Research article

A novel hybrid walk-forward ensemble optimization for time series cryptocurrency prediction

David Opeoluwa Oyewola a,*, Emmanuel Gbenga Dada b, Juliana Ngozi Ndunagu c

a Department of Mathematics and Statistics, Federal University Kashere, Gombe, Nigeria
b Department of Mathematical Sciences, Faculty of Science, University of Maiduguri, Nigeria
c Department of Computer Science, National Open University of Nigeria, Nigeria

ARTICLE INFO

Keywords:
Cryptocurrency
Blockchain
Gated recurrent unit
Walk-forward
Optimization

ABSTRACT

Cryptocurrency is an advanced digital currency that is secured by encryption, making it nearly impossible to forge or duplicate. Many cryptocurrencies are blockchain-based with decentralized networks. The prediction of cryptocurrency prices is a very difficult task because of the absence of an appropriate analytical basis to substantiate their claims. Cryptocurrencies are also dependent on several variables, such as technical advancement, internal competition, market pressure, economic concerns, security, and political considerations. This paper proposed the hybrid walk-forward ensemble optimization technique and applied it to predict the daily prices of fifteen cryptocurrencies, such as Cardano (ADA-USD), Bitcoin (BTC-USD), Dogecoin (DOGE-USD), Ethereum Classic (ETC-USD), Chainlink (LINK-USD), Litecoin (LTC-USD), NEO (NEO-USD), Tron (TRX-USD), Tether (USDT-USD), NEM (XEM-USD), Stellar (XLM-USD), Ripple (XRP-USD), and Tezos (XTZ-USD). A performance comparison of these cryptocurrencies was done using classical statistical models, machine learning algorithms, and deep learning algorithms on different cryptocurrency time series. Simulation results show that our proposed model performed better in terms of cryptocurrency prediction accuracy compared to the classical statistical model and machine and deep learning algorithms used in this paper.

1. Introduction

Machine learning (ML) and deep learning (DL) have their foundations in artificial intelligence (AI). ML is a branch of artificial intelligence, while DL is a subclass of ML. Deep learning is critical to the growth and development of AI in several ways. Using ML and DL techniques for daily data prediction yields better outcomes and aids in the understanding of some unnoticed characteristics of the dataset. The field of cryptocurrency has greatly developed to the point that it is estimated to be worth a billion dollars. Comprehending such a massive digital currency can be tough, and estimating the change in trend is critical since a change in trend might result in gains or losses for any cryptocurrency. Moreover, the number of cryptocurrencies has increased over the last few years as additional currencies have been introduced. The emergence of digital currencies in 2008, as well as the quick rise in Bitcoin values in 2017, sparked widespread condemnation in global financial and economic circles. Investors in digital currencies enjoyed tremendous profits during this period [4].

Nakamoto [2] opines that Bitcoin is the most popular cryptocurrency. Bitcoin was invented by an anonymous individual or group of individuals using the nickname, whose network of connections was inaugurated in 2009. Bitcoin is a newcomer to the currency markets, though it is officially listed as a means of exchange instead of a currency, and its price behaviour is still unclear. This presents new opportunities for investigators and financial experts to identify commonalities and contrasts with traditional financial currencies. This is when we particularly consider its very different nature in comparison to more conventional currencies or monetary systems. According to [5], which is one of the
most famous sites delivering almost real-time statistics on the many cryptocurrencies listed on global exchanges. Bitcoin’s market capitalization is predicted to be approximately $201 trillion in April 2020. About 4,000 cryptocurrencies have been in operation since January 2021 [6]. According to market capitalization, Bitcoin, Ethereum, XRP, Tether, and Bitcoin Cash are among the top five cryptocurrencies [7].

The effectiveness of ML approaches for stock market prediction has been studied in [8, 9, 10, 11, 12, 13, 14]. Findings from these studies indicate that these techniques might become useful for predicting cryptocurrency prices as well. Nevertheless, the use of machine learning algorithms in digital currency has so far been confined to the evaluation of Bitcoin prices, which has been done using random forests [15], Bayesian neural networks [16], long-short-term memory neural networks [17], data mining, and neural networks [8, 18]. These researchers were able to predict, to varying degrees, the price variations of Bitcoin and found that neural network-based algorithms produced the best results. Deep reinforcement learning has proved effective in predicting the prices of 12 cryptocurrencies [19, 20].

The digital currency market has grown into an international trend. It is especially renowned for its unpredictability and heterogeneity, garnering the curiosity of both new and experienced investors [21]. Forecasting financial time series is difficult because these series are characterized by temporary, hetero-multicollinearity problems, interruptions, aberrations, and rising multi-polynomial elements, making market movement prediction extremely difficult [22]. The complex properties of financial time series, as well as the massive amounts of data that must be analyzed in order to correctly predict financial time series, have prompted the development of more advanced methods, algorithms, and models. Recently, ML and data mining techniques, which are frequently used in financial market forecasting, have produced better results compared to simple technical or fundamental research methodologies. Machine learning approaches are capable of identifying patterns and predicting market opportunities [23].

The major contribution of this research is the application of a hybrid walk forward ensemble optimization technique for cryptocurrency prediction. The proposed ensemble technique uses advanced machine learning models as component learners, which are centred on combinations of autoregressive integrated moving average (ARIMA), holt winter’s exponential smoothing (HWES), decision tree (BAG), stochastic gradient boosting (SGB), random forest (RF), long short term memory (LSTM), gated recurrent unit (GRU), and recurrent neural network (RNN). A wide-ranging simulation analysis was carried out to evaluate the performance of the proposed model. Furthermore, the effectiveness of the predictions of each forecasting model is evaluated using mean, standard deviation, minimum, and maximum, which represents an important test of reliability for each of the models. The contributions of this work are outlined below:

1. This paper proposed the hybrid walk forward ensemble optimization technique for cryptocurrency prediction and analysis of Cardano (ADA-USD), BitcoinCash (BCH-USD), Dogecoin (DOGE-USD), Ethereum Classic (ETC-USD), Chainlink (LINK-USD), Litecoin (LTC-USD), NEO (NEO-USD), Tron (TRX-USD), Tether (USDT-USD), NEM (XEM-USD), Stellar (XLM-USD), and Ripple (XRP-USD).
3. A hybrid walk-forward ensemble optimization algorithm that can accurately predict cryptocurrencies to generate a high and considerable financial reward for investors was presented.

4. The effectiveness of the proposed model was analyzed using different performance metrics.

2. Related works

Several studies have been done in the area of predicting time series for cryptocurrencies. For example, Mudassir et al. [24] presented a time-series machine learning system for forecasting Bitcoin prices. The system uses regression models based on the learning process to forecast short- and medium-term Bitcoin price movements and pricing. With classification models that score up to 65% for the next day’s forecast and 62%–64% accuracy in the seventh–ninetieth-day forecast, these proposed models are very effective. The inaccuracy rate for daily price forecasts was 1.44%, but it ranges from 2.88% to 4.10% for seven to ninety days. The proposed models outperform the existing models in the literature. Kyriaziset et al. [25] applied GARCH models to estimate the unpredictability of cryptocurrencies during bearish markets. The authors investigated the volatility of specific cryptocurrencies and their influence on three of the most popular digital currencies, namely Bitcoin, Ethereum, and Ripple. The effect of the decreases in these three cryptocurrencies, as well as that of the DCC-GARCH on the returns of other virtual currencies, was considered using the ARCH and GARCH models. The data used for the study was between January 1 and September 16,
2018. The findings show that the major digital currencies are also adversely affected in difficult times.

Gatabazi, Mba, and Pindza [26] used the fractional Lotka-Volterra model (FGLVM) to model the transaction counts of Bitcoin, Ripple, and Bitcoin. Findings show that the proposed system is both chaotic and dynamic. Moreover, despite the disorder shown by exposure to Lyapunov, the three-dimensional Lotka-Volterra system showed parabolic patterns. The performance of the proposed model was good. Zbikowski in [27] applied support vector machines (SVM) with box theory and volume weighting to predict price direction in the Bitcoin market. The intention was to generate trading strategies utilizing some set of technical indicators computed from Bitcoin's historic data as input. A simple B&H strategy was employed as a base learner, which yielded an ROI of 4.86% and was exceeded by the BOX-SVM, which produced an ROI of 10.6%, while the VW-SVM generated an ROI of 33.5%. The simulation results showed that the proposed system performed better than the other models compared in the study.

A similar effort was made by Mallqui and Fernandes [28] to predict the daily Bitcoin price direction. Aside from the OHLC values and volume, the researchers conducted some tests by inserting other blockchain indicators and a few others external indicators. Many feature extraction strategies were employed, with the OHLC values and volume always being the most important attributes. The authors tested different ensemble and individual learning models. However, the SVM and an ensemble of recurrent neural networks and decision tree classifiers produced the best performance.

### Table 6. Classical linear statistical cryptocurrency time series result.

| Cryptocurrency | Classical | MSE    | RMSE   | MAE    |
|----------------|-----------|--------|--------|--------|
| ADA-USD        | ARIMA     | 1.2471 | 1.1167 | 0.9986 |
|                | SARIMA    | 1.1231 | 1.0597 | 0.9476 |
|                | HWES      | 1.1248 | 1.0605 | 0.9484 |
| BCH-USD        | ARIMA     | 326341.4374 | 571.2630 | 484.6991 |
|                | SARIMA    | 152255.8246 | 390.1997 | 304.1406 |
|                | HWES      | 155190.1468 | 393.9418 | 308.8400 |
| BNB-USD        | ARIMA     | 98626.6265 | 314.0487 | 256.2038 |
|                | SARIMA    | 100358.0655 | 316.9264 | 258.2038 |
|                | HWES      | 100442.3864 | 316.9264 | 258.2038 |
| BTC-USD        | ARIMA     | 33566503.0620 | 18318.4497 | 15345.9441 |
|                | SARIMA    | 383177290.0518 | 19574.9148 | 16828.9033 |
|                | HWES      | 376282825.9250 | 19397.4953 | 19397.4953 |

### Table 7. Machine learning cryptocurrency time series result.

| Cryptocurrency | Machine Learning | MSE    | RMSE   | MAE    |
|----------------|------------------|--------|--------|--------|
| ADA-USD        | BAG              | 0.0119 | 0.1091 | 0.0765 |
|                | SGB              | 0.0104 | 0.1021 | 0.0683 |
|                | RF               | 0.0125 | 0.1118 | 0.0793 |
| BCH-USD        | BAG              | 5649.986 | 75.1663 | 44.3070 |
|                | SGB              | 5587.885 | 74.7521 | 42.3190 |
|                | RF               | 6350.812 | 79.6919 | 46.8026 |
| BNB-USD        | BAG              | 1475.144 | 38.4076 | 26.4980 |
|                | SGB              | 1313.134 | 36.2371 | 22.6948 |
|                | RF               | 1716.968 | 41.4363 | 28.9523 |
| BTC-USD        | BAG              | 4967302 | 2228.744 | 1687.474 |
|                | SGB              | 4888106 | 2210.906 | 1687.474 |
|                | RF               | 8006971 | 2829.659 | 2137.399 |

2018. The findings show that the major digital currencies are also adversely affected in difficult times.
models used include stochastic differential equations (SDE) and continuous-time stochastic process techniques. The continuous-time performance of AA was evaluated using machine learning and mimics the auditory system in a way similar to that of the human ear. The auditory algorithm (AA) for stock market prediction. The technique determined accuracy, even though the size of the dataset used was small. The use of ANNs did not result in any substantial improvement in prediction results. It was observed that the proposed system has an accuracy of less than 55%. It was observed that the prediction of the next time step in binary form. The data with a sampling frequency ranging from daily to minute. The goal is to forecast the price direction of the next time step in binary form. The technique was validated by experimental results. The technique can trade Bitcoin in a geometric brownian motion (GBM). The findings demonstrated that the overall performance of AA is better than that of other models studied since it dramatically decreased forecast error to the smallest possible level.

Akyildirim et al. [29] applied SVM, linear regression (LR), RF, and artificial neural network (ANN), as well as historical prices and technical indicators, to predict some of the most popular cryptocurrencies using data with a sampling frequency ranging from daily to minute. The goal is to forecast the price direction of the next time step in binary form. The proposed system has an accuracy of less than 55%. It was observed that the use of ANNs did not result in any substantial improvement in prediction accuracy, even though the size of the dataset used was small.

Oyewola et al. [30] presented a nature-inspired method called the auditory algorithm (AA) for stock market prediction. The technique mimics the auditory system in a way similar to that of the human ear. The performance of AA was evaluated using machine learning and continuous-time stochastic process techniques. The continuous-time models used include stochastic differential equations (SDE) and geometric brownian motion (GBM). The findings demonstrated that the overall performance of AA is better than that of other models studied since it dramatically decreased forecast error to the smallest possible level.

Liu et al. [41] proposed deep reinforcement learning and proximal policy optimization (PPO) models for automatic Bitcoin trading. It draws a comparison among high-performing machine learning-based models for static price predictions such as SVM, multi-layer perceptron (MLP), LSTM, temporal convolutional network (TCN), and transformer. Simulation results indicated that LSTM performs better than all the other ML models compared in the work. The authors created an autonomous trading scheme using PPO and LSTM based on the policy. The superiority of the proposed model over other customary trading approaches was validated by experimental results. The technique can trade Bitcoin in a
virtual environment with symmetric data and achieve a 31.67 percent higher yield than the optimum benchmark, outperforming it by 12.75 percent. The proposed model can generate higher returns during both periods of price fluctuations and sharp rises, which paved the way for research into developing a single deep learning-based cryptocurrency trading tactic. Envisioning the trading process shows how the model manages and controls increased transactions, providing stimulus and demonstrating that it can be extended to other credit derivatives. Livieris et al. [42] proposed ensemble learning models for cryptocurrency forecasting using hourly prices. In the proposed model, deep learning was combined with ensemble-averaging, bagging, and stacking. The authors combined ensemble models with deep learning models such as LSTM, bi-directional LSTM, and convolutional layers. The ensemble models’ performances were evaluated, and experimental analysis showed that ensemble learning and deep learning can be mutually important for creating powerful, steady, and dependable forecasting models. The summary of related works is presented in Table 1.

Below is a list of abbreviations/notations and their meanings in Table 2.

3. Methodology

This section discusses the dataset used for our study and the different techniques used for the predictions of cryptocurrencies under study.

3.1. Description of dataset

The paper explores hybrid walk-forward optimization of cryptocurrencies using classical statistical, machine learning, and deep learning models. The cryptocurrencies used in the analysis are Cardano (ADA-USD), BitcoinCash (BCH-USD), BinanceCoin (BNB-USD), Bitcoin (BTC-USD), Dogecoin (DOGE-USD), Ethereum Classic (ETC-USD), Chainlink (LINK-USD), Litecoin (LTC-USD), NEO (NEO-USD), Tron (TRX-USD), Tether (USDT-USD), NEM (XEM-USD), Stellar (XLM-USD), Ripple (XRP-USD) and Tezos (XTZ-USD). The data on cryptocurrencies was collected from Yahoo Finance [31], from January 01, 2018 to June 30, 2021, daily. The cryptocurrency data accounts for 1277 entries for each of the currencies, with a total of 19,155 observations for all cryptocurrencies.
3.2 Classical statistical model, machine and deep learning techniques

3.2.1 Auto regressive integrated moving average (ARIMA)

An autoregressive integrated moving average (ARIMA) is a statistical analysis model that forecasts future trends based on historical data [32]. ARIMA smoothes time series data using lagged moving averages and is composed of three components: autoregressive (AR), integrated (I), and moving average (MA). Autoregressive (AR) models depict a dynamic variable that regresses on its own lags or previous values, whereas integrated (I) models depict the difference between raw observations to allow the time series to become stable. The moving average (MA) takes into account the relationship between an observation and the residual error from a moving average model applied to delayed observations. ARIMA requires three hyper-parameters for the trend, which are (a = autoregressive order) (i = differencing order), and (v = moving average order). ARIMA models can be represented mathematically as depicted in Eq. (1):

\[ r_t = I + \gamma_1 r_{t-1} + \gamma_2 r_{t-2} + \ldots + \gamma_a r_{t-a} + \theta_1 \epsilon_{t-1} + \theta_2 \epsilon_{t-2} + \ldots + \theta_v \epsilon_{t-v} \]

(1)

Where \( r_t \) is regressed at time \( t \), \( \gamma \) are the coefficients, \( a \) is the autoregressive order, \( v \) is the moving average order, \( \theta \) is the weighted moving average, and \( \epsilon_t \) is the error at time \( t \).

3.2.2 Seasonal autoregressive integrated moving-average (SARIMA)

The seasonal autoregressive integrated moving average (SARIMA) is an extension of the autoregressive integrated moving average (ARIMA) that specifically accepts single-time series data with a seasonal component [33]. In the seasonal component of the series, SARIMA adds three new hyper-parameters for auto-regression (AR), differencing (I), and a moving average (MA), and an additional seasonal parameter such as (P = seasonal autoregressive order) (D = seasonal difference order) (Q = seasonal moving average order), and \( m \) is the number of time-steps of a
Figure 4. Train (blue) and Test (red) set of Real-Time BCH-USD.

Figure 5. Train (blue) and Test (red) set of Real-Time BNB-USD.

Figure 6. Train (blue) and Test (red) set of Real-Time BTC-USD.
Figure 7. Train (blue) and Test (red) set of Real-Time DOGE-USD.

Figure 8. Train (blue) and Test (red) set of Real-Time ETC-USD.

Figure 9. Train (blue) and Test (red) set of Real-Time LINK-USD.
Figure 10. Train (blue) and Test (red) set of Real-Time LTC-USD.

Figure 11. Train (blue) and Test (red) set of Real-Time NEO-USD.

Figure 12. Train (blue) and Test (red) set of Real-Time TRX-USD.
seasonal period, respectively. The SARIMA mathematical equation is represented in (2):

$$\phi(B^s) = \theta(B^s) a_t$$

(2)

Where $\phi$ is the Box-cox power transformation, $s$ is the number of seasons per year, $B$ is the backward shift operator, $D$ is the times to produce a series, $\phi(B^s)$ is the seasonal autoregressive (AR) of order $P$, $\theta(B^s)$ is the seasonal moving average (MA) of order $Q$, $d$ is the order of the non-seasonal differencing parameter, $a_t$ is the identically independently distributed (IID) with a mean of zero and variance of $\sigma_a^2$.

3.2.3. Holt winter’s exponential smoothing (HWES)

Holt winter exponential smoothing (HWES) is employed in [34] for predicting time series data that shows both trends and variations in seasons. HWES models are also known as “triple exponential smoothing technique” models because they take trends and seasonality into account as an exponentially weighted linear function of data from previous phases. The mathematical equations are as shown in (3), (4), 5 and (6):

$$\hat{y}_t = l_t + h b_t + s_t$$

(3)

$$l_t = \alpha(y_t - s_{t-m}) + (1 - \alpha)(l_{t-1} + b_{t-1})$$

(4)

$$b_t = \beta(l_t - l_{t-1}) + (1 - \beta)b_{t-1}$$

(5)

$$s_t = \gamma(y_t - l_{t-1} - b_{t-1}) + (1 - \gamma)s_{t-1}$$

(6)

Where $l_t$ is the level at time $t$, $b_t$ is the trend at time $t$, $s_t$ is the seasonal component at time $t$ with corresponding smoothing parameters $\alpha, \beta, \gamma$, and $m$ is the daily frequency of the seasonality.

3.2.4. Decision tree (BAG)

A decision bagging tree (BAG) is a statistical model for covariate-based outcome prediction. The model suggests a prediction rule that defines unwanted sub-sets of data, for example, population sub-sets that are hierarchically constructed by a series of binary data divisions. A tree can be used to represent the hierarchical binary partition set. In each subgroup, the projected result is determined by the average of the individual results within the subset. The goal is to create a prediction rule that minimizes loss functions and also quantifies the difference between predicted and actual values [35].

![Figure 13. Train (blue) and Test (red) set of Real-Time USDT-USD.](image1)

![Figure 14. Train (blue) and Test (red) set of Real-Time XEM-USD.](image2)
Figure 15. Train (blue) and Test (red) set of Real-Time XLM-USD.

Figure 16. Train (blue) and Test (red) set of Real-Time XRP-USD.

Figure 17. Train (blue) and Test (red) set of Real-Time XTZ-USD.
### 3.2.5. Stochastic gradient boosting (SGB)

Stochastic gradient boosting (SGB) is an ensemble learning method that combines boosting with decision-making, such as a decision tree, and predicts by weighing together all the trees. The SGB is created along the direction of gradient descent from the previous tree loss function. SGB’s main objective is to minimize this loss function between the regression function and the actual function by training the regression function [36]. SGB mathematical equations are shown in (7) and (8):

\[
Y = \min_{x_k} \sum_{i} y_k R_{m-1}(x_k) + y
\]  
\[
\rho_k = -y_k \log p_k(x)
\]

Where \(x_k\) is the input variable, \(k\) is the number of K-trees each with the terminal nodes at iteration \(m\), and \(R\) is the regression function.

### 3.2.6. Random forest (RF)

Random forest (RF) is an example of an ensemble machine learning technique. RF builds several distinct decision trees during training. Predictions from the entire tree are combined to attain the ultimate prediction. RF works by randomly picking features that increase prediction accuracy and result in better efficiency. The RF does not only retain the advantages of the trees, but it generally produces better results than a decision tree [37]. For high-dimensional data modeling, the RF can effectively manage missing values and handle continuous, categorical, and binary data. The mathematical equation for RF is given in Eq. (9):

\[
Y = \frac{1}{i} \sum_{i} b \langle x_1, x_2, ..., x_p \rangle
\]

Where \(x_p\) is the feature vector of input values, \(p\) is the dimension property of the available vector for the base learners, \(b_i\) is the base learners at iteration \(i\).

### 3.2.7. Long short term memory (LSTM)

The LSTM is a type of recurrent neural network (RNN) that has the ability to manage long-term dependencies. This enhances the ability of the LSTM to learn from experience. The effectiveness of LSTM becomes more pronounced when there are very lengthy and unspecified delays between data [38]. An LSTM network is comprised of three gates, which are the input gate, the output gate, and the forget gate. These gates help the network arbitrarily retain a value for a lengthy period. One of the key benefits of LSTM networks is their ability to solve the vanishing gradient problem, which makes network training problematic for lengthy strings of words or integers. Gradients are utilized to update RNN parameters and to represent long word or integer sequences; however, as the gradients get reduced, network training becomes practically impossible. This drawback is solved by LSTM networks, which also enable the detection of long-term connections between words or numbers in sequences with great spatial separation. The mathematical equations are expressed in (10), (11), (12), (13) and (14):

\[
e_t = f_t e_{t-1} + i_t \sigma_t
\]
\[
\bar{e}_t = o_t \tanh e_t
\]
\[
f_t = \beta \langle W_e x_t + h_{t-1} + e_{t-1} \rangle
\]
\[
i_t = \beta \langle W_i x_t + h_{t-1} + e_{t-1} \rangle
\]
\[
h_t = o_t \tanh \bar{e}_t
\]

Where \(e_t\) is the memory at time \(t\), \(\sigma\) is the new memory at time \(t\), \(o_t\) output gate at time \(t\), \(f_t\) is the forget gate at time \(t\) and \(h_t\) is the activation function.
3.2.8. Gated recurrent unit (GRU)

The gated recurrent unit (GRU) is a significantly simpler variant of the LSTM. The forget and input gates merge into one called the update gate and include an extra gate termed the reset gate. The final model is simpler and has become more popular than the basic LSTM versions. However, a gated recurrent unit such as the LSTM modulates data inside the unit without a distinct memory cell [39]. The GRU activation function at time \( t \) is a linear interpolation between the prior activation function and the activation function of the candidate. The mathematical equations for GRU are presented in (15), (16), (17), and (18):

\[
\begin{align*}
    u_t &= \gamma(W_{ux} + h_{t-1}) \\
    h_t &= (1 - u_t)h_{t-1} + u_t \overline{h}_t \\
    \overline{h}_t &= \tanh(W_x h + c_t) \\
    c_t &= \gamma(W_x + ch_{t-1})
\end{align*}
\]

Where \( u_t \) is the update gate, \( c \) is the active reset gate, \( h_t \) is the activation function and \( \overline{h}_t \) is the candidate activation function.

3.2.9. Recurrent neural network (RNN)

The RNN is an artificial neural network that employs sequential data or time series data. RNNs use training data to learn, just like feedforward and convolutional neural networks (CNNs) do. They stand out due to their “memory,” which allows them to affect the present input and output by using data from previous inputs. Unlike traditional deep neural networks, which assume that inputs and outputs are unconnected, the outputs of recurrent neural networks are dependent on the previous components in the sequence. The RNN uses a prior step that might influence the choice at the present moment. RNN has two sources of input, such as the current one and the recent past, that are combined with the determination of a reaction to a new input [40]. The mathematical equation for RNN is presented in (19):

\[
d_t = \tanh(W_{dh} h_{t-1} + x_t)
\]

Where \( d_t \) is the hidden state, \( d_{t-1} \) is the previous hidden state and \( x_t \) is the input variable.

3.2.10. Performance evaluation

The accuracy test of the fifteen selected cryptocurrencies is evaluated using:

| Algorithms | Parameters of classical statistical model, machine and deep learning models. |
|------------|--------------------------------------------------------------------------------|
| ARIMA      | Seasonal_periods 7, Initialization_method ‘Known’, Initial_level ‘estimated’, Trend ‘add’, Seasonal ‘add’, Smoothing_level 0.4, Smoothing_shape 0.2, Smoothing_seasonal 0.01, Order auto_arima |
| SARIMA     | Seasonal_periods 7, Initialization_method ‘Known’, Initial_level ‘estimated’, Trend ‘add’, Seasonal ‘add’, Smoothing_level 0.4, Smoothing_shape 0.2, Smoothing_seasonal 0.01, Order auto_arima |
| HWES       | Seasonal_periods 7, Initialization_method ‘Known’, Initial_level ‘estimated’, Trend ‘add’, Seasonal ‘add’, Smoothing_level 0.4, Smoothing_shape 0.2, Smoothing_seasonal 0.01, Order auto_arima |
| BAG        | max_depth 4, min_impurity_split 1e-07, min_samples_leaf 1, min_samples_split 2, min_weight_fraction_leaf 0.0, Presort False, random_state None, Splitter Best |
| SGB        | max_depth 4, min_impurity_split 1e-07, min_samples_leaf 1, min_samples_split 2, min_weight_fraction_leaf 0.0, Presort False, random_state None, Splitter Best |
| RF         | max_depth 4, min_impurity_split 1e-07, min_samples_leaf 1, min_samples_split 2, min_weight_fraction_leaf 0.0, Presort False, random_state None, Splitter Best |
| LSTM       | Units 50, return_sequences True, Dropout 0.2, optimizer Rmsprop, loss Mean_squared_error, epochs 30, batch_size 150 |
3.2.10.1. Mean absolute error (MAE). Consider a set of real-time closing values $R_p$ and the predicted values $\tilde{R}_p$. MAE is given as follows:

$$\frac{1}{n} \sum_{n=1}^{n} |R_p - \tilde{R}_p|$$  \hspace{1cm} (20)

3.2.10.2. Root mean square error (RMSE). RMSE is given in Eq. (21):

$$\sqrt{\frac{1}{n} \sum_{n=1}^{n} (R_p - \tilde{R}_p)^2}$$  \hspace{1cm} (21)

3.2.10.3. Mean square error (MSE). MSE is given as:

$$\frac{1}{n} \sum_{n=1}^{n} (R_p - \tilde{R}_p)^2$$  \hspace{1cm} (22)

Where $n$ is the trading days.

The description of the different cryptocurrencies used in this paper is in Table 3.

Table 4 presents the results of stationarity test using augmented dickey-fuller (ADF) of cryptocurrencies.

Depicted in Table 5 is the Optimum automated ARIMA fitting for classical statistical time series.

Table 6 depicts the experimental results of classical linear statistical cryptocurrency time series.

Presented in Table 7 is the experimental results of machine learning cryptocurrency time series.

Depicted in Table 8 is the simulation results of deep learning cryptocurrency time series.

Table 9 is the statistical results of hybrid walk-forward ensemble optimization cryptocurrency time series.

3.3. Proposed hybrid walk-forward ensemble optimization

Analysis and prediction of time series are frequently considered to be among the hardest and most demanding tasks in machine learning. This research presents a new system that is an improvement on the classical statistical model, machine learning, and deep learning algorithms. The proposed method makes use of walk-forward ensemble optimization for time-series cryptocurrency prediction. The proposed system offers solutions to the problems that characterized the original low-quality time series data, thereby generating high-quality time series data to effectively train and fit classic deep learning and machine learning models. This analysis is carried out in four stages:

- Data visualization of cryptocurrency.
- Dividing the dataset into training and test sets
- The optimal model in each classical statistical model, machine learning technique, and deep learning technique is determined using performance measures such as Root Mean Square (RMSE), Mean Square Error (MSE), and Mean Absolute Error (MAE).
- Application of walk-forward ensemble optimization on the prediction results.

The forecasting of cryptocurrency prices is a very difficult task because of the absence of a satisfactory analytical proof to substantiate their claims. Cryptocurrencies are also dependent on some variables, such as technical advancement, internal competition, market pressure, economic concerns, security, and political considerations. The suggested improvement of the walk-forward ensemble would help to overcome the major problem in cryptocurrency. The algorithm can properly forecast cryptocurrency prices to produce considerable financial benefit for investors, as explained in section 3.4.
technique will be performed on the best-selected model, such as classical statistical models, machine learning algorithms, and deep learning algorithms, which will help to minimize the dispersion of a predictive model and improve the average prediction performance over any given member in the ensemble. The stacking ensemble method was utilized in this study.

It is an ensemble approach that uses a meta-regression model to integrate several regression models. The basis models utilized in this study are the best classical statistical models, machine learning models, and deep learning models produced from the dataset, and the meta-model is trained on features returned (as output) by the base models. The meta-models in consideration are the same optimal classical statistical models, machine learning models, and deep learning models that are interchanged regularly. To obtain the greatest accuracy, the meta-model aids in the discovery of features in base models. The final stage is the application of walk-forward optimization to the prediction results obtained from stage three. Then, using walk-forward optimization, each training-testing set is moved forward through the time series by specific data patterns. The comprehensive numerical experiments and statistical analysis will improve the predictive performance of the model. The experiment was conducted using a miniconda installation and all the necessary libraries such as python 3.7, pandas, numpy, scipy, sklearn, seaborn, pmdarima, keras, sklearns, and statsmodels.

4. Result and discussion

Figure 2 is the heatmap visualization of the fifteen cryptocurrencies used in this study. Heatmaps are colour-coded diagrams to visualize data. In this study, heatmaps were utilized to cross-examine cryptocurrency data in a tabular format by placing variables in the rows and columns and color-coding the cells. The x-axis represents the rows of the real-time data, while the y-axis represents the columns of the real-time cryptocurrencies. The location of the missing values in Figure 2 is in rows 840 and 1020. This shows the presence of missing values in all the fifteen cryptocurrencies' real-time data. Fifteen cryptocurrencies considered in this study were split into training and test sets. The training set is from 1st January, 2018–31st December 2020, consisting of 85% of the data, while the remaining 15% is for the test set from 1st January, 2021–30th June 2021, as shown in Figures 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, and 17.

Table 3 shows the summary statistics of each of the fifteen selected cryptocurrencies from January 2018 to June 2021. The daily mean, standard deviation, minimum, and maximum are shown. The daily mean of ADA-USD, DOGE-USD, TRX-USD, XEM-USD, XLM-USD, XRP-USD, and XTZ-USD is small compared to other selected cryptocurrencies. ADA-USD, DOGE-USD, TRX-USD, USDT-USD, XEM-USD, XLM-USD, XRP-USD, and XTZ-USD also have small volatility, which is within the range of 0–2 compared with other stocks, which indicates that the cryptocurrency prices fluctuate slowly and tend to be more stable. BTC-USD has a maximum price of $6353.45 due to high supply and demand.

Table 4 shows the stationarity test results before and after differencing using the Augmented Dickey-Fuller (ADF) test of all the selected cryptocurrencies. The ADF test consists of test statistics and critical values at 1%, 5%, and 10% confidence intervals. Stationarity means that the statistical properties of a cryptocurrency, such as its mean, variance, and covariance, do not change over time. Before differencing columns, the ADF test is higher than any of the critical values, which shows the presence of non-stationary in twelve, except in BCH-USD, USDT-USD, and XEM-USD. After differencing, ADF tests are applied to detrended values, and they all show the presence of stationarity.

Automated ARIMA fitting takes into account the Akaike Information Criterion (AIC) and Bayesian Information Criterion (BIC) values to determine optimal parameters. The lower these values, the better the model. Table 5 shows the optimal automated ARIMA fitting for all the selected classical statistical time-series, such as the autoregressive integrated moving average (ARIMA) and the seasonal autoregressive integrated moving average (SARIMA). Where P (seasonal autoregressive order), D (seasonal differencing order), Q (seasonal moving average order), and m are the frequencies of the daily cryptocurrency time series. The Automated ARIMA uses the AIC and BIC values generated by experimenting with different combinations of variables (a, i, v, P, D, Q, and m) to fit the model into the chosen cryptocurrency.

Table 6 shows the classical linear statistical cryptography time series of ARIMA, SARIMA, and HWES. The classical linear technique was used to determine the best algorithms among the three algorithms, ARIMA, SARIMA, and HWES, which can effectively predict cryptocurrency datasets, while Table 10 consists of all the parameters considered in all the techniques to obtain the best accuracy result. The overall performance of every one of the cryptocurrencies utilized in this study was reported by MASE, RAE, and MSLE. In eight out of fifteen selected cryptocurrencies, HWES performs excellently better than both SARIMA and ARIMA. This shows that Auto ARIMA is unable to select the best trends and seasonal parameters when predicting cryptocurrencies’ time series. The HWES technique considers the average as well as trends and seasonality.

Table 7 reports the machine learning cryptocurrency time series of decision tree bagging (BAG), stochastic gradient boosting (SGB), and random forest (RF). The three machine learning techniques were utilized to determine the best algorithms among the three algorithms, such as BAG, SGB, and RF. In all the fifteen selected cryptocurrencies, SGB performs excellently, better than both BAG and RF. The deep learning results of the performance measure are also shown in Table 8. The deep learning models chosen are: long short-term memory (LSTM), gated recurrent unit (GRU), and recurrent neural network (RNN). GRU performs excellently in all fifteen selected cryptocurrencies.

Figure 18 shows the comparison of the classical statistical model, machine learning, and deep learning algorithms used in this study by summing up each selected cryptocurrency in the classical statistical model, machine learning model, and deep learning model. Deep learning has the fewest errors, followed by machine learning. Classical linear models perform woefully, and it shows they are not suitable for predicting cryptocurrency. Presented in Table 9 is the result of the Hybrid Walk-Forward Ensemble Optimization Cryptocurrency Time Series. In all the fifteen selected cryptocurrencies, WGRU performs excellently, but WHWES and WSGB performed woefully. This shows that walk-forward with the ensemble can help GRU perform excellently when predicting cryptocurrency.

The computational time of classical statistical models, machine learning, deep learning, and hybrid walk-forward ensemble optimization is shown in Table 10. The first column consists of all the algorithms used in this research, while the second column is the mean in sec/loop and the third column is the standard deviation. In two of the three classical statistical learning methods, such as ARIMA and SARIMA, the length of time required to perform a computational process is very high. It may be due to the automated ARIMA fitting of both the ARIMA and SARIMA models. In machine learning algorithms, the length of time it takes for SGB to perform a computational process is much higher than that of BAG and RF. Moreover, in deep learning, LSTM and GRU have much higher computational processes than RNN. The computational process of WHWES, WGRU, and WSGB is slightly increased, but it is very minimal compared to LSTM and GRU. The parameters used for the classical statistical models, machine learning models, and deep learning models considered in this research are shown in Table 11.

5. Conclusion

One of the foundational tools of data science is time series forecasting. It is one of the most extensively utilized analytic tools in businesses and organizations. All businesses want to plan for the future. As a result, time series forecasting serves as a linchpin for looking into the most likely future and making appropriate plans. Time-series forecasting, like any other data science approach, is comprised of a variety of techniques and methods. The hybrid walk-forward ensemble optimization model for
time series forecasting is proposed in this study. The proposed technique takes care of missing values in cryptocurrency data, which are caused by a variety of reasons, including equipment failures, changes in monitor placement, periodic maintenance, and human mistakes. It also solved the problem of bias, which is often caused by disparities between observed and unobserved data in an incomplete dataset. The performance of our model was encouraging, and to the best of our knowledge, no research work has been published in the literature on the regression of real-time cryptocurrency using hybrid walk-forward ensemble optimization with distinct phases. The techniques investigated in this paper are automated ARIMA, the Augmented Dickey-Fuller test, classical statistical model, and machine and deep learning algorithms. The proposed method outperforms all the aforementioned techniques. One of the limitations of this work is the inability to obtain a large dataset for cryptocurrency. In the future, experiments will be conducted using other machine learning models, such as the Gaussian process and cubist, to confirm the strength of hybrid walk-forward ensemble optimization. Moreover, this model will be integrated into the stock market, cryptocurrency, or any time series to be used for real-time monitoring and forecasting.

Declarations

Author contribution statement

David Opeoluwa Oyewola: Conceived and designed the experiments; Performed the experiments; Analyzed and interpreted the data; Contributed reagents, materials, analysis tools or data; Wrote the paper.

Emmanuel Gbenga Dada: Conceived and designed the experiments; Analyzed and interpreted the data; Contributed reagents, materials, analysis tools or data; Wrote the paper.

Juliana Ngozi Ndunagu: Performed the experiments; Wrote the paper.

Funding statement

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

Data availability statement

Data will be made available on request.

Declaration of interests statement

The authors declare no conflict of interest.

Additional information

No additional information is available for this paper.

References

[1] Y. Indukar, Time series analysis of cryptocurrencies using deep learning & Npmsheet, in: 2021 International Conference on Emerging Smart Computing And Informatics (ESC), IEEE, 2021, March, pp. 306–311.
[2] L. Catania, S. Grassi, Modelling cryptocurrency financial time-series, 2017. Available at SSRN 3028486.
[3] L. Catania, S. Grassi, F. Ravazzolo, Predicting the volatility of cryptocurrency time-series, in: Mathematical and Statistical Methods for Actuarial Sciences and Finance, Springer, Cham, 2018, pp. 203–207.
[4] Y. Ma, F. Ahmad, M. Liu, Z. Wang, Portfolio optimization in the era of digital financialization using cryptocurrencies, Technol. Forecast. Soc. Change 161 (2020), 120965.
[5] S. Nakamoto, Bitcoin: a peer-to-peer electronic cash system, Decentraliz. Bus. Rev. (2008), 21260.
[6] R. Bagshaw, Top 10 cryptocurrencies by market capitalization, 2021. Available at, https://www.yahoons.com/news/top-10-cryptocurrencies-market-capitalization-160046487.html. (Accessed 6 July 2021).
[7] L. Conway, The 10 Most Important Cryptocurrencies Other than Bitcoin, 2021. Retrieved from, https://www.investopedia.com/tech/most-important-cryptocurrencies-other-than-bitcoin/
[8] D. Enke, S. Thawromwong, The use of data mining and neural networks for forecasting stock market returns, Expert Syst. Appl. 29 (4) (2005) 927–940.
[9] W. Huang, Y. Nakamori, S.Y. Wang, Forecasting stock market movement direction with support vector machine, Comput. Oper. Res. 32 (10) (2005) 2513–2522.
[10] P. Ou, H. Wang, Prediction of stock market index movement by ten data mining techniques, Mod. Appl. Sci. 3 (12) (2009) 28–42.
[11] M. Gavrilov, D. Anguelov, P. Indyk, R. Motwani, Mining the stock market (extended abstract) which measure is best?, in: Proceedings of the Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2000, August, pp. 487–496.
[12] K.S. Kannan, P.S. Sekar, M.M. Sathik, P. Arumugam, Financial stock market forecast using data mining techniques, in: Proceedings of the International Multiconference of Engineers and computer scientists 2010, p. 4. March.
[13] A.F. Sheta, S.E.M. Ahmed, H. Faris, A comparison between regression, artificial neural networks and support vector machines for predicting stock market index, Soft Comput. 7 (8) (2015) 2.
[14] A. P. Chang, C.H. Liu, C.Y. Fan, J.L. Lin, C.M. Lai, An ensemble of neural networks for stock trading decision making, in: International Conference on Intelligent Computing, Springer, Berlin, Heidelberg, 2009, September, pp. 1–10.
[15] I. Madan, S. Sahaja, A. Zhao, Automated bitcoin trading via Machine Learning algorithms, 2015. URL20.
[16] H. Jiang, J. Lee, An empirical study on modelling and prediction of bitcoin prices with bayesian neural networks based on blockchain information, IEEE Access 6 (2017) 5427–5437.
[17] S. McNally, J. Roche, S. Caton, Predicting the price of bitcoin using Machine Learning, in: 2018 26th Euromicro International Conference on Parallel, Distributed and Network-Based Processing (PDP), IEEE, 2018, March, pp. 399–343.
[18] K. Hegazy, S. Mumford, Comparative Automated Bitcoin Trading Strategies, CS229 Project, 2016, p. 27.
[19] A.G. Shilling, Market timing: better than a buy-and-hold strategy, Financ. Anal. J. 48 (2) (1992) 46–50.
[20] Z. Jiang, J. Liang, Cryptocurrency portfolio management with deep reinforcement learning, in: 2017 Intelligent Systems Conference (InprintS), IEEE, 2017, September, pp. 905–913.
[21] B. Hileman, M. Rauchs, Global cryptocurrency benchmarking study, Cambridge Centre Alterm. Finan. 33 (2017) 33–113.
[22] E. Hadavandi, H. Shavandi, A. Ghanbari, Integration of genetic fuzzy systems and artificial neural networks for stock price forecasting, Knowl. Base Syst. 23 (8) (2010) 800–808.
[23] J. Nadkarni, R.F. Neves, Combining NeuroEvolution and principal component analysis to trade in the financial markets, Expert Syst. Appl. 103 (2018) 184–195.
[24] M. Mudassir, S. Bennbaia, D. Unal, M. Hammadeh, Time-series forecasting of Bitcoin prices using high-dimensional features: a Machine Learning approach, Neural Comput. Appl. (2020) 1–15.
[25] N.A. Kyriazis, A survey on empirical findings about spillovers in cryptocurrency markets, J. Risk Financ. Manag. 12 (4) (2019) 170.
[26] P. Gatazzi, J.C. Mba, E. Piindza, Fractional grey Lotka-Volterra models with application to cryptocurrencies adoption, Chaos: Interdisc. J. Nonlin. Sci. 29 (7) (2019), 073116.
[27] K. Zbikowski, Using volume weighted support vector machines with walk forward testing and feature selection for the purpose of creating stock trading strategy, Expert Syst. Appl. 42 (4) (2015) 1797–1805.
[28] D.C. Malqui, R.A. Fernandes, Predicting the direction, maximum, minimum and closing prices of daily Bitcoin exchange rate using Machine Learning techniques, Appl. Soft Comput. 75 (2019) 596–606.
[29] E. Aykildirim, A. Goncu, A. Sensoy, Prediction of cryptocurrency returns using Machine Learning, Ann. Oper. Res. 297 (1) (2021) 5–36.
[30] D.O. Oyewola, A. Ibrahim, J.A. Kwanamu, E.G. Dada, A new auditory algorithm in integrated moving average and quantile regression for daily food sales forecasting, FUDMA J. Sci. (FJS) 4 (2) (2020) 371.
[31] D.O. Oyewola, A. Ibrahim, J.A. Kwanamu, E.G. Dada, A new auditory algorithm in integrated moving average and quantile regression for daily food sales forecasting, FUDMA J. Sci. (FJS) 4 (2) (2020) 371.
D.O. Oyewola, K.A. Al-Mustapha, E.G. Dada, O.A. Kennedy, Stock market movement direction with ensemble Deep Learning Network, J. Niger. Assoc. Math. Phys. 53 (2019) 103–116.

F.R. Liu, M.Y. Ren, J.D. Zhai, G.Q. Sui, X.Y. Zhang, X.Y. Bing, Y.L. Liu, Bitcoin transaction strategy construction based on deep reinforcement learning, in: 2021 IEEE 2nd International Conference on Big Data, Artificial Intelligence and Internet of Things Engineering (ICBAIE), IEEE, 2021, pp. 180–183.

I.E. Livieris, E. Pintelas, S. Stavrouliannia, P. Pintelas, Ensemble Deep Learning models for forecasting cryptocurrency time-series, Algorithms 13 (5) (2020) 121.