMMF)
Vol. 2, No. 2, Summer & Autumn 2022
Research paper

Measuring the Accuracy and Precision of Random Forest, Long Short-Term Memory, and Recurrent Neural Network Models in Predicting the Top and Bottom of Bitcoin price

Emad Koosha[1], Mohsen Seighaly[2], Ebrahim Abbasi[3]

[1] Financial engineering Ph.D. Candidate, Department of Financial Management, Qazvin Branch, Islamic Azad University, Qazvin, Iran

Emadkoosha92@gmail.com

[2] Department of Financial Management, Qazvin Branch, Islamic Azad University, Qazvin, Iran
mohsenseighaly@gmail.com

[3] Department of management, faculty of social sciences and economics, Alzahra University, Tehran, Iran

abbasiebrahim2000@alzahra.ac.ir

Abstract:
The purpose of the present research is to use machine learning models to predict the price of Bitcoin, representing the cryptocurrency market. The price prediction model can be considered as the most important component in algorithmic trading. The performance of machine learning and its models, due to the nature of price behavior in financial markets, have been reported to be well in studies. In this respect, measuring and comparing the accuracy and precision of random forest (RF), long-short-term memory (LSTM), and recurrent neural network (RNN) models in predicting the top and bottom of Bitcoin prices are the main objectives of the present study. The approach to predicting top and bottom prices using machine learning models can be considered as the innovative aspect of this research, while many studies seek to predict prices as time series, simple, or logarithmic price returns. Pricing top and bottom data as target variables and technical analysis indicators as feature variables in the 1-hour time frame from 1/1/2018 to 6/31/2022 served as input to the mentioned models for learning. Validation and testing are presented and used. 70% of the data are considered learning data, 20% as validation data, and the remaining 10% as test data. The result of this research shows over 80% accuracy in predicting the top and bottom Bitcoin price, and the random forest models prediction is more accurate than the LSTM and RNN models.

Keywords: Algorithmic Trading, Random Forest, Recurrent Neural Network, Long-Short term memory, Top and bottom price prediction.
JEL Classification: F17, F19, G17, B17, C53

Introduction

With the development of new technologies and software products, financial markets undergo constant changes (Reimann, 2018). The development of software technolo-

[2]Corresponding author
Received: 23/07/2022 Accepted: 14/09/2022
https://doi.org/10.22054/jmmf.2022.69198.1068

gies for transactions in financial markets aims to help traders make decisions in the market. The facilitation of decision-making by software technologies has caused intelligent trading systems to comply with a transparent trading algorithm, so the transaction decisions are made based on the predetermined algorithm by the intelligent trading system. Algorithmic trading using computers has changed into one of the most popular and attractive areas in the academic field of finance and computer science in recent years (Hu, 2015; Rivera, 2015).

Algorithmic trading is a process in which computers are programmed to make appropriate orders for the transaction to maximize returns and minimize the risk of transactions simultaneously (Reimann, 2018).

Managers use algorithms in different ways. Money management and index funds, pension funds, quantitative funds, and hedge funds use algorithms to implement investment decisions. Portfolio managers utilize suitable portfolio techniques and various stocks for their organization and implement decisions by adopting an algorithm. The algorithm selects the best price, time, and number of assets to enter the market. Sometimes this approach makes a decision independent of human transactions (Kissele, 2014).

In general, the reasons traders use algorithmic trading methods can be classified as follows (Mousavi, 2019):

1 One of the main reasons for using algorithmic trading is that there is no need to manage emotions in financial markets.

2 Data preparation and processing and using the obtained information for decision-making in transactions are processes that must be undertaken for the entry or exit of the capital of a real or financial asset. However, conducting such processes may not be possible for humans for two reasons; data processing may be complex, or the time available for processing may be very short, so they should be completed in a few seconds. Therefore, due to either of these reasons, using intelligent trading systems and algorithmic trading is preferable to manual trading.

3 The ability to test the strategy for a long time with a lot of data (Back Test) makes the uncertainty in the transaction approach tend toward the atmosphere of profit with lower risk. In fact, it is possible to test a trading strategy because it is developed in the form of an intelligent algorithmic trading system, which enables one to determine the strengths and weaknesses of the given strategy in a considerable period. By identifying the strengths and weaknesses, one can seek to enhance the strengths and eliminate the weaknesses of a strategy. However, a manual trader in the financial markets does not know how likely they will succeed or how much potential profit and loss may incur in the position (transaction) they start. That is why a trader may make a profit in a period, and in another period, not only may lose the same profit but also suffer a loss. Therefore, implementing trading strategies that

consist of analytical systems, risk management, and capital management in the form of algorithmic trading can provide a trader with a long-term roadmap and sustainable profit with a logical risk.

4 The limitations of the financial market in which the transaction is carried out are another issue that a manual trader may not be able to consider. For example, external factors and even the impact of the trading strategy on itself due to the low market depth (market impact) may turn a profitable trading strategy with low capital into a losing strategy when the capital is high.

As stated earlier, algorithmic trading is developing every day, and the most critical element of such transactions is the analytical system or, in other words, the price prediction model. In a trading strategy, if a prediction method can provide an appropriate return with a logical success rate, a successful strategy can be achieved through relevant risk and capital management indicators.

One of the essential steps in designing algorithmic trading is the price analysis and prediction system. Hence, the actors looking for an intelligent trading system with a reliable and sustainable return are aware of the necessity and importance of the price prediction model. In this respect, the present study intends to find a suitable analytical system for price prediction. In this research, instead of predicting price data or simple or logarithmic returns as a time series, the top and bottom prices (peak and trough points) are predicted as target variables using technical analysis indicators with a machine learning approach. Therefore, this study, using the values of technical analysis indicators as characteristic variables, seeks to predict the top and bottom points of the Bitcoin price in a 1-hour time frame from 2018 to the end of the first half of 2022. The model presented in this study claims that in the next step of the market, it creates a price bottom or top point. To this aim, LSTM, RNN, and RF models (0 bottom and 1 top) are used for the prediction, and their results are compared. Predicting the financial markets through machine learning classification models can be considered as the contribution of this research to the existing body of knowledge in this area. The significance of this study lies in the intelligent trading systems need for prediction models with optimal accuracy.

Theoretical background and literature review

Algorithmic trading

The most common algorithmic trading methods implement the technical analysis. However, the efficient market hypothesis (EMH) states that technical analysis is not useful for profitability. It briefly explains that gaining income through investment or trading is difficult. On the other hand, machine learning can extract and model the behavioral pattern of the data by repeatedly reducing the gap between the predicted and actual values. In most uses of machine learning, the focus is on functional modeling, which measures the difference between the collected time series data and the data obtained from the inferred model produced over time. (Peng & Lee, 2021)

So far, various methods have been proposed for algorithmic trading. In the present study, machine learning models of Recurrent Neural Network (RNN), Random Forest (RF), and Long-Short term memory (LSTM) are employed to predict the top and bottom prices. Therefore, this section reviews studies related to price prediction that have used these methods.

Studies on algorithmic trading using the RF model

Random forests (RF) is a non-parametric and nonlinear regression and classification algorithm that was first proposed by Ho (1995) and developed by Breiman (2001). Creamer and Freund (2004) used this technique to successfully predict corporate performance and measure corporate governance risk in Latin American banks. They compared the performance of RF with logistic regression and Adaboost. They concluded that RF consistently yields better results.

Lariviere and Vandenpoel (2005) presented the merits of RF in financial predictions. They showed that RF regression could be used to explore customer retention and profitability. They analyzed a sample of one hundred thousand customers using data obtained from a large financial services company in Europe. They found that RF techniques produced better validation and test sample results than linear regression and logistic regression models.

Maragoudakis and Serpanos (2010) employed a method called Markov Blanket Random Forest to predict the direction of stock markets. They indicated that their strategy outperformed a simple buy-and-hold investment strategy, reporting an average of 12.5% to 26% for the initial period and 16% to 48% for the other periods. They also reported a better performance compared to linear regression, SVMs, and ANNs.

Qin et al. (2013) used the gradient-boosted RF method to predict the direction of the Singapore stock market. By raising the weight of individual forest trees and using a forgetting factor to handle market changes, their experimental results revealed that their proposed methods could generate higher returns than the buy-and-hold strategy.

Zbikowski and Grzegorzewski (2013) used a novel online adaptation method to allow RF to adapt to non-stationary financial time series.

Xu et al. (2013) demonstrated the capability of RF algorithms in selecting features to predict stock price trends.

Moritz and Zimmermann (2014) used an RF model to predict stock returns using US CRSP data. In this model, the high decile is bought while the low decile is sold as a trading strategy.

Lohrmann and Luukka (2019) developed a classification model using the RF to predict the opening and closing price returns of S& P500 stocks.

Basak et al. (2019) used RF models and gradient-boosted decision trees (XG-Boost) along with a set of technical analysis indicators to analyze the mid-term and long-term prediction performance of stock price returns.

Sadorsky (2021) used the RF to predict the direction of the prices of clean energy stock exchange trade funds.

Krauss et al. (2017) compared different deep learning methods, such as deep neural networks, gradient-boosted decision trees, and random forests. They found that daily returns were provided according to the closing prices of the S&P 500 from December 1992 to October 2015 to predict the probability of the markets better performance for each stock in the coming day. As a trading strategy, the ten stocks with the highest probability were bought, and the ten stocks with the lowest probability were short-sold, all with equal monetary weight. RF appeared to achieve the highest return of any of the above-mentioned deep learning methods, with a return of 0.43% per day before transaction costs.

As these studies suggest, the performance of the RF model is acceptable compared to some machine learning models, and the important point is that all of them were seeking to predict the price and return.

Studies on algorithmic trading using the LSTM model

Ghosh et al. (2021) used RF and LSTM networks as training methods to investigate their effectiveness in predicting directional movements out of the sample of S&P 500 stocks. In this study, the authors introduced multi-dimensional settings, which included returns based not only on the closing prices but also on the opening prices and intraday returns. As a trading strategy, they used Krauss et al. (2017) and Fisher and Krausss (2018) strategy as a benchmark. On each trading day, they bought ten stocks with the highest probability and sold ten stocks with the lowest probability of outperforming the market in terms of daily returns - all with the same monetary weight. Their empirical findings indicated that the proposed multi-dimensional settings provided a daily return of 0.64% using LSTM networks and 0.54% using RF before transaction costs. Hence, they had a better performance than the one-dimensional settings of Fisher and Krauss (2018) and Krauss et al. (2017), which only included daily returns considering the closing prices, with corresponding daily returns of 0.41% and 0.39% for LSTM and RF, respectively.

Siami-Namini and Namin (2018) compared LSTM and ARIMA models. Their empirical results on financial data showed that LSTM performs better than ARIMA in terms of lower prediction errors and higher accuracy.

Fisher and Krauss (2018), following the study of Krauss et al. (2017), used LSTM networks as a deep learning methodology and obtained a return of 0.46% per day before applying transaction costs. They concluded that the methodology used outperforms all memoryless methods mentioned in Krauss et al. (2017).

Sang and Di Pierro (2018) utilized an LSTM neural network for learning and improving traditional trading algorithms used in technical analysis. They argued that the network could learn market behavior and predict when a given strategy was more likely to succeed. They showed that the combinational strategy of neural network prediction and the traditional technical analysis performed better than technical analysis alone.

Lee and Yun (2018) compared three types of recurrent neural networks, including recurrent neural networks with the gated recurrent unit and neural networks with long-short-term memory (LSTM) for predicting stock returns. The results revealed that the LSTM neural network had the best performance. They also created a portfolio based on threshold limit prediction according to the results obtained from LSTM neural network predictions. This model was more data-oriented than the existing models for portfolio development. Their empirical findings pointed to the fact that this portfolio has promising returns.

Khare et al. (2017) predicted the short-term prices of 10 unique stocks listed on the NYSE using MLP and LSTM. Their study showed that the LSTM model successfully predicted the future price trend at almost all points. However, the model could not predict the exact price with the necessary accuracy. On the other hand, the MLP model could depict future trends and predict the prices with very high accuracy compared to the LSTM model.

Sharma et al. (2021) observed that both the LSTM model and the autoregressive moving average model with exogenous variables (ARIMAX), considering the sentiment analysis, could significantly improve the prediction of stock price trends.

Saiful Islam et al. (2020) presented a new model that combined two robust neural networks used for predicting time series: Gated Recursive Unit (GRU) and Long-Short-Term Memory (LSTM), to predict future closing prices of forex currencies. In addition, they compared the performance of their model with an independent LSTM model, an independent GRU model, and a statistical model based on a simple moving average (SMA), where the combined GRU-LSTM model outperformed all other models.

Studies on algorithmic trading using the RNN model

Singh Saud and Shakya (2019) conducted a novel analysis of the recurrence period of the parameter used with recurrent neural networks. They also compared the performance of three deep learning models, i.e., vanilla RNN, LSTM, and GRU, in predicting the stock prices of two commercial banks listed on the Nepal Stock Exchange. Their investigation revealed that GRU was the most successful in predicting stock prices.

Hiransha et al. (2018) used four deep learning models and a linear ARIMA prediction model to predict stock prices listed on the National Stock Exchange of India (NSE) and the New York Stock Exchange (NYSE). Researchers trained four networks, i.e., MLP, RNN, LSTM, and CNN, with the stock price of TATA MOTORS from NSE, and this model was used to predict the stock of NSE and NYSE. They observed that the models were able to identify patterns in both stock markets and concluded that there was a basic dynamic common to both stock markets. The results showed that linear models like ARIMA could not identify the basic dynamics in different time series, and deep learning models outperformed the ARIMA model. In addition, CNN performed better than the other three deep learning architectures.

Selvin et al. (2017) proposed an overlapping sliding window-based approach with RNN, LSTM, and CNN. A window size of 100 minutes was set with a data overlap of 90 minutes, and the prediction was made for 10 minutes in the future. This study showed that CNN provided more accurate results than the other two models.

Local studies in Iran

Studies show that the research process in Iran lags behind the global research process. Among the early studies, one can refer to Sajjadi and Sefidchian (2018). They investigated the impact of technical analysis indicators on the short-term returns of the stockholders. They found that the simple moving average and relative strength index strategies had predictive power and could identify price patterns to make profitable trades. They examined the usefulness of the simple moving average and the relative strength index for the next three and seven days.

In another study, Raeisi and Zakizadeh (2011) tested a large number of moving average rules on 20 selected companies on the Tehran Stock Exchange for 111 months (March 2001 to June 2010). By comparing the average annual returns resulting from the simple moving average, the exponential moving average (with 16 different pairs of parameters), and the buy and hold strategy, they showed that applying moving average techniques produced higher returns. The buy and hold strategy yielded a net annual return of 11.87%, while the EMA (1.25) earned a net return of 104.4%.

Tehrani and Esmailis study (2013) on the impact of using important indicators of technical analysis on the short-term returns of investors in the Tehran Stock Exchange showed that some of the indicators used in this research were more dispersed than the buy and hold method. Some of them also had lower relative dispersion. In their research, the stochastic indicator with an average coefficient of variation of 1.63 was declared to be the most stable. The weighted moving average index with an average coefficient of variation of 6.99 was shown as the most unstable trading strategy from 2003 to 2005.

Abbasi et al. (2020) compared the performance of strategies based on technical analysis indicators with buy and hold strategies and concluded that strategies based on technical analysis could yield better returns than the buy and hold strategy.

Therefore, using indicators can be helpful in prediction.

Zargari and Lari (2015) utilized a combination of technical analysis indicators such as Ichimoku clouds and relative strength index with some mathematical and trading rules to predict the next days stock market price. They used the relative strength index with its default parameter, i.e., 14 days. They presented a profitable algorithm to do it automatically. They tested their proposed model between 2010-2014, which approximately tripled the initial capital.

Alamooti, Haddadi, and Nademi (2017) modeled and evaluated the performance of different models of short-long-term memory, Markov switching, and hyperbolic Garch in predicting OPEC crude oil price fluctuations. They used the model to predict OPEC oil price fluctuations from 2010 to 2016 and measured its accuracy based on the RMSE criterion. The results of this evaluation pointed to the superiority of the two-regime Markov switching Garch model on a one-day horizon. Also, the long-term memory model predicted oil price fluctuations in 1 and 10-day prediction horizons better than the competing models.

Moshari et al. (2018) compared the golden points in the automobile industry stock price diagram of the Iranian capital market from 2010 to 2015 using the results of models and optimized them through the genetic algorithm. Prediction of the golden points with appropriate accuracy and the effect of optimization on error reduction was the outcome of their research. Bashiri and Paryab (2019) compared the performance of RF models, support vector, boosted-gradient, and perceptron multilayer neural network for predicting the price of Bitcoin using the data of 9 other cryptocurrencies. They observed that the accuracy of the boosted-gradient model was higher than the rest.

Research questions

Considering the objective of the study, which is predicting the top and bottom price of bitcoin using machine learning models, the following research questions are formulated:

1 What is the accuracy of predicting Bitcoin's top and bottom prices using RF, LSTM, and recurrent neural network models?

2 Is the accuracy of predicting Bitcoin's top and bottom prices using the RF model higher than LSTM and recurrent neural network models?

3 What are the consequences of improving the accuracy and precision of predicting Bitcoin's top and bottom prices for traders and investors?

Research Methodology

Since the target financial market in the present study is the cryptocurrency market (Bitcoin), using a credible database is crucial. Hence, the candlestick price data (OHLCV) of Bitcoin in the 1-hour time frame was selected as the most significant cryptocurrency representing this market. To extract this data, the Python Historic-Crypto module was used, which extracts data from the API of the CoinBase Pro exchange.

The cryptocurrency market has a 4-year cyclical behavior due to halving the mining reward. Therefore, the period of 2018-2022 was selected from the entire Bitcoin data available since 2010. In 2018, Bitcoin experienced a stagnant market and then a decline. From 2019 to 2021, it also experienced an upward trend due to the Covid-19 pandemic and the halving of the reward in 2020. In 2022, there will also be a recession and a downward trend that can be considered a return to 2018. It can be argued that our sample represents all cyclical phases. The first 70% of the data are fed to the model as training data. 20% of the data are considered validation data, and the last 10% of the data are provided to the model as test data.

Due to GPU sharing, this study was conducted using Python programming language and its valid modules in the Google Colab platform. Also, NumPy, Pandas, ta, TensorFlow, Sklearn, and Scipy libraries were specifically used for implementation.

The variables of the study are divided into two main categories. The target variable is the variable we are trying to predict. Our target variable in this research is the top or bottom price (1 or 0), which is calculated using the Awesome Oscillator (AO) indicator from the closing price of candles. Other variables are technical analysis indicators as the feature variables for predicting the top and bottom prices. The correlation of over 150 indicators and oscillators with the target variable was examined to select these variables from the existing libraries. Indicators (numerical) whose Pearson correlation was above 0.7 with a p-value of 0.05 were selected as feature variables. The list of these indicators is presented below. The learning data seek to discover the relationship between the list of indicators and the target variable, i.e., the top and bottom prices, using RF, RNN, and LSTM models. The numbers related to the indicator and oscillator are normalized by dividing by the closing price number to be on the same scale. As mentioned, 70% of the data are used as learning data, 20% as validation data, and the rest as test data. The list of indicators and oscillators used in this study is as follows:

Table1: The list of feature variables used in the models

Variable name	Name in the model	Variable name	Name in the model
Average volume	'volume_sma_em'	Volume (money flow index)	volume_mfi
fluctuation KCP	'volatility_kcp'	fluctuation BBP	'volatility_bbp'

trend MACD	'trend_macd'	fluctuation DCP	'volatility_dcp
trend ADX	'trend_adx_pos'	Trend difference MACD	'trend _macd _diff'
Trend difference VOREXT	'trend _vortex _ind _diff'	VORTEX trend	'trend _vortex _ind _pos'
trend AROON	'trend _aroon _up'	trend CCI	'trend_cci'
momentum RSI	'momentum _rsi'	trend STC	'trend_stc'
momentum UO	'momentum _uo'	momentum TSI	'momentum_ tsi'
momentum STOCH SIGNAL	'momentum _stoch _signal'	STOCHASTIC momentum	'momentum_ stoch'
momentum AO	'momentum_ao'	momentum WR	'momentum_wr'
AO	'ao'	momentum ROC	'momentum_ roc'
Above moving average, 10	'aboveEMA10'	Relative strength index	'RSI'
Above moving average, 20	'aboveEMA20'	Above moving average, 15	'aboveEMA15'
Above moving average, 40	'aboveEMA40'	Above moving average, 30	'aboveEMA30'
Above moving average, 60	'aboveEMA60'	Above moving average, 50	'aboveEMA50'

In the above list, technical analysis indices are introduced in relation to the volume of transactions, price fluctuations, trends, price momentum, and binary indicators. These indicators have significant relationships with the target variable (top or bottom price). It should be mentioned that binary indicators are added from the feature engineering section of the research to this list because they play a significant role in improving the models accuracy. To better explain binary indicators, consider the above indicators of EMA(10), which is a binary, 0 and 1, indicator. If the price is above the ten exponential moving average, it will be 1; if it is below it, it is assigned 0. The descriptive statistics of Bitcoin price data in the mentioned time interval are as follows:

Model implementation method

RNN and LSTM Components

It should be noted that the specified hyperparameters were obtained through Grid Search and the dropout layer was used for each layer to reduce overfitting.

Table 2: Descriptive statistics of Bitcoin price data

row	Indicator name	Indicator value
1	Price date number	39484
2	The largest data	68639
3	The smallest data	3139
4	Median	10142
5	Mean	20384
6	Mode	6399
7	SD	17876
8	Skewness	0.97
9	Kurtosis	-0.56

Table 3: Models Structures

<table>
<tr><th>Row</th><th>Model Name</th><th>Layers Information</th><th>Unit Number</th><th>Fraction of the input units to drop</th><th>Activation Function</th></tr>
<tr><td rowspan="3">1</td><td rowspan="3">RNN</td><td>Layer 1 (input Layer)</td><td>64</td><td>-</td><td>hyperbolic tangent (tanh)</td></tr>
<tr><td>Layer 2</td><td>64</td><td>-</td><td>hyperbolic tangent (tanh)</td></tr>
<tr><td>Output layer (Dense)</td><td>2 (Binary)</td><td>-</td><td>sigmoid</td></tr>
<tr><td rowspan="5">2</td><td rowspan="5">LSTM</td><td>Layer 1 (input Layer)</td><td>64</td><td>-</td><td>hyperbolic tangent (tanh)</td></tr>
<tr><td>Dropout</td><td>-</td><td>0.2</td><td>-</td></tr>
<tr><td>Layer 2</td><td>64</td><td>-</td><td>hyperbolic tangent (tanh)</td></tr>
<tr><td>Dropout</td><td>-</td><td>0.2</td><td>-</td></tr>
<tr><td>Output layer (Dense)</td><td>2 (Binary)</td><td>-</td><td>softmax</td></tr>
</table>

Research stages

Implementation Models

As mentioned in the previous section, the data related to Bitcoin price from 2018 to 2022 are used in a 1-hour time frame after the cleaning process. Figure 2 shows the Bitcoin price from 2018 to the end of the first half of 2022 in a linear manner.

In Figure 3, for the last 100 data of the Bitcoin chart, the bottom and top prices

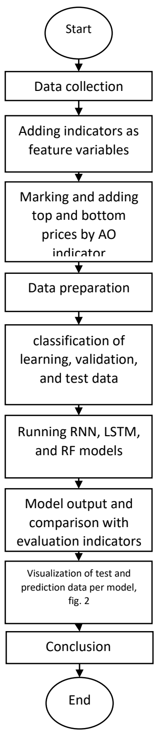
Table 4: RNN and LSTM Learning and compile arguments

Model Name	Stop Learning index			Compile models index		
	Monitor	patience	verbose	optimizer	loss	metrics
RNN	loss	3	1	adam	binary_ crossentropy	['accuracy']
LSTM	loss	3	1	adam	sparse_ categorical_ crossentropy	['accuracy']

Table 5: Random Forest Structures

Row	Model Name	Criterion	Number of trees in the forest	maximum depth of the tree	Number of jobs to run in parallel	Minimum number of samples required to be at a leaf node
1	Random Forest	gini	20	20	using all processors	2

Figure 1: Research stages



are specified, which are given as the target data for the machine learning models:

The number of data is 39484. Of all these data, 20663 (52%) show the top, and

Figure 2: Line graph of Bitcoin price in dollars

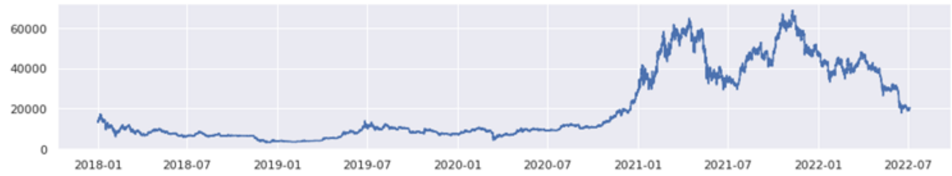
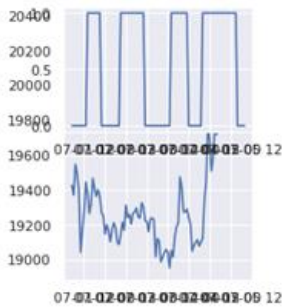
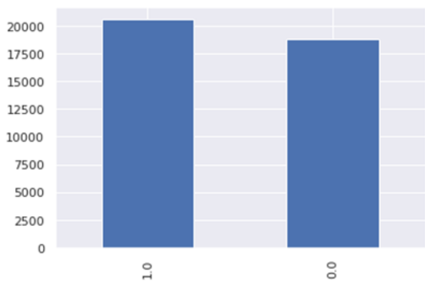


Figure 3: Bitcoin prices, together with the bottom and top situation for the last 100 data



18,821 (47%) show the bottom data. This refers to the relative balance between the number of top and bottom data.

Figure 4: The number of the top (1) and bottom (0) situations in Bitcoin price data



Model outputs:

In this section, the output of each model is presented and analyzed independently with the accuracy, precision, recall, and F1 indicators. In the next stage of the research, the performance of the models is compared with the specified indices. By accuracy, it is meant the result of dividing the correctly-predicted cases into all cases. The precision index is the result of dividing the positive cases recognized as true by the positive cases recognized as true or false. Finally, the recall index is the positive cases recognized as true divided by the sum of the positive data recognized as true and the negative cases recognized as false. The F1 score, calculated as follows, is an average of precision and recall indices:

Table 6: Confusion matrix

		Prediction by algorithm	
Real tag		No	Yes
	Yes	True Positive (TP)	False Negative (FN)
	No	False Positive (FP)	True Negative (TN)

$$Accuracy = \frac{(TP + TN)}{(TP + TN + FP + FN)}$$

$$precision = \frac{TP}{(TP + FP)}$$

$$Recall = \frac{TP}{(TP + FN)}$$

$$f1score = \frac{2 \times (precision \times Recall)}{precision + Recall}$$

First, the output and evaluation indicators of RNN are provided. As indicated in Tables 3 and 4 and explained in the methodology section of the RNN network structure, the output of the layers and parameters is as follows:

As shown in Table 7, the accuracy of RNN in predicting the top and bottom prices is 82.2%. Its precision in recognizing tops is 81%, and the recall index of this model, which indicates the values of real tops as recognized by the model, is 83%. These criteria for bottom prices are 84% and 81%, respectively. These values point to the appropriate performance of the model in predicting the situation (bottom or top). The indicators are presented in Table 8.

The results of the LTSM model, based on the structure presented in the methodology section, are as follows:

As shown in Table 9, the performance evaluation indicators show an accuracy of 81.56%. The precision of the LTSM model in recognizing tops is 82%, and the recall index of this model, which indicates the values of real tops as recognized

Table 7: RNN model output

Layer	Number of parameters	Output form
Simple_ RNN	4224	(64,31,None)
Simple_ RNN_ 1	8256	(64, None)
Dense	65	(1,None)
Total number of parameters	12545	
Learned parameters	12545	
Non-learned parameters	0	

Table 8: Evaluation indicators of the RNN model

RNN model accuracy 82.22%				
Target	Support	Recall	F1	Precision
0	2047	0.83	0.81	0.84
1	1902	0.82	0.83	0.81
Accuracy	3949	0.82	-	-
Macro average	3949	0.82	0.82	0.82
Average weight	3949	0.82	0.82	0.82

Table 9: LTSM model output

Layer	Number of parameters	Output form
LSTM	16896	(64,31,None)
Dropout	0	(64,31,None)
LSTM_ 1	33024	(64,31,None)
Dropout_ 1	0	(64,31,None)
LSTM_ 2	33024	(64,31,None)
Dense_ 1	130	(2, None)
Total number of parameters	83074	
Learned parameters	83074	
Non-learned parameters	2	

by the model, is 79%. These criteria for bottom prices are 81% and 84%, respectively. These values indicate the model's appropriate performance in predicting the situation (bottom or top). The indicators are presented in Table 10.

Table 10: Evaluation indicators of the LTSM model

LTSM model accuracy 81.56%				
Target	Support	F1	Recall	Precision
0	2047	0.83	0.84	0.81
1	1902	0.80	0.79	0.82
Accuracy	3949	0.82	-	-
Macro average	3949	0.82	0.81	0.82
Average weight	3949	0.82	0.81	0.82

According to Table 8, the RF model shows an accuracy of 83%. The precision of the model in recognizing tops is 82%, and its recall index, which shows the real tops recognized by the model, is 83%. These criteria for bottoms are 84% and 83%, respectively. These values show that the model has an appropriate performance in predicting the situation (top or bottom). These indicators are presented in Table 11.

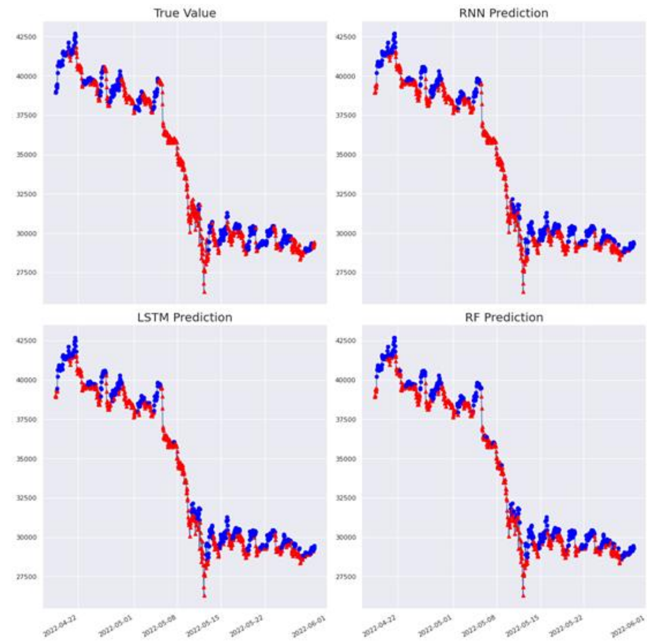
Table 11: Evaluation indicators of the RF model

RF model - accuracy 83%				
Target	Support	F1	Recall	Precision
0	2047	0.84	0.83	0.84
1	1902	0.83	0.83	0.82
Accuracy	3949	0.83	-	-
Macro average	3949	0.83	0.83	0.83
Average weight	3949	0.83	0.83	0.83

According to the values of precision, accuracy, recall, and F1 index reported in the previous tables, the performance of these three models in predicting the top and bottom prices of Bitcoin can be compared as follows. For visual simplicity of comparison, the whole 10% of the test data is not evaluated, and only the last thousand data are presented.

In graph (1), the price of Bitcoin is shown along with its top and bottom (tops in blue and bottoms in red). The upper left graph with the title True Value shows the linear diagram of the Bitcoin price along with the marking of the real top and bottom. Other graphs titled RNN Prediction, LSTM Prediction, and RF Prediction, respectively, depict the Bitcoin line graph with the marked predicted top

Figure 5: Comparing the prediction of three models concerning the top and bottom prices of Bitcoin with real values for the last thousand data



and bottom. As shown in the picture, when the price is in an ascending phase, the graph is marked with consecutive tops in blue, and when the price is in a descending (negative phase, the graph is marked with consecutive bottoms in red. The graph above shows the optimal precision of all three models in predicting Bitcoin's top and bottom prices. However, as it is clear in the picture, in some parts of the price graph, the prediction models have errors. In the following, we look for a model with better performance by collectively comparing the three models' precision, accuracy, recall, and F1 indices.

In Table 12, the performance of the models is grouped together. The F1-Score index can be suitable for comparing models because it is obtained from the combination of precision and recall indices and considers the balance. However, the important point is that we have this index separately for tops (1) and bottoms (0). Depending on which algorithmic trading system is designed based on this analytical system, the error in detecting tops and bottoms will be of different sensitivity. In Table 9, the performance of the models is compared.

As shown in Table 9, considering the F1-Score criterion, the RF model can better predict both tops and bottoms. Therefore, it can be stated that the random

Table 12: comparison of the evaluation indices of RF, RNN, and LSTM models

	Model	Top/bottom	Accuracy	precision	Recall	F1-Score
1	RNN	0	82.22%	0.84	0.81	0.83
		1		0.81	0.83	0.82
2	LSTM	0	81.56%	0.81	0.84	0.83
		1		0.82	0.79	0.80
3	RF	0	83%	0.84	0.83	0.84
	1			0.82	0.83	0.83

forest model (RF) with the F1 criterion outperforms RNN and LSTM models in predicting the condition of the top or bottom prices.

Discussion and conclusion

The previous section presented and reviewed the output and evaluation indicators independently. This section intends to answer the research questions clearly. In the previous section (data analysis), each model's accuracy, precision, recall, and F1 score were presented separately. Also, in table 9, the performance of the models was evaluated together. In general, the models presented in this research can predict with high accuracy 80% of the state of the top or bottom prices of Bitcoin. Thus, the output of these models can form the basis of an algorithmic trading system to yield a suitable output. Among these models, the RF model has a higher predictive accuracy with a higher F1 index than LSTM and RNN. In this regard, the questions of this study can be answered as follows:

1. The accuracy and precision of RF, LSTM, and RNN models are presented in Table 9.

2. The accuracy and precision of RF models are higher than the LSTM and RNN models, according to the F1 index.

3. Since all models can predict the top and bottom prices of Bitcoin with high accuracy of over 80%, the algorithmic trading actors can utilize these models' results for designing an intelligent trading system.

In the studies conducted in Iran, the accuracy and precision of the RF model in predicting the top and bottom prices were acceptable. In the present study, the accuracy of this models prediction increased to 80%, while in other studies, it was reported to be 56% (Bashiri & Paryab, 2019) and 69% (Moshari et al., 2018). The result of this study is consistent with Basak et al.s (2019) study. They, too, compared the RF model with XGBoost, ANN, SVM, and logistic regression models

and indicated that the RF model enjoyed higher accuracy.
The following can be presented as suggestions for future research:

- Providing a trading strategy based on the prediction of the RF model; that is, the prediction of this model is used and back-tested with the stop loss and take profit of a trading strategy.

- Using multi-asset data simultaneously and integration; that is, the model, instead of learning and predicting only Bitcoin data, can simultaneously learn several assets in the same market (Ethereum, Litecoin, etc.) or even several markets (gold, currency, US stock index, etc.) and then predict.

- Using a hybrid model so that it can aggregate the results of the models and then draw conclusions. Although the RF model is a superior model with the indicators, the performance of the models may be weakened or strengthened at different times and have a dynamic behavior. Therefore, adding a model as a leader model, managing the output prediction of the models, and combining and aggregating model outputs can enhance the performance and accuracy of the prediction.

- Using other indicators to detect the top and bottom prices, such as ZigZag, and comparing it with the output of the present model that used the AO indicator.

- Adding fundamental and sentimental market variables as feature variables to enhance prediction accuracy can be very helpful. Also, comparing the models and the impact of adding such data can contribute to the findings of studies.

Bibliography

[1] Aghakhani, K. & Karimi (2013). Investigating stock price prediction methods in the stock market and introducing the optimal method. Paper presented at The National Electronic Conference on New Achievements in Engineering and Basic Sciences, Ardabil, Iran.

[2] Bashiri & Paryab (2019). Bitcoin price prediction using machine learning algorithms, Applied Economics Quarterly, 10(34,35), 1-13.

[3] Tehrani & Esmaili, (2013). Investigating the impact of using important indicators of technical analysis on the short-term returns of investors in the Tehran Stock Exchange, Financial Knowledge of Securities Analysis (Financial Studies), 5(13), 21-33.

[4] Shirzor Aliabadi, Z. & Ramzanzadeh, H. (2019). Bitcoin and prediction of its future. Paper presented at the 6th International Conference on Industrial and Systems Engineering, Mashhad, Iran.

[5] Abdi, N., Delara, Ch. & Daneshjoo, P. (2018). Bitcoin price prediction using LSTM neural network. Paper presented at 9th National Conference on Computer Science and Engineering and Information Technology, Babol, Iran.

[6] Moshari M., Didekhani H., Khalili Damghani, K. & Abbasi, E. (2018). Designing a hybrid intelligent model to predict the golden points of stock prices, Investment Science Quarterly, 8(29), 45-65.

[7] Mohammadi Alamouti M., Haddadi M. & Nademi Y. (2017). Modeling and evaluation of prediction of different models of short-term memory, long-term memory, Markov switching, and hyperbolic Garch in predicting OPEC crude oil price fluctuations, Financial Engineering and Securities Management Quarterly, 9 (34), 249-272.

[8] Mousavi, S. O. (2018). Algorithmic trading technology: New solutions in Iran's stock market, Challenge Publications.

[9] Abbasi Ebrahim, Samavi Mohammad Ebrahim, Koosha Emad,(2020), Performance Evaluation Of The Technical Analysis Indicators In Comparison With The Buy And Hold Strategy In Tehran Stock Exchange Indices, ADVANCES IN MATHEMATICAL FINANCE AND APPLICATIONS , vol 5, no 3, pp 285 to 301.

[10] Basak, Suryoday & Kar, Saibal & Saha, Snehanshu & Khaidem, Luckyson & Dey, Sudeepa. (2018). Predicting the direction of stock market prices using tree-based classifiers. The North American Journal of Economics and Finance. 47. 10.1016/j.najef.2018.06.013.

[11] Asquith, p., Mikhail, M.B., Au, A. S. (2005). Information content of equity analyst reports. Journal of Financial Economics, 75, 245-282.

[12] Avramov, D., Chordia, T., Goyal, A. (2006). Liquidity and autocorrelations in individual stock returns. Journal of Finance, 61(5), 2365-2394.

[13] Basuki, K. (2019). No Title No Title. ISSN 2502-3632 (Online) ISSN 2356-0304 (Paper) Jurnal Online Internasional & Nasional Vol. 7 No.1, Januari -Juni 2019 Universitas 17 Agustus 1945 Jakarta, 53(9), 1689-1699. www.journal.uta45jakarta.ac.id

[14] Booth, A., Gerding, E., & McGroarty, F. (2014). Automated trading with performance weighted random forests and seasonality. Expert Systems with Applications, 41(8), 3651-3661. https://doi.org/10.1016/j.eswa.2013.12.009

[15] Borges, T. A., & Neves, R. F. (2020). Ensemble of machine learning algorithms for cryptocurrency investment with different data resampling methods. Applied Soft Computing Journal, 90, 106187. https://doi.org/10.1016/j.asoc.2020.106187

[16] Chopra, N., Lakonishok, J., Ritter, J. R. (1992). Measuring abnormal performance: do stocks overreact? Journal of Financial Economics, 31(2), 235-268.

[17] Ciner, C. (2019). Do industry returns predict the stock market? A reprise using the random forest. Quarterly Review of Economics and Finance, 72, 152-158. https://doi.org/10.1016/j.qref.2018.11.001

[18] Cooper, M. (1999). Filter rules based on price and volume in individual security overreaction. Review of Financial Studies, 12(4), 901-935.

[19] Da, Z., Liu, Q., Schamburg, E. (2013). A closer look at the short-term return reversal. Management Science, 60(3), 658-674.

[20] Fama, E. . (1991). Efficien capital markets. Journal of Finance, 46(5), 1575-1617.

[21] Fischer, T., & Krauss, C. (2018). Deep learning with long short-term memory networks for financial market predictions. European Journal of Operational Research, 270(2), 654-669. https://doi.org/10.1016/j.ejor.2017.11.054

[22] Creamer, German. (2004). Predicting Performance and Quantifying Corporate Governance Risk for Latin American Adrs and Banks. Proceedings of the Second IASTED International Conference on Financial Engineering and Applications.

[23] Ghosh, P., Neufeld, A., & Sahoo, J. K. (2021). Forecasting directional movements of stock prices for intraday trading using LSTM and random forests. Finance Research Letters, June 2020, 102280. https://doi.org/10.1016/j.frl.2021.102280.

[24] Hu, G., Ye, Y., Zhang, Y., & Hossain, M. S. (2019). Improved recurrent neural networks (RNN) based intelligent fund transaction model. 2019 IEEE Globecom Workshops, GC Wkshps 2019 - Proceedings. https://doi.org/10.1109/GCWkshps45667.2019.9024476

[25] Lariviere, Bart & Van den Poel, Dirk. (2005). Predicting Customer Retention and Profitability by Using Random Forests and Regression Forests Techniques. Expert Systems with Applications. 29. 472-484. 10.1016/j.eswa.2005.04.043.

[26] Jiang, G.J., Oomen, R. C. (2008). Testing for jumps when asset prices are observed with noise_ a awap variance approach. Journal of Econometrics, 144(2), 352-370.

[27] Jiang, G. J., & Zhu, K. X. (2017). Information Shocks and Short-Term Market Underreaction. Journal of Financial Economics, 124(1), 43-64. https://doi.org/10.1016/j.jfineco.2016.06.006

[28] Khare, Kaustubh & Gupta, Prafull & Attar, Vahida. (2017). Short term stock price prediction using deep learning. 482-486. 10.1109/RTEICT.2017.8256643.

[29] Kim, M. (2021). Adaptive trading system integrating machine learning and back-testing: Korean bond market case. Expert Systems with Applications, 176(December 2020), 114767. https://doi.org/10.1016/j.eswa.2021.114767

[30] Krauss, C., Do, X. A., & Huck, N. (2017). Deep neural networks, gradient-boosted trees, random forests: Statistical arbitrage on the S&P 500. European Journal of Operational Research, 259(2), 689-702. https://doi.org/10.1016/j.ejor.2016.10.031

[31] Riemann Ruiz-Cruz, (2018), Portfolio modeling for an algorithmic trading based on control theory, IFAC PapersOnline, 390-395.

[32] Robert Kissell, (2014), The science of Algorithmic Trading and Portfolio Management

[33] Lohrmann, C., & Luukka, P. (2019). Classification of intraday S&P500 returns with a Random Forest. International Journal of Forecasting, 35(1), 390-407. https://doi.org/10.1016/j.ijforecast.2018.08.004

[34] Ma, C., Zhang, J., Liu, J., Ji, L., & Gao, F. (2021). A parallel multi-module deep reinforcement learning algorithm for stock trading. Neurocomputing, 449, 290-302. https://doi.org/10.1016/j.neucom.2021.04.005

[35] Ma, Y., Han, R., & Wang, W. (2021). Portfolio optimization with return prediction using deep learning and machine learning. Expert Systems with Applications, 165(June 2020), 113973https://doi.org/10.1016/j.eswa.2020.113973

[36] Maragoudakis, Manolis & Serpanos, Dimitrios. (2010). Towards Stock Market Data Mining Using Enriched Random Forests from Textual Resources and Technical Indicators. IFIP Advances in Information and Communication Technology. 339. 278-286. 10.1007/978-3-642-16239-8_37.

[37] Moritz, Benjamin & Zimmermann, Tom. (2016). Tree-Based Conditional Portfolio Sorts: The Relation between Past and Future Stock Returns. SSRN Electronic Journal. 10.2139/ssrn.2740751.

[38] M, Hiransha & Gopalakrishnan, E. A & Menon, Vijay & Kp, Soman. (2018). NSE Stock Market Prediction Using Deep-Learning Models. Procedia Computer Science. 132. 1351-1362. 10.1016/j.procs.2018.05.050.

[39] M.S. Islam, E. Hossain,(2021) ,Foreign exchange currency rate prediction using a GRU-LSTM hybrid network, Soft Computing Letters, Volume 3,100009,ISSN 2666-2221.

[40] Nti, I. K., Adekoya, A. F., & Weyori, B. A. (2019). Random Forest Based Feature Selection of Macroeconomic Variables for Stock Market Prediction. American Journal of Applied Sciences, 16(7), 200212.https://doi.org/10.3844/ajassp.2019.200.212

[41] Paiva, F. D., Cardoso, R. T. N., Hanaoka, G. P., & Duarte, W. M. (2019). Decision-making for financial trading: A fusion approach of machine learning and portfolio selection. Expert Systems with Applications, 115, 635-655. https://doi.org/10.1016/j.eswa.2018.08.003

[42] Perry Sadorsky, 2021, A Random Forests Approach to Predicting Clean Energy Stock Prices, JRFM, 2021, vol. 14, issue 2, 1-20.

[43] Sang, C., & Di Pierro, M. (2019). Improving trading technical analysis with TensorFlow Long Short-Term Memory (LSTM) Neural Network. Journal of Finance and Data Science, 5(1), 1-11. https://doi.org/10.1016/j.jfds.2018.10.003

[44] Saud, A. S., & Shakya, S. (2020). Analysis of look back period for stock price prediction with RNN variants: A case study on banking sector of NEPSE. Procedia Computer Science, 167(2019), 788-798. https://doi.org/10.1016/j.procs.2020.03.419

[45] Selvin, S., Vinayakumar, R., Gopalakrishnan, E. A., Menon, V. K., & Soman, K. P. (2017). Stock price prediction using LSTM, RNN and CNN-sliding window model. 2017 International Conference on Advances in Computing, Communications and Informatics, ICACCI 2017, 2017-Janua, 1643-1647. https://doi.org/10.1109/ICACCI.2017.8126078

[46] Sima Siami-Namini, Akbar Siami Namin, (2018), Forecasting Economics and Financial Time Series: ARIMA vs. LSTM, Cornel University,.

[47] Speiser, J. L., Miller, M. E., Tooze, J., & Ip, E. (2019). A comparison of random forest variable selection methods for classification prediction modeling. Expert Systems with Applications, 134, 93-101. https://doi.org/10.1016/j.eswa.2019.05.028

[48] Tan, Z., Yan, Z., & Zhu, G. (2019). Stock selection with random forest: An exploitation of excess return in the Chinese stock market. Heliyon, 5(8), e02310. https://doi.org/10.1016/j.heliyon.2019.e02310

[49] Thakur, M., & Kumar, D. (2018). A hybrid financial trading support system using multi-category classifiers and random forest. Applied Soft Computing Journal, 67, 337-349. https://doi.org/10.1016/j.asoc.2018.03.006

[50] Wang, W., Li, W., Zhang, N., & Liu, K. (2020). Portfolio formation with preselection using deep learning from long-term financial data. Expert Systems with Applications, 143, 113042. https://doi.org/10.1016/j.eswa.2019.113042

[51] Yun, H., Lee, M., Kang, Y. S., & Seok, J. (2020). Portfolio management via two-stage deep learning with a joint cost. Expert Systems with Applications, 143, 113041. https://doi.org/10.1016/j.eswa.2019.113041

How to Cite: Emad Koosha[1], Mohsen Seighaly[2], Ebrahim Abbasi[3], *Measuring the Accuracy and Precision of Random Forest, Long Short-Term Memory, and Recurrent Neural Network Models in Predicting the Top and Bottom of Bitcoin price*, Journal of Mathematics and Modeling in Finance (JMMF), Vol. 2, No. 2, Pages:107–127, (2022).