Comparison of Distance Measurement in Time Series Clustering for Predicting Bitcoin Prices

Ui-Jun Baek
Computer Information and Science
Korea University
Sejong, Korea
pb1069@korea.ac.kr

Mu-Gon Shin
Computer Information and Science
Korea University
Sejong, Korea
tm0309@korea.ac.kr

Min-Seong Lee
Computer Information and Science
Korea University
Sejong, Korea
min0764@korea.ac.kr

Boseon Kim
Computer Information and Science
Korea University
Sejong, Korea
boseon12@korea.ac.kr

Jee-Tae Park
Computer Information and Science
Korea University
Sejong, Korea
pjj5846@korea.ac.kr

Myung-Sup Kim
Computer Information and Science
Korea University
Sejong, Korea
tmskim@korea.ac.kr

***Abstract*—Since the development of Bitcoin, the first blockchain-based cryptocurrency, many cryptocurrencies have formed and have traded in markets. The integrity and anonymity of cryptocurrency was enough to raise its value and its price gained worldwide attention. Therefore, many studies are being carried out to predict the price of cryptocurrency for make a profit. We cluster time series through K-Medoids algorithm and train and evaluate each cluster with predictive models. We also examine the predictive performance in Bitcoin price according to the various distance measurement of clustering.**

Keywords—blockchain, bitcoin, time series clustering, correlation coefficient

I. INTRODUCTION

Bitcoin was developed by Satoshi Nakamoto as the first blockchain-based cryptocurrency and became the most popular. Since then, many cryptocurrency have been developed, traded on the market, and have a huge market size. Although price predicting studies are being conducted on various cryptocurrency including Bitcoin, they indicate the predictive performance that is not sufficient for practical use.

We carried out this study by referring to [1], [2]. Abhishta, et al. [1] performed an event analysis to evaluate whether there is an impact of a DDoS attack on the volume traded on the exchange in 17 different cases. they refer that most of negative impact(13 of 17) due to a DDoS attack is recovered within the same day. In the other cases, negative impacts are recovered within two days or are not recovered within five days. Vasek, et al. [2] examined the potential drivers of Bitcoin prices, ranging from fundamental sources to speculative and technical ones. They also refer that Bitcoin forms a unique asset possessing properties of both a standard financial asset and a speculative one. Based on the paper we refer to, we assume that the time series of bitcoin prices have distinguishable distributions by certain factors including positive effects and negative effects. We cluster the entire price time series based on K-Medoids algorithm and compare the average silhouette coefficient according to the various distance measurement algorithms of K-Medoids. We also create an LSTM-based predictive model for each time series cluster and evaluate its predictive performance.

Following the introduction of Chapter 1, Chapter 2 describes the basic concepts of this paper and the existing studies. Chapter 3 describes the experimental process and evaluation methods, and Chapter 4 shows the experimental results and analyze the results. In Chapter 5, we brief conclusion, our contributions, limitations and future works.

II. RELATED WORKS

This chapter describes the K-Medoids algorithm and various distance measurement algorithm. It also briefs on existing studies on bitcoin price prediction and time series clustering.

A. K-Medoids clustering

The K-Medoids algorithm belongs to partitioning method that divides given data into multiple partitions. The goal of K-Medoids is to split a given n d-dimensional data objects into $k(\leq n)$ set $S(S_1, S_2, \dots, S_k)$ with maximum cohesion between objects in the set, and when μ_i is centroid of set S_i:

$$\underset{S}{Argmin} \sum_{i=1}^{k} \sum_{x \in S_i} \|x - \mu_i\|^2 \qquad (1)$$

K-Medoids are similar to K-Means, and K-Medoids choose the centroid of each cluster from among the given data objects, but K-Means selects the average position of the data object in a cluster as the centroid. In general, K-Medoids algorithm is more robust to noise and outliers as compared to K-Means. Clusters can be assessed with methods such as the silhouette method. In K-Medoids, it is difficult to find an optimal solution because the results may vary depending on the initial center points of a given k, and [3] was proposed to supplement it.

B. Distance measurement algorithms

In this paragraph we explain the four methods of measuring distances compared in this paper, which include Euclidean, Cosine, Pearson correlation, and Spearman's rank correlation. Euclidean distance is the length of the line connecting the two points, and when, in Cartesian coordinates, $(x_1, x_2, .., x_n)$ and $y\,(y_1, y_2, \dots, y_n)$ are two points in Euclidean n-space, distance D_{ed} is as follows:

This work was supported in part by the Institute of Information and Communications Technology Planning and Evaluation (IITP) grant funded by the Korean Government (MSIT) (No.2018-0-00539, Development of Blockchain Transaction Monitoring and Analysis Technology), in part by the Technology Innovation Program (No. 20008902, Development of SaaS SW Management Platform based on 5 Channel Discovery technology for IT Cost Saving) funded by the Ministry of Trade, Industry & Energy (MOTIE, Korea) and the Korea Evaluation Institute of Industrial Technology (KEIT).

$$D_{ed}(x,y) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2} \qquad (2)$$

Cosine similarity refers to a similar degree between vectors measured using cosine values of the angle between two vectors in the inner product space. Therefore, cosine similarity determines the similarity in direction other than the size of the vector. If the direction is completely same, the value is 1; if the direction is completely opposite, the value is -1. When two vectors $x(x_1, x_2, \ldots, x_n)$ and $y\ (y_1, y_2, \ldots, y_n)$ are given, the cosine similarity D_{cs} is as follows:

$$D_{cs}(x,y) = \frac{\sum_{i=1}^{n} x_i \times y_i}{\sqrt{\sum_{i=1}^{n}(x_i)^2} \times \sqrt{\sum_{i=1}^{n}(y_i)^2}} \qquad (3)$$

The Pearson correlation coefficient represents a linear correlation between the two variables $x(x_1, x_2, \ldots, x_n)$ and $y\ (y_1, y_2, \ldots, y_n)$ and has a value between -1 and 1.If the coefficient value is 1, it is determined that the two variables have a perfect positive linear correlation, and if the value is -1, it is determined that the two variables are a perfect negative linear correlation. $x(x_1, x_2, \ldots, x_n)$ and $y\ (y_1, y_2, \ldots, y_n)$ are given, the Pearson correlation coefficient D_{pc} is as follows:

$$D_{pc}(x,y) = \frac{\sum_{i=1}^{n}(x_i - \overline{x})(y_i - \overline{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \overline{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \overline{y})^2}} \qquad (4)$$

The Spearman's rank correlation coefficient measures the statistical dependence between the ranks of two variables. The Spearman's rank correlation coefficient between the two variables is equal to the Pearson correlation coefficient between the ranking values of the two variables. When two vectors $x(x_1, x_2, \ldots, x_n)$ and $y\ (y_1, y_2, \ldots, y_n)$ are given and if *rx* is the rank of x and *ry* is the rank of *y*, the Spearman's rank correlation coefficient D_{sr} is as follows:

$$D_{sr}(x,y) = \frac{\sum_{i=1}^{n}(rx_i - \overline{rx})(ry_i - \overline{ry})}{\sqrt{\sum_{i=1}^{n}(rx_i - \overline{rx})^2}\sqrt{\sum_{i=1}^{n}(ry_i - \overline{ry})^2}} \qquad (5)$$

The characteristics that prices can change and the high growth rate, cryptocurrency has become a form of speculative asset, and the price prediction of cryptocurrency has become a global concern. Reference [4] examines whether bitcoin returns are predictable by a large set of bitcoin price-based technical indicators through constructing a classification tree-based model for return prediction using 124 technical indicators. [4] reports that using big data and technical analysis can help predict bitcoin returns that are hardly driven by fundamentals. Also, Reference [4] reports that using big data and technical analysis can help predict Bitcoin returns that are hardly driven by fundamentals. Reference [5,6,7,8] uses the deep learning method to predict the price of Bitcoin. [5] used Bayesian Neural Networks and reported that the BNN model succeeded in relatively accurate direction prediction. [6] used Ensembles of Neural Networks and reported that the ensemble method, Genetic Algorithm based Selective Neural Network Ensemble, was able to perform well for the classification task with consistent accuracy of around 58% to 63%. [7] used backpropagation neural network (BPNN), genetic algorithm neural network (GANN), genetic algorithm backpropagation neural network (GABPNN), neuro-evolution of augmenting topologies (NEAT) and compared accuracy between these methods. [8] used a recurrent neural network (RNN), long short-term memory (LSTM), and gated recurring unit (GRU) and compared accuracy between them. [9,10,11] predicted the price of Bitcoin through machine learning such as a support vector machine (SVM). [12] analyzed and predicted the price of Bitcoin through the volume of Twitter and Google Trends data. [13] analyzed Twitter signals as a medium for user sentiment to predict the price fluctuations of a small-cap alternative cryptocurrency called ZClassic. Like the above studies, various researches are being conducted to predict the price of the cryptocurrency, but they do not have high accuracy. Therefore, we perform time-series clustering that can cluster time series according to distribution characteristics as a kind of preprocessing and help with learning. We also compare the predictive performance of time series clustered by various distance measurement algorithms and investigate the distance measuring algorithms that can lead to the greatest improvement in predictive performance.

III. Experiment

This chapter describes all the processes of the experiment and evaluation methods. The overall process of this experiment is shown in Figure1.

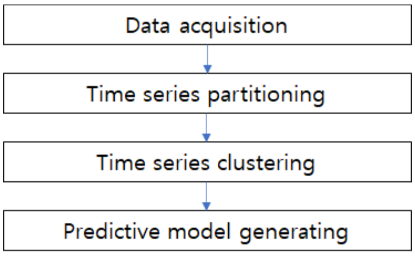

Fig. 1. Overall process of experiment

A. Data acqusition

Bitcoin time series data can be downloaded from Blockchain.com[14]. We downloaded the market price time series from June 1, 2017 to June 1, 2020, and the number of data is 1,095.

B. Time series clustering

We partition the downloaded time series data into a certain split unit. For various comparisons, we partitioned whole time-series dataset into 10 day intervals from unit 10 to unit 50. Each time series data set includes a time series of (length of whole time series - split unit + 1). The partitioning process is shown in Figure 2.

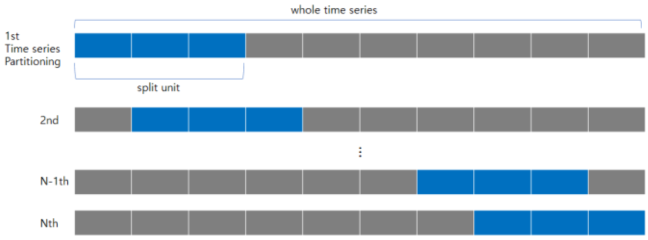

Fig. 2. Partitioning process - length of whole time series of example is 10, split unit of example is 3. As a result, the example generates 8 (10-3+1) time series partitions.

C. Time series clustering

We cluster time series partitions through K-Medoids algorithm. We also applied and compared four methods of

TABLE I. EXPERIMENTAL RESULT

Unit	k	Euclidean			Cosine			Pearson			Spearman's rank		
		SC	ACC	ACC (k=1)	SC	ACC	ACC (k=1)	SC	ACC	ACC (k=1)	SC	ACC	ACC (k=1)
10	3	0.653	0.487	**0.508**	0.55	**0.504**	0.501	0.423	**0.524***	0.497	0.437	0.511	**0.516**
	4	0.567	**0.527**	0.482	0.386	0.502	**0.525**	0.406	0.495	**0.514**	0.401	**0.551***	0.516
	5	0.582	**0.512**	0.505	0.405	0.491	**0.509**	0.319	**0.506**	0.482	0.3356	0.491	**0.529***
	6	0.554	0.485	0.525	0.33	0.494	**0.514**	0.304	**0.579***	0.481	0.337	**0.553**	0.5
	7	0.489	0.501	0.512	0.256	0.481	**0.518**	0.296	**0.542***	0.513	0.265	0.509	**0.524**
20	3	0.632	**0.507**	0.506	0.53	**0.5**	0.485	0.449	0.498	**0.518**	0.638	**0.546***	0.523
	4	0.578	0.481	**0.507**	0.444	**0.529**	0.521	0.441	0.506	**0.528***	0.347	0.514	**0.52**
	5	0.587	**0.527**	0.486	0.402	**0.523**	0.508	0.4	0.51	**0.517**	0.314	**0.551***	0.496
	6	0.455	**0.508**	0.492	0.336	**0.515**	0.514	0.4	**0.549**	0.482	0.299	**0.554***	0.496
	7	0.469	**0.511**	0.487	0.353	**0.509**	0.484	0.399	**0.568***	0.529	0.289	**0.515**	**0.515**
30	3	0.624	0.522	**0.526**	0.58	**0.528**	0.511	0.446	0.5	**0.503**	0.42	**0.55***	0.481
	4	0.610	**0.521**	0.508	0.412	**0.5**	0.484	0.396	**0.526**	0.494	0.419	**0.548***	0.506
	5	0.589	**0.522**	0.52	0.352	**0.525**	0.52	0.341	**0.559***	0.501	0.355	**0.534**	0.497
	6	0.577	0.482	**0.497**	0.32	**0.507**	0.486	0.268	**0.554**	0.504	0.35	**0.573***	0.499
	7	0.536	0.484	**0.509**	0.299	**0.496**	0.484	0.253	**0.516**	0.484	0.296	**0.563***	0.483
40	3	0.598	**0.499**	0.497	0.51	0.497	**0.515**	0.434	**0.579***	0.529	0.355	0.492	**0.504**
	4	0.571	**0.512**	0.493	0.447	**0.486**	0.484	0.403	**0.535**	0.483	0.352	**0.54***	0.515
	5	0.574	0.486	**0.523**	0.44	**0.497**	0.487	0.383	**0.572***	0.508	0.313	**0.566**	0.482
	6	0.520	0.489	**0.515**	0.44	0.502	**0.516**	0.338	**0.554***	0.495	0.282	**0.515**	0.502
	7	0.480	**0.51**	0.5	0.334	**0.516**	0.5	0.276	**0.517**	0.513	0.274	0.506	**0.525***
50	3	0.570	0.527	**0.53**	0.401	0.506	**0.519**	0.447	**0.53***	0.518	0.45	**0.49**	**0.49**
	4	0.561	0.511	**0.527**	0.38	0.508	**0.52**	0.375	**0.501**	0.497	0.389	**0.57***	0.484
	5	0.521	0.518	**0.524**	0.342	**0.486**	0.485	0.349	**0.546***	0.525	0.386	**0.51**	0.497
	6	0.524	**0.516**	0.496	0.297	0.482	**0.501**	0.317	**0.578***	0.505	0.362	**0.546**	0.496
	7	0.476	**0.495**	0.495	0.29	0.501	**0.51**	0.317	0.524	**0.529***	0.264	**0.491**	0.481

SC : Silhouette Coefficient, **ACC**: Accuracy, **Label *** : The distance measurement method that represents the highest accuracy in each trace

distance measurements mentioned in the related works to compare results according to the distance measurements methods. Because the K-Medoids algorithm require fixed number *k* and the results can vary depending on K, we change K in the range of 3 to 8 and clustered the time series. The quality of all clustering results is evaluated by the silhouette coefficient and compared in Chapter 4. When the cohesion a_i is the average distance between elements in the cluster to which point belongs and the separation b_i is the distance from point i to the nearest cluster among all other clusters, the silhouette coefficient S_i for a point i is calculated as follows:

$$S_i = \frac{b_i - a_i}{max(a_i, b_i)} \tag{6}$$

D. Predictive model generation

We generate an LSTM-based predictive model and train each cluster by input. One predictive model can train one cluster which is a partitioned time series. For example, If the number of K is seven, there are seven predictive models generated. Predictive model is implemented with Python, Tensorflow and Keras. Function Early stop is applied for proper training, and the number of hidden layers is fixed at five. For proper evaluation, the input data is divided in a ratio of 6:2:2 and the division process is performed randomly. The accuracy of the predictive model is compared in Chapter 4. We labeled the directions of fluctuation as 1 and −1, and measure accuracy based on the predicted values:

$$\begin{aligned} TP &: real\ value \equiv 1 \cap predicted\ value > 0 \\ TN &: real\ value \equiv 1 \cap predicted\ value < 0 \\ FP &: real\ value \equiv 0 \cap predicted\ value < 0 \\ FN &: real\ value \equiv 0 \cap predicted\ value > 0 \end{aligned} \tag{7}$$

$$Accuracy = \frac{TP + FP}{TP + FN + FP + FN}$$

IV. EXPERIMENTAL RESULT

In this chapter, we present and interpret the experimental results according to the distance measurement method and experimental parameters in Table 1.

The clustering result with Euclidean distance measurement shows a higher silhouette coefficient compared to other distance measurement methods and tends to decrease slightly as *k* increases. However, it does not show better prediction performance than the prediction result without clustering (*k*=1). This is thought to be because the clustering does not reflect the time series characteristics. Also, there is no correlation between the split unit, *k*, and accuracy. The clustering result with cosine distance measurement is lower than Euclidean, but it shows a higher silhouette coefficient than other distance measurement methods. Similarly, as *k* increases, the silhouette coefficient decreases slightly. This method also does not reflect the time series properties, so it does not have better predictive performance than predicted results without distinct clustering. As with Euclidean, there is no correlation between a split unit, *k*, and accuracy. Clustering results with Pearson's correlation distance measurements show lower silhouette coefficients than Euclidean and cosine methods. The result of the Pearson distance measurement method is better than the previous two methods because it can reflect time series characteristics. Results with Pearson distance measurements outperformed prediction results without clustering, except for six out of a total of 25 traces. Also, it can be seen that the performance is improved in the range of 30 to 50 split units. Results with Pearson's correlation distance measurement show up to 9.8 percent better prediction accuracy than results without clustering. Clustering results with the Spearman correlation distance measurement show similar results to the case of Pearson correlations, with a prediction performance improvement of up to 8.6 percent above the results without clustering. In summary, except for 4 of the total traces, the results with Pearson distance measurement and Spearman distance measurement are the highest. These results mean that when performing a clustering with a distance measurement method that can reflect the characteristics of a time series, segmenting the time series, and learning it to generate a multi-prediction model, performance can be improved in time-series prediction.

V. Conclusion

We split the Bitcoin time series and cluster them to generate multiple predictive models. We compare the prediction model without clustering and the prediction model without clustering. In addition, we experimented with changing the distance measurement algorithm, the number of Ks, and the unit of division to investigate various results. As a result, the prediction results with Euclidean distance measurement and cosine distance measurement do not show any significant performance improvement compared to the prediction results without clustering. On the other hand, the prediction results with Pearson distance measurement and Spearman distance measurement showed a clear performance improvement over the non-clustered prediction results and improved prediction accuracy of up to 9.8 percent. This is because Pearson distance measurement and Spearman distance measurement can analyze the correlation between time series well.

We simply collect time series data with only one feature, and the prediction model is not optimized, so we do not have high prediction performance. However, we expect that performance improvement can be achieved by applying the clustering method to the previous studies mentioned in the related study.

In the future, we will experiment by applying various clustering methods and distance measurements, including the partitioning clustering methods, and apply it to previous studies.

References

[1] A. Abhishta, R. Joosten, S. Dragomiretskiy, and L. J. Nieuwenhuis, “Impact of Successful DDoS Attacks on a Major Crypto-currency Exchange,” in 2019 27th Euromicro International Conference on Parallel, Distributed and Network-Based Processing (PDP), 2019, pp. 379–384.

[2] L. Kristoufek, “What are the main drivers of the Bitcoin price? Evidence from wavelet coherence analysis,” PloS one, vol. 10, no. 4, 2015.

[3] I. S. Jacobs and C. P. Bean, “Fine particles, thin films and exchange anisotropy,” in Magnetism, vol. III, G. T. Rado and H. Suhl, Eds. New York: Academic, 1963, pp. 271–350.

[4] E. Schubert and P. J. Rousseeuw, “Faster k-Medoids clustering: improv- ing the PAM, CLARA, and CLARANS algorithms,” in International Conference on Similarity Search and Applications, 2019, pp. 171–187.

[5] J.-Z. Huang, W. Huang, and J. Ni, “Predicting bitcoin returns using high-dimensional technical indicators,” The Journal of Finance and Data Science, vol. 5, no. 3, pp. 140–155, 2019.

[6] H. Jang and J. Lee, “An empirical study on modeling and prediction of bitcoin prices with bayesian neural networks based on blockchain information,” Ieee Access, vol. 6, pp. 5427–5437, 2017.

[7] E. Sin and L. Wang, “Bitcoin price prediction using ensembles of neural networks,” in 2017 13th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD), 2017, pp. 666–671.

[8] A. Radityo, Q. Munajat, and I. Budi, “Prediction of Bitcoin exchange rate to American dollar using artificial neural network methods,” in 2017 International Conference on Advanced Computer Science and Information Systems (ICACSIS), 2017, pp. 433–438.

[9] D. Zhao, A. Rinaldo, and C. Brookins, “Cryptocurrency Price Prediction and Trading Strategies Using Support Vector Machines,” arXiv preprint arXiv:1911.11819, 2019

[10] S. Velankar, S. Valecha, and S. Maji, “Bitcoin price prediction using machine learning,” in 2018 20th International Conference on Advanced Communication Technology (ICACT), 2018, pp. 144–147.

[11] R. Mittal, S. Arora, and M. P. S. Bhatia, “Automated cryptocurrencies prices prediction using machine learning,” Division of Computer Engineering, Netaji Subhas Institute of Technology, India, vol. 8, pp. 2229– 6956, 2018.

[12] J. Abraham, D. Higdon, J. Nelson, and J. Ibarra, “Cryptocurrency price prediction using tweet volumes and sentiment analysis,” SMU Data Science Review, vol. 1, no. 3, p. 1, 2018.

[13] T. R. Li, A. Chamrajnagar, X. Fong, N. Rizik, and F. Fu, “Sentimentbased prediction of alternative cryptocurrency price fluctuations using gradient boosting tree model,” Frontiers in Physics, vol. 7, p. 98, 2019.