Deep Learning Based for Cryptocurrency Assistive System

Muhammad Zakhwan Mohamed Rafik[1*], Noraisyah Mohamed Shah[1] ,Nor Azizah Hitam[2], Faisal Saeed[3,*], Shadi Basurra[3]

[1] Universiti Malaya, Kuala Lumpur 50603, Malaysia
[2] Computer Information Science - Higher Colleges of Technology, UAE

[3] DAAI Research Group, Department of Computing and Data Science, School of Computing and Digital Technology, Birmingham City University, Birmingham B4 7XG, UK

zakhwanedu@gmail.com, noraisyah@um.edu.my, azizah.hitam@aiu.edu.my, faisal.saeed@bcu.ac.uk, shadi.basurra@bcu.ac.uk

Abstract: Cryptocurrency is branded as a digital currency, an alternative exchange currency system with significant ramifications for the economies of rising nations and the global economy. In recent years, cryptocurrency has infiltrated almost all financial operations; hence, cryptocurrency trading is frequently recognized as one of the most popular and promising means of profitable investment. Lately, with the exponential growth of cryptocurrency investments, many Alternative Coins (Altcoins) resurfaced to mimic the fiat currency. There are several methods to forecast cryptocurrency prices that have been widely used in forecasting fiat and stock prices. Artificial Intelligence (AI) ,Machine Learning(ML) and Deep Learning(DL) provide a different perspective on how investors can estimate crypto price trend and movement. In this paper, as cryptocurrency price is time-dependent, Recurrent Neural Network (RNN) is presented due to RNN's nature, which is well suited for Time Series Analysis (TSA). The topology of the proposed RNN model consists of three stages which are model groundwork, model development, and testing and optimization. The RNN architecture is extended to three different models specifically Long Short-Term Memory (LSTM), Gated Recurrent Unit (GRU), and Bi-Directional Long Short-Term Memory (LSTM). There are a few hyperparameters that affect the accuracy of the deep learning model in predicting cryptocurrency prices. Hyperparameter tuning set the basis for optimizing the model to improve the accuracy of cryptocurrency prediction. Next, the models were tested with data from different coins listed in the cryptocurrency market. Then, the model was experimented with different input features to figure out how accurate and robust these models in predicting the cryptocurrency price. GRU has the best accuracy in forecasting the cryptocurrency prices based on the values of Root Mean Square Error (RMSE), Mean Absolute Percentage Error (MAPE), and Executional Time, scoring 2.2201, 0.8076, and 200s using the intraday trading strategy as input features.

Keywords: Deep Learning, Cryptocurrency Prediction, Time Series Analysis (TSA), Alternative Coins (Altcoins)

1 Introduction

Cryptocurrencies have been dubbed a digital currency, an alternative exchange currency system with substantial implications for emerging nations and the global economy [1].The excitement around cryptocurrency is undeniable, particularly in recent years, as it has permeated virtually all financial activities. As a result, cryptocurrency trading is often regarded as one of the most popular and promising forms of successful investing.

Nonetheless, compared to the traditional fiat market, this ever-expanding financial industry is characterized by high volatility and price swings over time. Chowdhury et al. Nowadays, bitcoin forecasting is widely regarded as one of the most challenging time-series prediction issues due to the vast number of unknown variables and the extreme volatility of cryptocurrency values, which results in complex temporal dependencies [2].

The basic approach to forecasting cryptocurrency prices is to look for patterns, or what one can refer to as price fluctuations, in the market. Cryptocurrency analysis is extrapolating previous data to forecast future cryptocurrency prices. With the fast advancement of technology, particularly in Artificial Intelligence (AI), experts' educated estimates are made by machines.

Many firms implement machine learning and deep learning methods to analyze and forecast data. Nowadays, all financial analysts, crypto market analysts, and scientist are eager to find the most accurate ways to forecast cryptocurrency price movement. Due to its peculiarities and volatile nature, bitcoin price data is more difficult to anticipate than financial time series data. Support Vector Machines (SVM) and Artificial Neural Networks (ANN), for example, are commonly employed, which is evident in [3] to forecast crypto prices and movements. Every algorithm has a different method for learning patterns and then predicting them [4].

[5] conducted prediction on Bitcoin Price, focusing on 3 input variables, Close Price, Gold Price and Tweets (sentiment). The latter found that GRU outperformed CNN with an RMSE of 179.23, however LSTM was the best model with 151.67. [6] studied the precision with which the direction of the Bitcoin price in United States Dollar (USD) can be predicted. Besides feature selection, they also used Bayesian optimization to select LSTM parameters. The Bitcoin dataset ranged from the 19th of August 2013 to 19th of July 2016. The latter used multiple optimization methods to improve the performance of deep learning methods. The primary problem of their work is overfitting. Elsewhere GRUs offer additional benefits due to having a more straightforward structure [7], predicting the future price using Open, High, Low, Close and Volume Price of historical data which results in GRU having a quite low RMSE at 0.2113.To evaluate the possibility of outperforming the market, this paper pays particular attention to deep learning topics for cryptocurrency price predictions. So, the main objective of this paper is to examine cryptocurrency prediction algorithms using artificial intelligence and

propose a suitable model for prediction, acquire relevant input features affecting cryptocurrency prices to achieve an accurate result when predicting, develop and optimise the deep learning-based algorithm for cryptocurrency close price prediction, analyse the effect of different trading days and various input features combination on deep learning models prediction and evaluate the performance of the proposed cryptocurrency prediction model.

2 Methods

2.1 Overview

In this paper, the techniques and methods used in identifying specific parameters or processes is described. Proper selection of certain parameters and specific processes is essential in any research because every chosen method must have a valid justification and referencing. So, typically, developing a neural network model for cryptocurrency forecasting involves many processes and methods, this can be seen in the flow shown in Figure 1 below. This research requires many steps, activities and processes before delivering the result. Figure 1 shows the phases involved and deliverables.

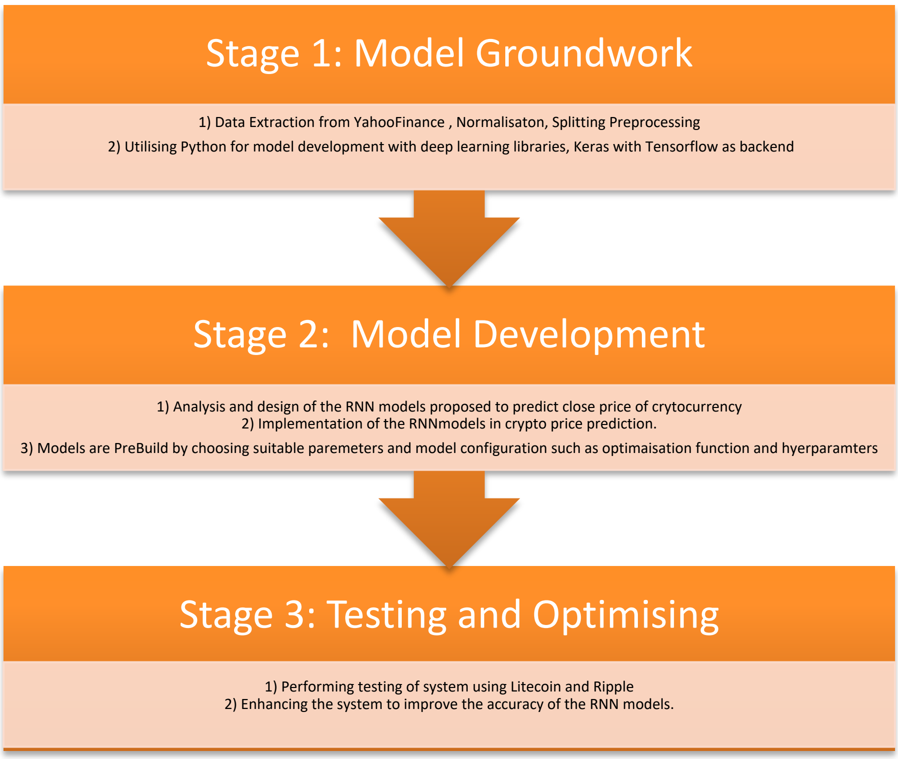

Figure 1. Overview of the research Methodology

The process for deploying the RNN model in predicting cryptocurrency price involves three stages. In Stage 1, to train the model, dataset was collected from YahooFinance containing historical price information from a rank 10 cryptocurrency: Litecoin. Dataset is gathered based on daily prices staring from 29th April 2013 – 27th February 2021. Data normalization is performed to increase the model's efficiency and accuracy.

In Stage 2, RNN-LSTM will be deployed as the predictive model. The model will be split into three parts which are training, validating, and testing. Models are then tuned to achieve optimum prediction by tuning the hyperparameters. In Stage 3, models are then tested, comparing the actual and predicted price. Enhancing is done by feeding different input features combination to the proposed models.

2.2 Model Groundwork

2.2.1 Data Extraction

There are a lot of steps taken during Stage 1. Firstly, data is extracted from YahooFinance. YahooFinance is a cryptocurrency exchange platform where we could obtain the data of crypto price freely and easily. However, it is limited to certain periods for certain data of crypto price. The data extracted from YahooFinance is required to be sorted and normalised so that it could be fitted to the RNN model that is used as well providing valid output result. The dataset was downloaded with .csv format which have some features like Date Open, High, Low, Close, Volume (OHLCV) and Marketcap. The Dataset of 2862 rows of which the row is based on the number of days, totaling up to 16,902 data points to be trained. Input features for the model set up is Close Price as the targeted predicted output is Close Price.

Table 1. Features Description of Litecoin LTC and Ripple XRP

Features	**Description**
Date	Date of observation.
Open	Opening price on the given day.
High	Highest price on the given day.
Low	Lowest price on the given day.
Close	Closing price on the given day.
Volume	Volume of transactions on the given day.

The Correlation Analysis is a method of analyzing the linear relationship between two variables. The two variables can be independent or correlated, and the strength of the relationship between two variables is called correlation. The correlation analysis uses the Pearson correlation coefficient. The Pearson correlation coefficient is a measure of the linear correlation between two variables. The Pearson correlation coefficient has a value between +1 and -1 due to the Kosi-Schwartz inequality, where +1 is a perfect positive linear correlation, 0 being no correlation, and -1 is a perfect negative linear relationship.

2.2.2 Data Normalization

The goal of normalization is to change quantitative values in the data set to a common scale, provided zero changes in the original range of values. The main idea behind normalization/standardization is always the same. Normalization/standardization is always based on the same principle. Variables measured on varying scales may not contribute equally to the model fitting & model learnt function, which may result in a bias. Before fitting a model to data from [0, 1] normalisation techniques such as Min Max Scaling are typically employed to address this possible issue.

2.2.3 Data Splitting

According to [8], data splitting are divided into train 70%, remaining 30% for both validation and test. Dataset is divided into a training set: observations between 29 April 2013–21 October 2018, which is 1988 trading days, a validation from 22 October 2018-23 December 2019 having 415 days and testing from 24 December 2019- 27 February 2021 also consist of 415 trading days. Similar data splitting is done for different altcoin, which is Ripple (XRP), whereby, Dataset is divided into a training set: observations between 5 August 2013 – 20 November 2018 which is 1934 trading days, a validation from 7 November 2018 - 9 January 2020 (400 trading days) testing from 10 January 2020- 27 February 2021 having 400 trading days.

2.3 Model Development

A RNN model is developed and modified by referring online sources. The most important function in the RNN is as below: the best combination of parameters. Hyperparameters are tuned to achieve the optimum predictive model.

2.3.1 Number of Epochs

In terms of artificial neural networks, an epoch refers to one cycle through the full training dataset. Usually, training a neural network takes more than a few epochs. In other words, if a neural network is fed with the training data for more than one epoch in different patterns, a better generalization is hoped when given a new "unseen" input (test data). For this work, the number of epochs are determined based on the commonly used values from existing research coupled with a trial and error values in determining the optimal value.

2.3.2 Adaptive Optimization Algorithm

Optimization algorithms are used to update weights and biases of a model to reduce error. Optimization algorithms can be divided into two main categories, which are constant learning rate algorithm and adaptive learning algorithm. The common first order optimization functions are Stochastic Gradient Descent (SGD), RMSProp and Adam.

Stochastic Gradient Descent (SGD) provides the foundation for several other learning algorithms, such as Adam and RMSProp; however, these algorithms have an adaptable learning rate. Indeed, the learning rate is a crucial hyperparameter in neural networks, as the loss function can be responsive or insensitive in certain directions of the parameter space.

For instance, gradients might get stalled at local minima or flat areas. The objective of the RMSProp method, a modified version of the AdaGrad algorithm [9], is to improve performance with non-convex functions. RMSProp modifies the gradient, g, by dividing the learning rate, η, by an exponentially declining average of squared gradients, θ which represents the error gradient.

$$\theta = \theta - \frac{\eta}{\sqrt{\mathbb{E}[g^2] + \epsilon}} g \ (1)$$

Adam developed by [10] is another adaptive algorithm and is nowadays one of the most used optimization algorithms. It is a combination of RMSProp and momentum SGD algorithms. β_1, β_2 represents initial decay rate,

$$\begin{aligned} m &= \beta_1 m + (1 - \beta_1) g \\ v &= \beta_2 v + (1 - \beta_2) g^2 \end{aligned} \ (2)$$

where m and v are estimates of the first moment (mean) and the second moment vectors of the gradient, g. These estimations are biased, so the authors compute a bias-correction at time step t

$$\begin{aligned} \hat{m} &= \frac{m}{1 - \beta_1^t} \\ \hat{v} &= \frac{v}{1 - \beta_2^t} \end{aligned} \ (3)$$

Hence, the update rule for given iteration:

$$\theta = \theta - \frac{\eta}{\sqrt{\hat{v}} + e} \hat{m} \ (4)$$

2.3.3 Dropout Rate

Dropout is a strategy that is designed to handle 2 major concerns overfitting, and bigger number of neurons. It prevents overfitting and enables the efficient combination of an exponentially large number of distinct neural network topologies [11]. The word "drop-out" refers to units in a neural network that are no longer active (both hidden and apparent). By dropping a unit from the network, we mean temporarily disconnecting it from all of its incoming and outgoing connections. The units to be dropped are chosen at random. In the simplest instance, each unit is preserved with a fixed probability p independent of other units, where p can be determined using a validation set or set to 0.5, which appears to be near to optimum for a broad variety of networks and tasks.

2.3.4 Batch Size

The batch size restricts the amount of samples displayed to the network prior to a weight change. When making predictions with the fitted model, the same constraint applies. In particular, the batch size employed while fitting. The model that determines how many forecasts must be made simultaneously. This becomes problematic when fewer forecasts are made than the batch size. Ideal results with a big batch size can be achieved however, predictions made for a single observation at a time while solving a problem involving a time series or sequence, may result in longer execution time. [12] utilises a batch size of 32, whereas [13] uses a batch size of 128.

2.4 Testing and Enhancing

Initially, the data gathered from YahooFinance in forecasting Litecoin (LTC). The LSTM model is tested with trial and error to obtain the best model which results in the highest accuracy or least average root mean square error in terms of share price. Then, the LSTM model is applied to other data from other coin which is Ripple (XRP).

Table 2 depicts different combination of input features to further examine the performance of forecasting for all models. Finally, the accuracy of LSTM GRU and Bi-LSTM model is compared and analyzed.

Table 2. Set of input features for testing and enhancing of predictive models

Input Fetaures Combination	Number of Features	Targeted Output
Close Price	1	Close Price
Open Price, High Price, Low Price and Close Price (OHLC)	4	
Open Price, High Price, Low Price ,Close Price and MarketCap (OHLCM)	5	
Open Price, High Price, Low Price ,Close Price Volume and MarketCap (OHLCVM)	6	

3 Results and Discussion

3.1 Overview

In this paper, the preliminary results of the proposed model and the data extraction method for the training validation and testing of the model were being discussed. As mentioned, the proposed model is RNN, and the parameters chosen for the model as mentioned previously were being applied and analyzed. A comparative analysis of RNN models is examined. There are 4 hyperparameters to be manipulated in the LSTM, GRU and Bi-LSTM model. Validation Loss and Training Loss are performed to overcome overfitting issue.

The first parameter is number of epochs, followed by Adaptive Optimization Algorithm, Batch Size and Dropout Rate. Then, each model is tested with different combination of input features to further observe the robustness of each model. The score indicators are RMSE, MAPE and Executional Time.

3.2 Evaluation of LSTM, GRU and Bi-LSTM model

3.2.1 Determine optimum Number of epochs in LSTM, GRU and Bi-LSTM model

In this experiment, the parameter of the model whereby the number of epochs is analyzed. The hyperparameter is tested to the training set of LTC data. The number of epochs used are 20,40,60,80, 100.

Table 3. Effect of number of epochs on LSTM, GRU and Bi-LSTM RMSE

Models	Number of Epoch	RMSE
LSTM	20	5.1693
	40	3.226
	60	0.1993
	80	**0.00736**
	100	0.6743
GRU	20	5.7259
	40	3.7321
	60	2.0293
	80	0.9216
	100	**0.0693**
Bi-LSTM	20	3.2946
	40	2.0293
	60	0.5294
	80	**0.4331**
	100	0.7723

Based on Table 3 and Table 4 above, it could be concluded that the optimum number of epochs are 80,100 and 80 with RMSE of 0.00736, 0.0693 and 0.7723 for LSTM, GRU and Bi-LSTM respectively.

3.2.2 Determine optimum optimization algorithm in LSTM, GRU and Bi-LSTM model

In this experiment, the parameter of the model whereby the adaptive optimizer is analyzed. The hyperparameter is tested to the training set of LTC data. The optimizer used are Adam and RMSprop.

Table 4. Effect of optimizers on LSTM,GRU and Bi-LSTM RMSE

Models	Optimiser	RMSE
LSTM	**Adam**	**2.0982**
	RMSProp	2.6613
GRU	**Adam**	**1.6474**
	RMSProp	3.3573
Bi-LSTM	**Adam**	**0.8732**
	RMSProp	3.2668

Based on table above, it could be concluded that the optimum optimiser are Adam algorithm with RMSE of 2.0982, 1.6474 and 0.8732 for LSTM, GRU and Bi-LSTM respectively.

3.2.3 Determine optimum Batch Size in LSTM, GRU and Bi-LSTM model

In this experiment, the parameter of the model whereby the batch size is analysed. The hyperparameter is tested to the training set of LTC data. The batch size used are 32,64 and 128**.**

Table 5. Effect of batch sizes on LSTM,GRU and Bi-LSTM RMSE

Models	Batch Size	RMSE
LSTM	32	2.3060
	64	2.9178
	128	**0.1993**
GRU	**32**	**0.9216**
	64	1.8813
	128	2.6249
Bi-LSTM	32	1.8765
	64	2.6425
	128	**0.8834**

Based on table above, it could be concluded that the optimum batch sizes are 128,32 and 128 with RMSE of 0.1993, 0.9216 and 0.8834 for LSTM, GRU and Bi-LSTM respectively.

3.2.4 Determine optimum Dropout Rate in LSTM, GRU and Bi-LSTM model

In this experiment, the parameter of the model whereby the batch size is analysed. The hyperparameter is tested to the training set of LTC data. The dropout rate used are 0.1,0.2,0.4, 0.5 and 0.7

Table 6. Effect of dropout rates on LSTM, GRU and Bi-LSTM RMSE

Models	Dropout Rate	RMSE
LSTM	**0.1**	**1.8503**
	0.2	1.8666
	0.4	2.9457
	0.5	2.9968
	0.7	1.9969
GRU	**0.1**	**1.6014**
	0.2	1.9527
	0.4	3.3642
	0.5	3.0203
	0.7	1.6015
Bi-LSTM	0.1	1.4321
	0.2	**0.7782**
	0.4	4.0991
	0.5	0.9963
	0.7	0.8765

Based on table above, it could be concluded that the optimum dropout rates are 0.1, 0.1 and 0.2 with RMSE of 1.8503, 1.6014 and 0.7782 for LSTM, GRU and Bi-LSTM respectively.

3.2.5 Evaluate Performance of LSTM, GRU and Bi-LSTM model with different input features on LTC and XRP

The combinations are selected based on the correlation weight of features towards the predicted output. Open Price, High Price, Low Price, Close Price (OHLC) are selected as a prediction benchmark which is similarly used by [14] which the latter used similar models to analyse their performances when predicting the close price of cryptocurrency.

In this experiment, the models are tested with different combination input features using the LTC and XRP dataset. The experiment will be carried out with four different input features combinations according to the cases namely, Close Price for a univariate model, Open Price, High Price, Low Price, Close Price (OHLC) , Open Price, High Price, Low Price, Close Price and Market Cap (OHLCM)and Open Price, High Price, Low Price, Close Price, Volume and Market Cap (OHLCVM). The overall performance of all predictive models are tabulated a follows;

Table 7. Summary of Performance Evaluation for RMSE, MAPE and Execution Time of all models for LTC and XRP closing prices.

Reference	Input Features	Model	Result					
			RMSE		MAPE (%)		Time (s)	
			LTC	XRP	LTC	XRP	LTC	XRP
Hameyal et . al 2021	OHLC Price	LSTM	3.069	-	0.8474	-	NA	-
		GRU	**0.825**	-	**0.2116**	-	NA	-
		Bi-LSTM	4.307	-	2.332	-	NA	-
This paper	Close Price	LSTM	2.5642	0.1260	0.8893	0.8893	480	480
		GRU	**2.4960**	**0.0237**	0.4888	0.4888	200	200
		Bi-LSTM	4.7732	0.1307	0.8664	0.8664	1200	1200
This paper	OHLC Price	LSTM	3.8869	0.0390	1.3596	1.5791	640	640
		GRU	**2.2201**	**0.0089**	0.8076	0.6620	200	200
		Bi-LSTM	4.9831	0.0513	0.9352	0.8237	4320	4320
This paper	OHLC Price and MarketCap	LSTM	3.2258	**0.0125**	0.7282	0.6020	800	800
		GRU	**3.0567**	0.0192	0.6357	1.4997	600	600
		Bi-LSTM	3.9137	0.0367	1.441	1.4537	4640	4640
This paper	OHLC Price Volume and MarketCap	LSTM	2.2237	0.0073	0.7782	3.4628	800	800
		GRU	**0.9589**	0.0338	0.6659	1.3054	600	600
		Bi-LSTM	2.5738	**0.0046**	1.2090	0.8723	4640	4640

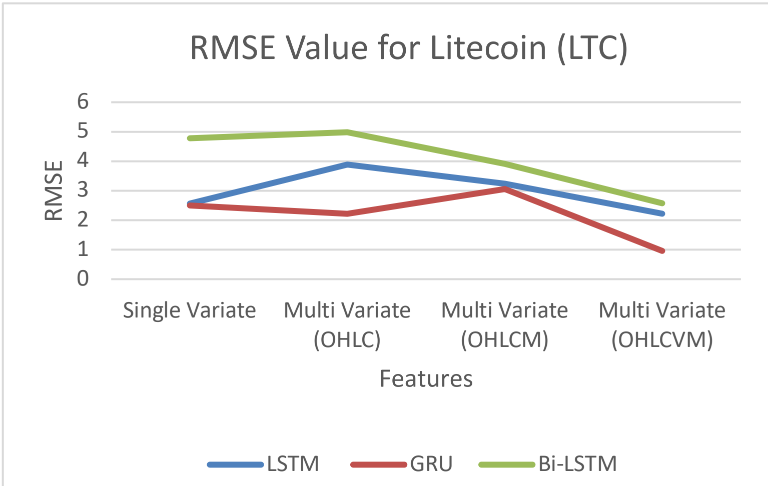

Figure 2. Line Chart Representation on the effect of different input features on LSTM,GRU and Bi-LSTM models for LTC

From Figure 2, GRU outperforms LSTM and Bi-LSTM when predicting the price of LTC. Based on previous work done by [14] also found that GRU outperforms LSTM and BILSTM, with 2.201 RMSE value in this paper when OHLC price is treated as input features which justifies the development and optimization of this model. The RMSE value obtained from this paper deviates by 1.671 in RMSE. The latter also obtained a RMSE value of 3.069 for LSTM (deviation of 16.44% from this paper) and 4.307 for Bi-LSTM (deviation of 10.82% from this paper) which suggests a closer value to the RMSE obtained in this paper and [14] findings for LSTM and Bi-LSTM models. The MAPE scored by [14] and this paper is not far off; LSTM of 0.874 and 0.8893 respectively, GRU of 0.2216 and 0.4888 respectively and Bi-LSTM of 2.332 and 0.8864 respectively. The above figure also shows that when the input features increase, GRU has an inconsistent result as the RMSE value fluctuates but still managed to outperform both LSTM and Bi-LSTM when all 6 features are fed. On the other hand, Bi-LSTM and LSTM shows a positive impact when the features increase.

Although, the RMSE value spiked when OHLC is tested out, they both shows gradual reduction in terms of RMSE value when OHLCM and OHLCVM are experimented. This shows that Bi-LSTM and LSTM are more robust and accurate as more input data are being fed to the model.

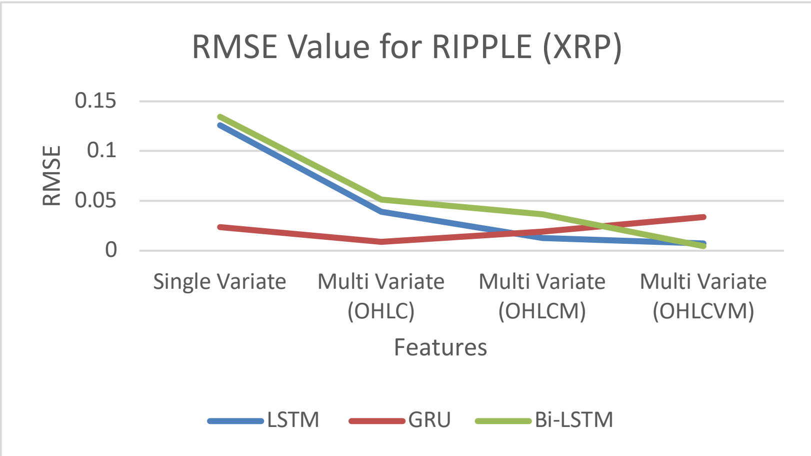

Figure 3. Line Chart Representation on the effect of different input features on LSTM,GRU and Bi-LSTM models for LTC

Figure 3 depicts the RMSE value when XRP are used as dataset which underwent similar experiment from [15] successfully verified that Bi-LSTM network is the most effective model when predicting the close price of XRP. The latter however used a different input feature which involves OHLC Price and Volume.

Bi-LSTM outperforms LSTM and GRU when all 6 features OHLCVM are treated as input, in contrast to when LTC is utilised. Bi-LSTM scored an astonishing 0.0046 in RMSE value scoring better than [15]findings; 0.979 in RMSE. In this paper, LSTM is not far off with only 0.0073 of RMSE while GRU performed the worst scoring a mere 0.0338 as for the RMSE.

Again, for single input features GRU performs the best with the lowest RMSE score of 0.0237, which is logical since GRU having a faster computational time and accomplished better result due to having only update and reset gate. As a matter of fact, a simpler model like GRU, caters for a smaller dataset size while high complexity model namely LSTM and Bi-LSTM are superior when dataset size broadens.

4 Conclusion

This paper discusses on the forecasting cryptocurrency prices using deep learning models as a tool for cryptocurrency investors. The proposed forecasting model has been made based on the studied reviews which were RNN ecxtensions. Performance scores – RMSE,MAPE and computational time - were calculated for LTC and XRP to test the accuracy of the proposed models. Based on these outcomes, GRU were exceptional in terms of performance for LTC for every different input features. This model is considered the best model however, LSTM and Bi-LSTM models showed superiority when the number of input features fed increased, indicating the memory capacity of the bi-directional architecture in predicting a time large time series data. For the extension of this work, sentiment analysis should be considered as a factor on how they influence the cryptocurrency price as well as performing dimensionality reduction technique to further experiment the performance of higher complexity models such as LSTM and Bi-LSTM.

References

[1] Nasir, M. A., Huynh, T. L. D., Nguyen, S. P., & Duong, D. (2019). Forecasting cryptocurrency returns and volume using search engines. Financial Innovation, 5(1), 1-13. https://doi.org/10.1186/s40854-018-0119-8

[2] Livieris, I. E., Pintelas, E., Stavroyiannis, S., & Pintelas, P. (2020). Ensemble deep learning models for forecasting cryptocurrency time-series. Algorithms, 13(5), 121.

[3] Hitam, N. A., & Ismail, A. R. (2018). Comparative performance of machine learning algorithms for cryptocurrency forecasting. Ind. J. Electr. Eng. Comput. Sci, 11(3), 1121-1128.

[4] Selvamuthu, D., Raj, R., Sahu, K., & Aggarwal, A. (2019, December). Tradeoff Between Performance and Energy-Efficiency in DRX Mechanism in LTE-A Networks. In 2019 IEEE International Conference on Advanced Networks and Telecommunications Sys-tems (ANTS) (pp. 1-5). IEEE.

[5] Aggarwal, A., Gupta, I., Garg, N., & Goel, A. (2019). Deep learning approach to determine the impact of socio economic factors on bitcoin price prediction. In 2019 Twelfth International Conference on Contemporary Computing (IC3) (pp. 1-5). IEEE.

[6] McNally, S., Roche, J., & Caton, S. (2018). Predicting the Price of Bitcoin Using Machine. 26th Euromicro International Conference on Parallel, Distributed and Network-based Processing (PDP), (pp. 339-343). Cambridge, UK.

[7] Tanwar, S., Patel, N. P., Patel, S. N., Patel, J. R., Sharma, G., & Davidson, I. E. (2021). Deep Learning-Based Cryptocurrency Price Prediction Scheme With Inter-Dependent Relations. IEEE Access, 9, 138633-138646

[8] Dutta, A., Kumar, S., & Basu, M. (2020). A gated recurrent unit approach to bitcoin price prediction. Journal of Risk and Financial Management, 13(2), 23.

[9] Hinton, G., Srivastava, N., & Swersky, K. (2012). Neural networks for machine learning lecture 6a overview of mini-batch gradient descent. Cited on, 14(8), 2

[10] Kingma, D. P., & Ba, J. (2014). Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980.

[11] Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: a simple way to prevent neural networks from overfitting. The journal of machine learning research, 15(1), 1929-1958

[12] Ranawat, K., & Giani, S. (2021). Artificial intelligence prediction of stock prices using social media. arXiv preprint arXiv:2101.08986.

[13] Koo, E., & Kim, G. (2021). Prediction of Bitcoin price based on manipulating distribution strategy. Applied Soft Computing, 110, 107738

[14] Hamayel, M. J., & Owda, A. Y. (2021). A Novel Cryptocurrency Price Prediction Model Using GRU, LSTM and bi-LSTM Machine Learning Algorithms. AI, 2(4), 477-496.

[15] Birim, S. O. (2022). An analysis for cryptocurrency price prediction using lstm, gru, and the bi-directional implications.