Pattern graph tracking-based stock price prediction using big data

Seungwoo Jeon, Bonghee Hong, Victor Chang

PII: S0167-739X(17)30199-1
DOI: http://dx.doi.org/10.1016/j.future.2017.02.010
Reference: FUTURE 3328

To appear in: *Future Generation Computer Systems*

Received date : 9 November 2016
Revised date : 18 January 2017
Accepted date : 6 February 2017

Please cite this article as: S. Jeon, et al., Pattern graph tracking-based stock price prediction using big data, *Future Generation Computer Systems* (2017), http://dx.doi.org/10.1016/j.future.2017.02.010.

Highlights (for review)

We propose a new complex methodology combined Dynamic Time Warping algorithm to find similar patterns, Stepwise Regression Analysis to find determinants that are most influenced by the stock price, and Artificial Neural Network for predicting.

We use Jaro-Winkler distance with Symbolic Aggregate approXimation (SAX) as prediction accuracy measure to verify our model

We construct a big data processing framework to handle the overall processes using big data open sources.

Pattern graph tracking-based stock price prediction using big data

Seungwoo Jeon[1]

BK21PLUS Creative Human Resource Development Program for IT Convergence

Pusan National University, Busan, South Korea

Bonghee Hong[2,*]

Dept. of Electrical and Computer Engineering

Pusan National University, Busan, South Korea

Victor Chang[3]

Information Management and Information Systems

Xi'an Jiaotong-Liverpool University, Suzhou, China

Abstract

Stock price forecasting is the most difficult field owing to irregularities. However, because stock prices sometimes show similar patterns and are determined by a variety of factors, we propose determining similar patterns in historical stock data to achieve daily stock prices with high prediction accuracy and potential rules for selecting the main factors that significantly affect the price, while simultaneously considering all factors. This study is intended at suggesting a new complex methodology that finds the optimal historical dataset with similar patterns according to various algorithms for each stock item and provides a more accurate prediction of daily stock price. First, we use a Dynamic Time Warping algorithm to find patterns with the most similar situation adjacent to a current pattern. Second,

[*] Corresponding author

Email addresses: i2825t@pusan.ac.kr (Seungwoo Jeon), bhhong@pusan.ac.kr (Bonghee Hong), ic.victor.chang@gmail.com (Victor Chang)

[1] Post Doctoral Researcher

[2] Professor

[3] Professor

we select the determinants most affected by the stock price using feature selection based on Stepwise Regression Analysis. Moreover, we generate an artificial neural network model with selected features as training data for predicting the best stock price. Finally, we use Jaro-Winkler distance with Symbolic Aggregate approXimation (SAX) as a prediction accuracy measure to verify the accuracy of our model.

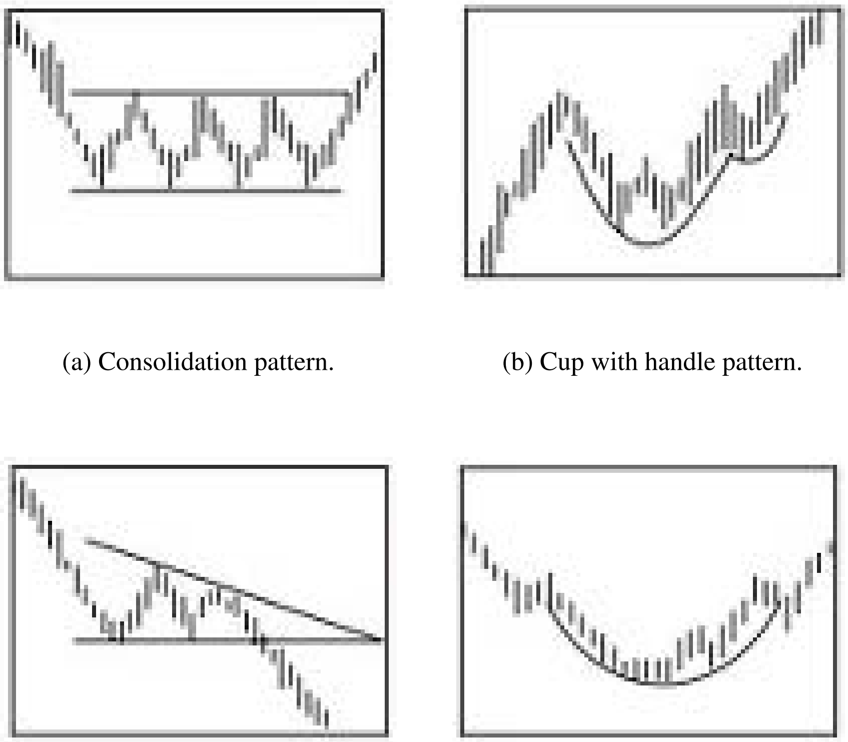
Keywords: Stock price prediction, Dynamic time warping, Feature selection, Artificial neural network, Jaro-Winkler distance, Symbolic Aggregate approXimation

1. Introduction

The stock price received from KOSCOM (Korea Securities Computing Corporation), which provides Korean financial IT solutions, consists of forty items (four groups: domestic and foreign or buying and selling) such as domestic selling high price, foreign selling opening price, and domestic buying trading amount. For example, even if there are stock prices with the same value, their inside combination is different. Domestic selling high price is downturn and domestic buying trading amount is upturn whereas domestic selling high price is upturn and domestic buying trading amount is downturn. Because of very changeable items, the goal is to predict the next stock price pattern graph using these items and for this prediction to be of value.

Analysis and prediction in the stock market are being studied using various methods, such as machine learning and text mining. First, regarding data mining studies using daily stock data, there are prediction research works based on support vector machine (SVM) ([8, 24]) in order to determine whether the new pattern data belongs to a certain pattern category, artificial neural network (ANN) ([31, 32]), to have good prediction even with a complex relationship between the variables, and autoregressive integrated moving average (ARIMA) ([40, 46]) to identify and predict time series variation. Unlike machine learning, there are several prediction research works based on the word analysis of news articles ([29, 38, 39]).

As these research works have predicted daily stock prices using daily closing price, it is not sufficient to make predictions in a time period as short as an hour and a half. Moreover, even if they have analyzed the significance of variables and increased the prediction accuracy of the model by eliminating unimportant variables, the error rates of the prediction are higher because of the use of any data contained in the outliers data.



(a) Consolidation pattern.

(b) Cup with handle pattern.

(c) Double bottom.

(d) Saucer.

Figure 1: Various stock patterns

The stock price graph consists of several patterns such as consolidation, cup with handle, double bottom, and saucer, as shown in Figure 1 [7, 49]. As these patterns repeatedly appear at fixed time intervals, finding a pattern parallel to the current pattern will enable prediction of the following pattern.

Focusing on this point, in this paper, we propose a new method for generating stock price predictions based on historical stock big data. First, unlike existing studies that mostly use closing price data, we use tick-by-tick data for short term prediction and aggregate them to transform non-continuous data to continuous data. Then, through a dynamic time warping algorithm we make some patterns similar to the current pattern and select important features affecting the stock price by using stepwise regression with them. Finally, we generate an artificial neural network using data to be completed in similar patterns and feature selection as input data for high predictive accuracy through learning in order to derive the best results.

Because the pattern size of the stock price is not fixed, it may appear for a short time once and for a long time another time, and the upper and lower sizes may be smaller or larger. In other words, determining singular points of the day is necessary for easily comparing the predicted graph and actual graph. For this, we use a prediction accuracy measure that combines Symbolic Aggregate approXimation (SAX) [36] and Jaro-Winkler distance [47]. This is to recognize the similarity (accuracy) between the predicted graph and actual graph by using the strings transformed from the two approximated graphs.

Thus, we propose a prediction system based on big data processing (Hadoop, Hive, RHive) and analysis (R) tools for next stock price prediction. The system composed of four connected computers includes five steps. As preprocessing, the first step is to transform tick-by-tick data to aggregated data at five-minute intervals to facilitate the prediction and make daily patterns with a five-minute generation unit using Hadoop and RHive query. The second step is to determine all similar patterns over three months by using the dynamic time warping algorithm provided by R function. Then, the system repeatedly removes insignificant variables through stepwise regression on the R function. Next, the system uses an artificial neural network to generate the final prediction model according to numerous simulations.

The main contributions of this paper can be summarized as follows.

- We generate a prediction model for the stock prices by utilizing the artificial neural network through dynamic time warping as the pattern matching algorithm and stepwise regression for the distinction of significant/insignificant variables with real tick-by-tick stock data.

- We evaluate our proposed model through prediction accuracy measure combined SAX and Jaro-Winkler distance for easily comparing singular points of the predicted graph and actual graph.

- To generate automatically predicted stock price, we have built a new system based on big data processing open source tools such as Hadoop and R.

The remainder of this paper is organized as follows. Section 2 presents a background to understanding stock research works and introduces the characteristics of that data as big data. In Section 3, we describe the target environment and define the problem. Sections 4 and 5 describe our new complex methodology and system architecture for handling overall processes, respectively. Section 6 presents our experiments for proving our proposal. In Section 7, we review various existing research works based on stock price forecasting. Section 8 presents a summary of our approach and contributions. Finally, Section 9 concludes the paper.

2. Background and Stock data

In this section, we first introduce various issues for finance, particularly stock prediction, and then, we identify big stock data collected by KOSCOM.

2.1. Background for Stock research

There are major issues related to financial analysis, such as cloud computing, stock prediction, and data security. First, cloud computing has taken center stage in the financial field [10, 12, 13, 35]. Especially, Chang [10] suggested the Heston model based on cloud computing to solve the constraints of the desktop, which calculates asset prices, volatility, etc. in stocks. As the second issue, many related papers have already been published, and stock prices are still being predicted using a variety of methods, such as machine learning and feature selection [22, 42, 43]. However, these methods are suitable for predicting the closing price with low liquidity because the daily data with closing price is not much. Recently, the large scale data processing issue has been actively discussed to overcome the previous limitation [5, 13]. In [5], twitter data (9,853,498 tweets) is used for stock prediction, and Chang et al. [13] have developed Organizational Sustainability Modeling (OSM) that can process enormous amounts of data quickly in finance. Lastly, data security in finance is a subject that is constantly being discussed [41]. They have developed a Cloud Computing Adoption Framework (CCAF) for securing cloud data, and it can protect data real-time and support various functions, such as intrusion prevention and convergent encryption. In this study, we deal with stock prediction in real data and big data processing as large scale data among the issues mentioned above.

2.2. Historical stock data as big data

Usually, to obtain historical and real-time stock price data, we are required to access a website and collect the data directly. However, even if the data are collected real-time, accumulating as much data as we can study in a short period of time is difficult. We were unable to obtain the data of various variables, such as the buying amount of domestic individual investors and selling price of a foreign institution as only the daily closing price in the historical data is available. In this unfavorable situation, we were able to obtain a big dataset of historical stock data from KOSCOM, as detailed in Table 1. They have supported transaction amounts and volume on stock trading from Korea Composite Stock Price Index (KOSPI) as it is called tick-by-tick data. The scale of the data collected is 10∼15 GB per month, and the number of records is approximately 6∼7million per month; the size of total dataset for three months (August 2014 ∼ October 2014) is 50 GB and approximately twenty million.

Table 1: Current details of the historical stock dataset

Classification	**Description**
Data unit	Millisecond interval per transaction
Number of items	200
Number of variables	51
Temporal extent	Three months (August 2014 to October 2014)
Data size per month	10∼15 GB
Number of data per month	6∼7 million

Table 2: Raw stock data as tick by tick

TRADE_DATE	**ISIN_CODE**	**TRD_TM**	**TRD_PRC**	**TRDVOL**	**...**
20141001	KR7005380001	090000001	190000	10	...
20141001	KR7005380001	090000001	190000	3	...
20141001	KR7005380001	090000838	190000	27	...
20141001	KR7005380001	090000984	190000	40	...
...	...	...	...	...	...

However, because there are many unnecessary variables, such as several codes and numbers among the 51 variables mentioned in Table 2, we cannot use them directly. Therefore, this study considers only the price and amount of selling and buying to affect the stock prices and to be used mainly from several studies, as shown in Table 3 [18, 43]. This table is composed of the date, time, item code, type, trade price, trade amount, opening price, high price, and low price. Moreover, there are two types of country type: domestic (code: 00) and foreign, two types of investor type: individual (code: 8000) and institutional, and two

Table 3: Example of stock raw data

Feature	Value
Date (yyyymmdd)	20140813
Time (hhmmssmmm)	090024000
ISIN code	KR7005380001
Country type	00
Investor type	8000
Trading type	BID
Trade price (won)	77,500
Trade amount	37
Opening price (won)	78,900
High price (won)	78,900
Low price (won)	76,600

types of trading type: buying (code: BID) and selling (code: ASK); stock price is the total sum of forty features.

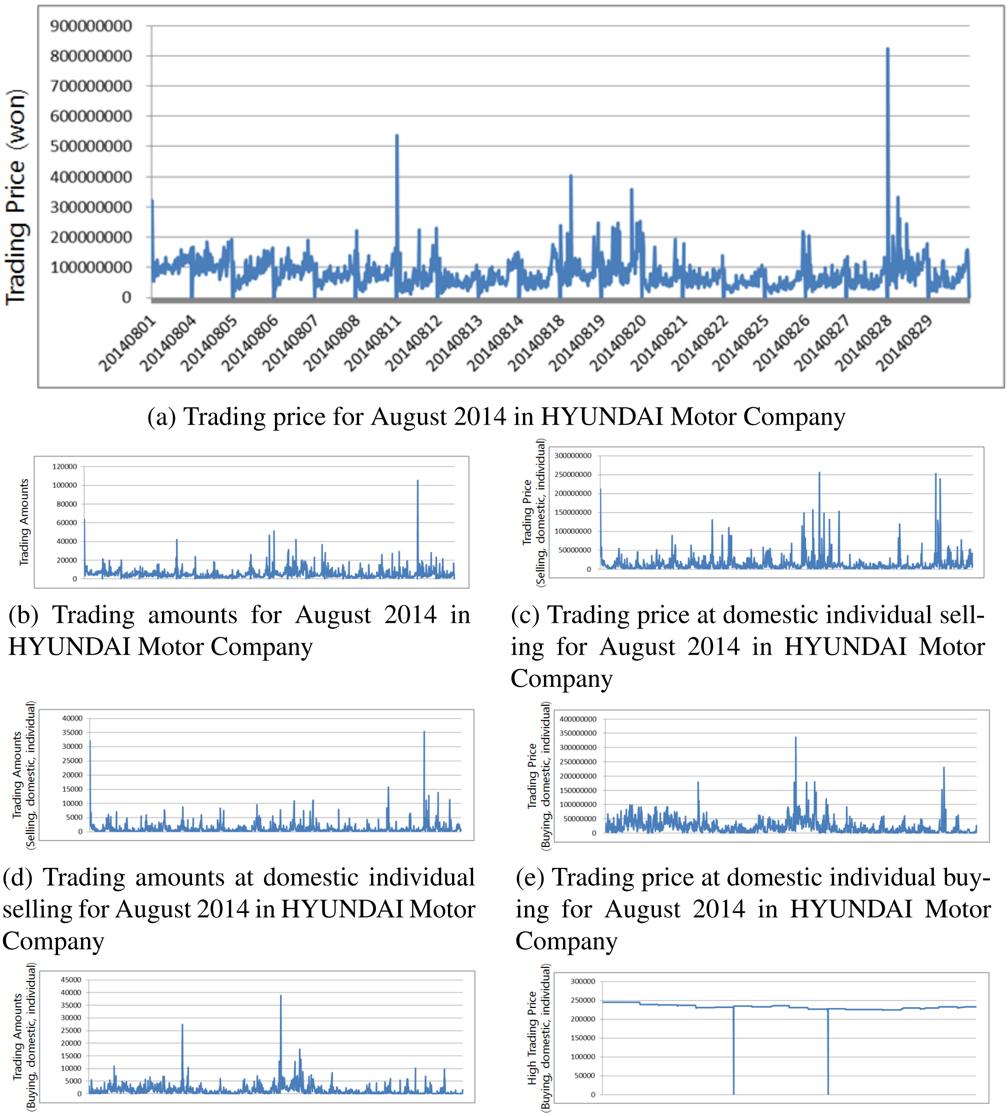
3. Target environment and Problem definition

In this section, we first introduce a target environment to predict stock price, and then we present the importance and problem of data selection faced in the past.

3.1. Target environment

The stock trading service aims to determine whether the price of a specific item increases or decreases and to make the maximum profit through buying and selling at reasonable prices. In this time, when we predict the stock price graph of the following day, assessing the current situation, such as a sharp fall in the trading price and gradual increase in the trading amounts, is necessary. Figure 2 describes the trading price and several features to comprise it in five-minute intervals during one month (August 2014) in HYUNDAI Motor Company. For example, at first sight, the trading prices between two days (August 6 and 29, 2014) are similar, as shown in Figure 2 (a). However, they are different in Figure 2 (b)$\sim$(f). Further, Figure 2 (g) does not seem to significantly affect the trading price.

Therefore, to predict the stock price of the following day, as shown in Figure 3, we need to use historical stock big data generated by the transaction for retrieving a similar situation as the current, unlike [1, 21, 43] that predict the one day future



(a) Trading price for August 2014 in HYUNDAI Motor Company

(b) Trading amounts for August 2014 in HYUNDAI Motor Company

(c) Trading price at domestic individual selling for August 2014 in HYUNDAI Motor Company

(d) Trading amounts at domestic individual selling for August 2014 in HYUNDAI Motor Company

(e) Trading price at domestic individual buying for August 2014 in HYUNDAI Motor Company

(f) Trading amounts at domestic individual buying for August 2014 in HYUNDAI Motor Company

(g) High trading price at domestic individual buying for August 2014 in HYUNDAI Motor Company

Figure 2: Trading price consists of various types of features

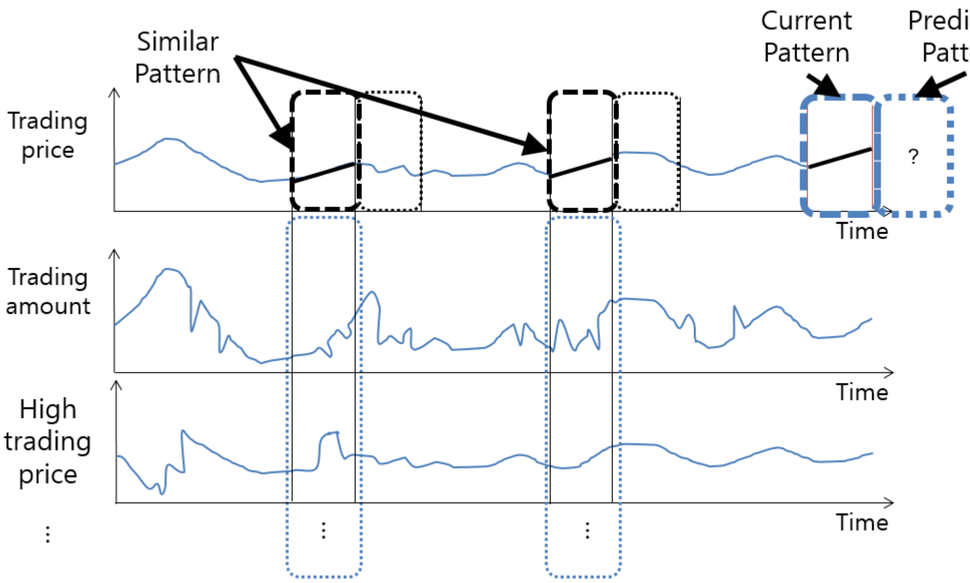


Figure 3: Predicted pattern will be generated from similar patterns of historical stock data.

closing price of individual stocks using daily stock prices composed of small data. In other words, because the trading price consists of several features, such as trading amount, high price, and low price, this study is intended at finding the most similar pattern to a combination of features among historical stock data and predicting the stock price by using them. In addition, to determine time range to be predicted in this study, defining the big data stock price prediction is necessary; see Definition 3.1.

Definition 3.1: Big data stock price prediction

The big data stock price prediction is forecasting future stock price in the same period and the item using patterns similar to the current pattern from among a huge volume of historical stock data.

3.2. Problem in use of historical data

As previously discussed, because the stock data consists of several features, if historical data are used as input for training in a prediction method without considering the relation among them just used, it might lead to large residuals between the real and predicted data. For example, as shown in Figure 4, given a current pattern with one day size in the trading price of an item, if historical stock data for a certain period of time are used when stock pattern of the following day is generated, a gap between the real and predicted stock graphs increases owing to outliers, such as sharply falling situation; these outliers have been studied in several papers [2, 14].

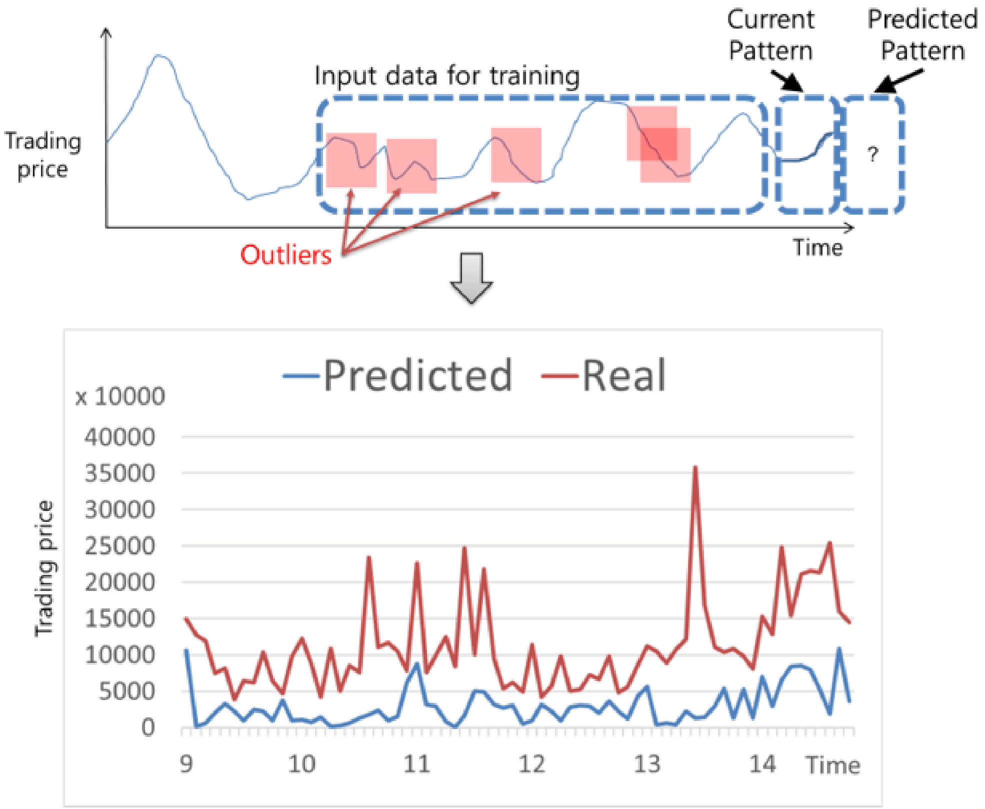


Figure 4: Example of problem in the selection of historical data.

In this paper, the problem mentioned above can be defined as in Definition 3.2

Definition 3.2: Selection criteria absence of input data for prediction

Among historical stock data with various stock price pattern graphs, the retrieval of specific criteria for selecting optimal historical data that can increase the prediction accuracy is defined as the selection criteria absence of input data for prediction.

Definition 3.2 is also expressed by a formula as follows.

$$f^* = \underset{f}{\text{argmin}} \sum^{i} \sum^{j}_{t_{i,j} \in T} L(S^{d-\gamma}_{t_{i-\alpha,j-\beta}}, forecast(s^{d}_{t_{i,j}})) \tag{1}$$

where f^* is finding suitable historical data with the least loss between the real and predicted data, $\underset{f}{\text{argmin}} f(x)$ means a function that finds x value to minimum $f(x)$. In addition, as $S^{d}_{t_{i,j}}$ is a historical dataset, it can be expressed as $\{S^{d}_{t_{i,j}}$, $S^{d-1}_{t_{i,j}}$, $S^{d-2}_{t_{i,j}}$, $\ldots$, $S^{d-n}_{t_{i,j}}\}$, t_i is hour, t_j is minute and d is day. $forecast$ function means a prediction method to use historical data as input, and L function is a loss method between the real and predicted data.

4. Outline of the proposed model

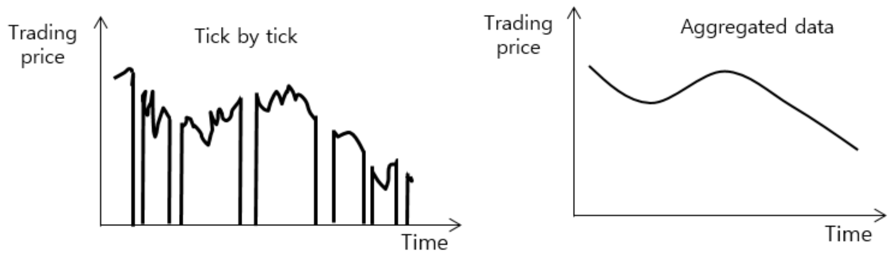
In this section, we describe the overall process, from data preprocessing for making continuous data, retrieving data with similar patterns and selecting input data, to the generation of the prediction model from the perspective of data analysis and processing.

4.1. Data Preprocessing: Aggregation of stock data

We have tick-by-tick data received from KOSCOM. Because the data is generated per transaction in a very short time, the trading price of tick by tick is zero at the time if the transaction is not carried out, as shown in Figure 5 (a) and eventually, the data is discrete data not continuous data, it is difficult for the data to predict stock price. In other words, it is necessary to transform discrete data into continuous data without zeros rather than using discrete data as raw data considering zero for easily predicting the stock price. Consequently, we generate aggregated data at five-minute intervals to revise the continuous flow of data shown in Figure 5 (b).

4.2. Stock Pattern generation with Sliding Window

Because a stock graph shows similar patterns aperiodically, we should find the dataset with similar patterns in big historical stock data. For this, above all, it is necessary to make patterns from aggregated data. Figure 6 shows the processes of patterning the aggregated data. The length of a pattern is one day, and patterns are



(a) Trading price per transaction. (b) Trading price after aggregation.

Figure 5: The need for aggregation in raw stock data

generated at five-minute intervals, e.g., by the sliding window method, for pattern matching analysis using various patterns. The number of patterns for one hour will be twelve.

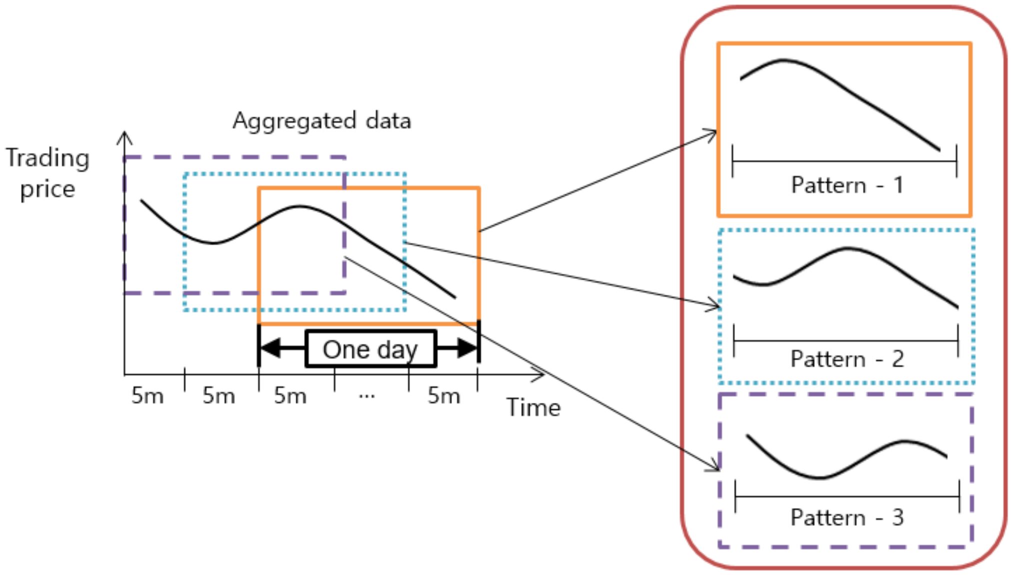


Figure 6: Method of patterning the aggregated data.

4.3. Pattern Retrieval using Dynamic Time Warping

Figure 7 shows seven similar patterns (dotted lines) and one current pattern (solid line) using Dynamic Time Warping in the graph of real stock price. The similar patterns can be found by comparing historical patterns and the current pattern. There are various methods for pattern matching such as Euclidean distance, Dynamic Time Warping (DTW)([4]), Edit Distance with Real Penalty (ERP)([15]), Longest Common Subsequence (LCSS)([44]) and Edit Distance on Real Sequence (EDR)([16]). We use a hierarchical clustering algorithm based on Euclidean distance for finding similar patterns quickly and simultaneously in a previous paper ([27]). However, because the Euclidean distance method does not accurately identify trading price trends owing to the limitation that the i^{th} point in a sequence should be calculated with the i^{th} point in the other, we find similar patterns using the DTW method that accurately identify trading price trends than anything else ([3, 17]).

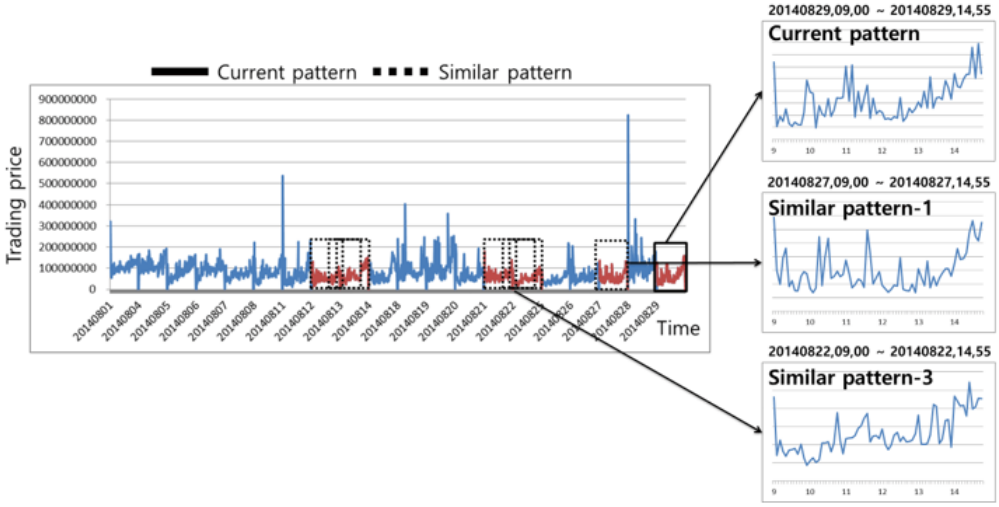
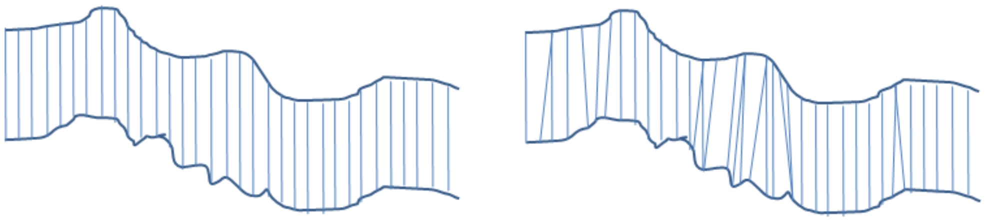


Figure 7: Similar stock patterns.



(a) Comparison of two data according to Euclidean

(b) Comparison of two data according to Dynamic Time Warping

Figure 8: Difference of Euclidean and Dynamic Time Warping

Figure 8 depicts the difference of Euclidean distance and DTW. Whereas an i^{th} point in one graph indicates the i^{th} point as the same location in the other graph at

Euclidean distance, an i^{th} point in one graph connects several points in the other at DTW. As an extreme example, given a sine curve and a cosine curve, when calculating the Euclidean distance, as the distance between two curves is big, the two curves are likely to be different patterns. Whereas, if using DTW, then the two curves are likely to be similar patterns. By doing so, it will be easy to find a pattern having a similar situation for the current pattern.

4.4. Feature Selection based on Stepwise Regression Analysis

After finding similar historical patterns with current trading price pattern, deciding determinants that are the most affected by the trading price and retrieving optimal historical pattern dataset using them is necessary. While this seems to be the same pattern between current and historical trading price, the relationship of the determinants may be different. For example, if the current trading price is 10,000 won owing to selling by institutional investors and historical trading price was 10,000 won owing to selling by individual investors, and they cannot be seen as the same.

In this paper, we choose main determinants using stepwise regression in both current and historical patterns for finding optimal historical pattern dataset; this is called “Feature selection”. Before creating each regression model, it must be normalized, because the units of total forty determinants are different. For example, whereas the unit of the amount is a number, the units of the price are won, dollar, and yen; the units of the determinants are made into one for making the comparison of determinants possible. Although there are various data transformation techniques, such as z-transform, log transform, and re-scaling range to [0,1], we use re-scaling range in R because it was often used for continuous data, as shown below, and the two tables, Table 4 and 5 show raw and transformed data.

```
x_i <-(x_i-min(x_i) /(max(x_i)-min(x_i))
```

Given the forty normalized determinants, we create a regression model using them. In this work, we consider the trading price as a dependent variable and the forty determinants as independent variables in the regression analysis, which is provided as two functions in R, as shown below. Above all, we use *lm* function to fit a linear model. y is a dependent variable, and x_1 to x_{40} are independent variables.

```
fit <- lm(y ~x_1+x_2+x_3+...+x_40, data=stock_data)
```

After fitting, we also use the step function in R for determining the final independent variables, the first factor represents the linear model and the second

Table 4: Before data transformation

	Total forty variables →				
	TRD_PRC	**TRDVOL**	**HIGH_PRICE in ASK, Individual and Domestic**	**LOW_PRICE in BID, Institutional and Foreign**	**...**
One thousand five hundreds per one month ↓	3509500	9	222000	218500	...
	4388000	147	222000	218500	...
	1533000	1221	218500	217000	...
	...	...	...	...	...

Table 5: After data transformation

	Total forty variables →				
	TRD_PRC	**TRDVOL**	**HIGH_PRICE in ASK, Individual and Domestic**	**LOW_PRICE in BID, Institutional and Foreign**	**...**
One thousand five hundreds per one month ↓	0.234579665	0.239762641	0.785714286	0.770877944	...
	0.487807624	0.090879625	0.785714286	0.770877944	...
	0.184445404	0.748927895	0.781512605	0.773019272	...
	...	...	...	...	...

factor determines the direction of the stepwise process combining forward and backward.

```
bidirectional <- step(fit, direction="both")
```

The procedure is organized as follows.

- Repeatedly add and remove a variable among all variables, and then conduct regression analysis with remainder.
- Select the final variable association with the highest value of R-Square as an explanatory power of regression model.

In the current pattern of Hyundai Motor Company, 15 variables remained after applying stepwise regression, as can be seen in Table 6.

Table 6: Result of stepwise regression in real stock data of Hyundai Motor Company

	ASK				BID			
	Domestic		Foreign		Domestic		Foreign	
	Indiv.	Insti.	Indiv.	Insti.	Indiv.	Insti.	Indiv.	Insti.
Trading Price	O	X	O	X	O	O	O	O
Trading Volume	O	X	X	O	O	O	O	X
High Price	X	X	X	X	X	X	X	X
Low Price	X	X	O	O	X	O	O	X
Opening Price	X	X	X	X	X	X	X	X

4.5. Predicted Stock data generation using Artificial Neural Network

Through the comparison of important determinants in both current and historical patterns, we select an optimal historical dataset, and use it for the input data of Artificial Neural Network. First, we calculate and compare the leverage of determinants in each regression model using lm.beta function provided by R. In this time, it is determined by the number of same elements equal to or greater than the threshold value of leverage; we can check the results in Table 7.

Table 7 shows the number of matching numbers existing between current and historical patterns to assess the similar pattern. Because the matching number in the second historical pattern is just two whereas that in the first historical pattern is four, we select the first historical pattern as the optimal historical dataset.

After selecting the historical dataset, to generate the predicted stock data, we use an artificial neural network algorithm because it is the most widely used in stock price forecasts ([9, 28, 33]) and has a high predictive power as an advantage over learning by iterative adjustment. Here, the optimal historical dataset will be used for training data as input data in the artificial neural network. As shown in the R code below, an ANN model is created with one dependent and four independents using neuralnet package ([20]).

```
neural <- neuralnet(TRD_PRC ~ BID_DOM_INDIV_TRD_PRC
+ BID_DOM_INSTI_TRD_PRC + BID_FOR_INDIV_TRD_PRC
+ BID_FOR_INSTI_TRD_PRC, data = training_data, hidden=3)
```

Because the units of input data are converted to [0,1] transform, the result of ANN must also be converted to the previous unit. As shown in the R code below, we generate predicted stock data using test data and ANN model based on optimal historical data. Then, it converts the unit of the predicted data to the original unit.

```
neural_results <- compute(neural, test_data)
```

Table 7: Selection of optimal historical dataset

(a) Calculation of determinants leverage

Leverage of determinants in current pattern		**Leverage of determinants in historical pattern - 1**		**Leverage of determinants in historical pattern - 2**	
Determinant	Leverage	Determinant	Leverage	Determinant	Leverage
BID_DOM_INSTI_TRD_PRC	0.340	BID_DOM_INSTI_TRD_PRC	0.318	BID_DOM_INSTI_TRD_PRC	0.000
BID_DOM_INSTI_TRDVOL	0.000	BID_DOM_INSTI_HIGH_PRICE	-0.000	BID_DOM_INSTI_TRDVOL	0.000
BID_DOM_INSTI_LOW	0.000	BID_DOM_INSTI_LOW	-0.000	BID_DOM_INSTI_HIGH_PRICE	-0.000
ASK_FOR_INSTI_TRDVOL	0.000	ASK_FOR_INSTI_TRD_PRC	0.549	ASK_FOR_INSTI_TRD_PRC	0.000
ASK_FOR_INSTI_LOW	-0.000	ASK_FOR_INSTI_TRDVOL	-0.000	ASK_FOR_INSTI_TRDVOL	-0.000
BID_FOR_INDIV_TRD_PRC	0.011	BID_FOR_INDIV_TRD_PRC	0.038	ASK_FOR_INSTI_LOW	-0.000
BID_FOR_INDIV_TRDVOL	-0.000	BID_FOR_INDIV_LOW	0.000	BID_FOR_INDIV_TRD_PRC	0.005
BID_FOR_INDIV_LOW	0.000	BID_DOM_INDIV_TRD_PRC	0.585	BID_FOR_INDIV_HIGH	-0.000
BID_DOM_INDIV_TRD_PRC	0.794	ASK_DOM_INDIV_TRD_PRC	0.687	BID_DOM_INDIV_TRD_PRC	0.000
BID_DOM_INDIV_TRDVOL	-0.000	ASK_DOM_INDIV_TRDVOL	-0.000	ASK_DOM_INDIV_TRD_PRC	0.000
ASK_DOM_INDIV_TRD_PRC	0.000	ASK_FOR_INDIV_TRD_PRC	-0.000	ASK_DOM_INDIV_TRDVOL	-0.000
ASK_DOM_INDIV_TRDVOL	-0.000	ASK_FOR_INDIV_TRDVOL	-0.000	BID_FOR_INSTI_TRD_PRC	0.471
ASK_FOR_INDIV_TRD_PRC	0.000	ASK_FOR_INDIV_HIGH	0.000	**Total number of determinants**	**12**
ASK_FOR_INDIV_LOW	-0.000	ASK_FOR_INDIV_LOW	-0.000		
BID_FOR_INSTI_TRD_PRC	0.389	BID_FOR_INSTI_TRD_PRC	0.713		
Total number of determinants	**15**	**Total number of determinants**	**15**		

(b) Final leverage applied by threshold

Leverage of determinants in current pattern		**Leverage of determinants in historical pattern - 1**		**Leverage of determinants in historical pattern - 2**	
Determinant	Leverage	Determinant	Leverage	Determinant	Leverage
BID_DOM_INSTI_TRD_PRC	0.340	BID_DOM_INSTI_TRD_PRC	0.318	ASK_FOR_INSTI_TRD_PRC	0.549
BID_FOR_INDIV_TRD_PRC	0.011	BID_FOR_INDIV_TRD_PRC	0.038	BID_FOR_INDIV_TRD_PRC	0.005
BID_DOM_INDIV_TRD_PRC	0.794	BID_DOM_INDIV_TRD_PRC	0.585	ASK_DOM_INDIV_TRD_PRC	0.687
BID_FOR_INSTI_TRD_PRC	0.389	BID_FOR_INSTI_TRD_PRC	0.713	BID_FOR_INSTI_TRD_PRC	0.471
Total number of determinants	**4**	**Matched number of determinants**	**4**	**Matched number of determinants**	**2**

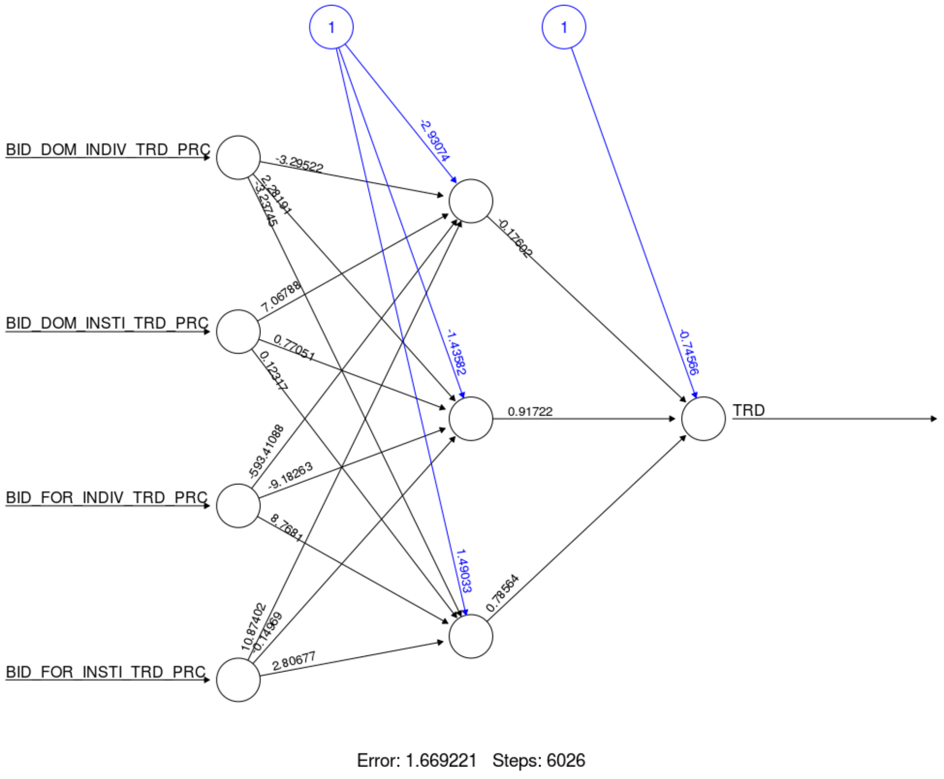
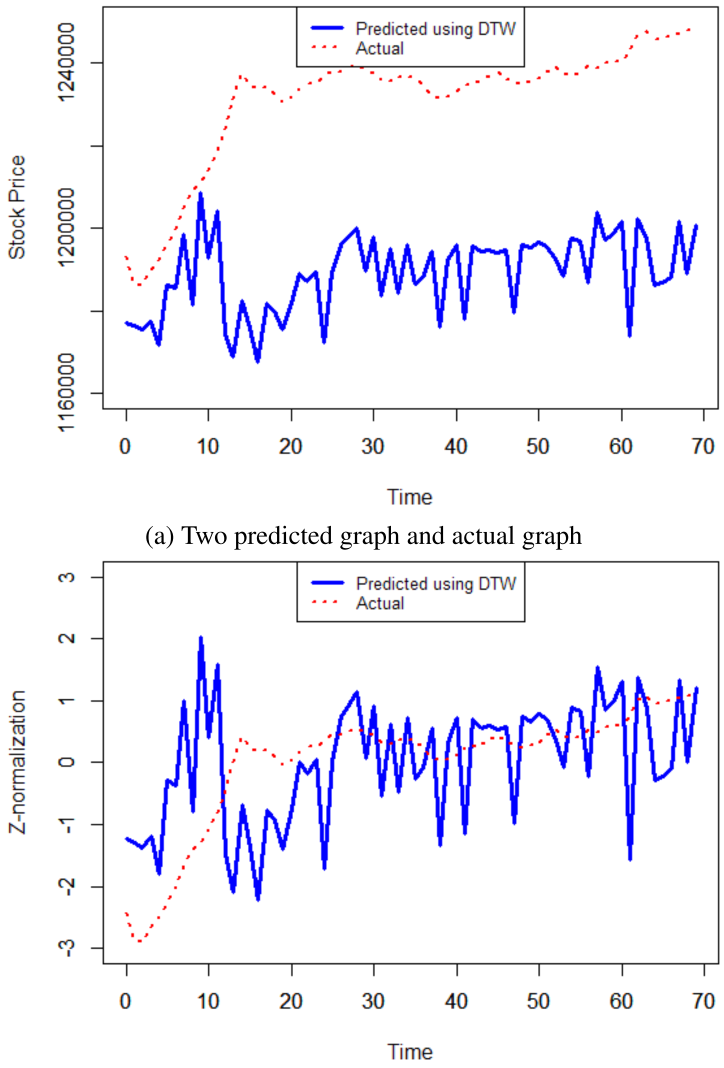


Figure 9: Graphical representation of ANN model with hidden layer 3

```
final_ANN_results <- neural_results$net.result*(max(x)-min(x))+min(x)
```

4.6. *Prediction accuracy measure using combined SAX and Jaro-Winkler distance*

There are various measures such as Beta (β), Standard Error, Mean squared error (MSE), R-squared value, Mean absolute percentage error (MAPE) and Root Mean Square Error (RMSE) for checking prediction accuracy in finance. Typically, the RMSE is used as a barometer of prediction accuracy by using differences between the predicted and real data [11, 26]. However, in this study, we use a combination of SAX and Jaro-Winkler distance as the new prediction accuracy measure to approximately find singular points of patterns because the pattern does not know when and where it occurs.



(a) Two predicted graph and actual graph

(b) Two transformed graphs

Figure 10: Z-normalization for comparison

This measure consists of two methods, SAX and Jaro-Winkler distance. As the SAX is used to transform time-series data into an approximated string, it consists of three steps and Figure 10 $\sim$ 11 show a good example of how the two graphs are similar although the interval between the predicted and actual data is large. First, it transforms the time-series data into the normalized time-series data to pick equal-sized areas in Figure 10. Second, it transforms the normalized data into the Piece-wise Aggregate Approximation (PAA) [48] representation as shown in Figure 11, and finally, it converts the PAA data into a string as follows.

```
Predicted data: adabcccddddd
Real data: aaccddcdddde
```

Then, the Jaro-Winkler distance is used to compare two transformed strings and calculate similarity(accuracy). As this distance is two complexed distance measures, Jaro distance is expressed by a formula as follows.

$$distance_j = \begin{cases} 0 & \text{if } m = 0 \\ \frac{1}{3}\left(\frac{m}{|s_1|} + \frac{m}{|s_2|} + \frac{m-t}{m}\right) & \text{otherwise} \end{cases} \tag{2}$$

where s_1 and s_2 are strings to compare, m is the number of matching characters and t is half the number of transpositions. Eventually, the Jaro-Winkler distance is expressed by a formula as follows.

$$distance_{jw} = distance_j + l * p(1 - distance_j) \tag{3}$$

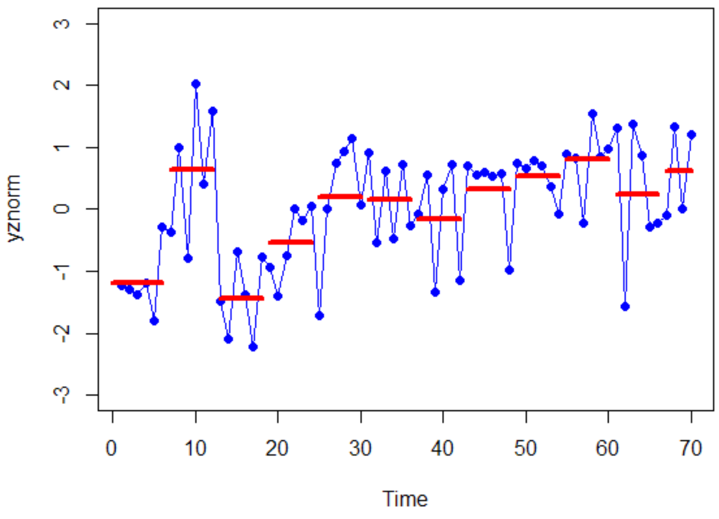
where l is the length of common prefix at the start of the string up to a maximum of 4 characters and as p is a constant scaling factor, the standard value for this constant is p=0.1. Through this distance, we have obtained the similarity from the next formula.

$$similarity = 1 - distance_{jw} \tag{4}$$

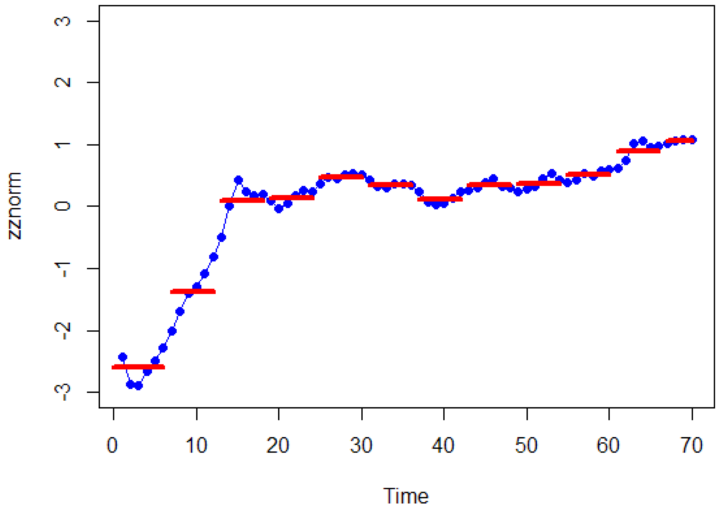
According to the above formulae, the similarity of the two strings mentioned above is 88.18%.

5. System architecture for stock price prediction

This section describes the series of operations performed for generating the final artificial neural network model. All processes are performed on a cluster composed of four connected computers (one master and three slaves) with Hadoop and RHive installed.



(a) Transformation of predicted data into PAA



(b) Transformation of actual data into PAA

Figure 11: PAA transformation

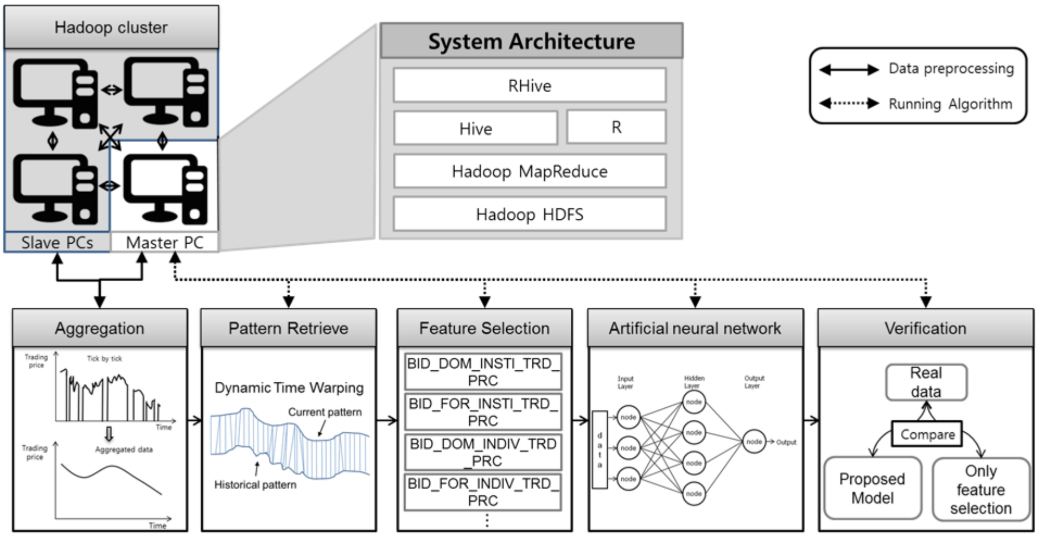


Figure 12: System Architecture

5.1. Hardware

As shown in Figure 12, our prediction system consists of four PCs that form a private cloud; one PC is the master and the others are slaves. Each PC has the same specifications, which are shown in Table 8, i.e., 1 TB hard disk, 3.10 GHz Intel Xeons CPU, 8 GB RAM, and Community ENTerprise Operating System (CentOS).

Table 8: Hardware specification

Classification	Specification
Number of computers	4 EA
Hard disk space	1 TB
CPU	3.10 GHz Intel Xeons
RAM	8 GB
OS	CentOS

5.2. Series of operations for generating predicted stock data

We suggest performing the following steps for generating a prediction model for big data processing and analysis tools, as shown in Figure 12.

Step 1 (Stock data aggregation and pattern generation as data preprocessing): We store stock data in Hadoop Distributed File Systems (HDFSs) of the

Hadoop-based clusters. Because we cannot manually modify the source code of MapReduce for extracting the exact data that is desired from each HDFS of the Hadoop cluster, we use a RHive tool so that HiveQL can be used to assist the search of the desired data similar to the select query of RDBMS. The reason for selecting RHive rather than RDBMS is to ensure insert and search performance in batches because the stock data for three months provided by the KOSCOM contains over seven million records, which will be stored as huge data in future real time collection. The HiveQL queries such as CREATE, LOAD and SELECT are as follows.

```
rhive.query(CREATE TABLE STOCK_PREDICTION
(TRADE_DATE STRING, BLKTRD_TP_CD STRING,
REGUL_OFFHR_TP_CD STRING, ISIN_CODE STRING, JONG_INDEX INT,
TRD_NO INT, TRD_PRC FLOAT, TRDVOL INT, TRD_TP_CD STRING,
TRD_DD STRING, TRD_TM STRING, NBMM_TRD_PRC FLOAT,
FUTRMM_TRD_PRC FLOAT, BID_MBR_NO STRING,
BIDORD_TP_CD STRING, BID_TRSTK_STAT_ID STRING,
BID_TRSTK_TRD_METHD_CD STRING, BID_ASK_TP_CD STRING,
BID_TRST_PRINC_TP_CD STRING, BID_TRSTCOM_NO STRING,
BID_PT_TP_CD STRING, BID_INVST_TP_CD STRING,
BID_FORNINVST_TP_CD STRING, BIDORD_ACPT_NO INT,
ASK_MBR_NO STRING, ASKORD_TP_CD STRING,
ASK_TRSTK_STAT_ID STRING, ASK_TRSTK_TRD_METHD_CD STRING,
ASK_ASK_TP_CD STRING, ASK_TRST_PRINC_TP_CD STRING,
ASK_TRSTCOM_NO STRING, ASK_PT_TP_CD STRING,
ASK_INVST_TP_CD STRING, ASK_FORNINVST_TP_CD STRING,
ASKORD_ACPT_NO INT, OPEN_PRICE FLOAT, HIGH_PRICE FLOAT,
LOW_PRICE FLOAT, LST_PRC FLOAT, ACC_TRDVOL INT,
ACC_AMT FLOAT, LST_ASKBID_TP_CD STRING, LP_HD_QTY INT,
DATA_TYPE INT, MSG_SEQ INT, BID_PROGM_ORD_DECL_TP_CD STRING,
ASK_PROGM_ORD_DECL_TP_CD STRING, BRD_ID STRING,
SESSION_ID STRING, DYNMC_UPLMTPRC INT, DYNMC_LWLMTPRC INT)
PARTITIONED BY (TRADE_DATE STRING)
ROW FORMAT DELIMITED FIELDS TERMINATED BY ′,′
LINES TERMINATED BY ′\n′)
```

```
rhive.query(LOAD DATA LOCAL INPATH
′KOSPI/201408/*.txt′ OVERWRITE INTO
TABLE STOCK_PREDICTION PARTITION(DATE= ′201408′)
```

```
rhive.query(SELECT * FROM STOCK_PREDICTION
WHERE ISIN_CODE='KR7005380001' AND TRADE_DATE LIKE '201408%'
ORDER BY TRADE_DATE ASC)
```

After extracting the data, they are aggregated at five-minute intervals using R because of the use of tick-by-tick data. Then, patterns are generated from them by the concatenation of similar patterns in R from the master computer. The pattern is created by using one-day data generated at five-minute intervals.

Step 2 (Pattern selection with dynamic time warping): To retrieve similar patterns using the current pattern, we use dtw for dynamic time warping in the dist function that measures a distance in R; this has two important advantages in that it does not require an unnecessary comparison operation irrelevant to the current pattern rather than hierarchical cluster supporting Euclidean distance and it can more accurately detect similar patterns.

Algorithm 1: Algorithm for pattern selection

```
input  : List of aggregated patterns $P_A$ and a current pattern $P_C$
output: List of similar patterns $P_S$ after the 'DTW' method is applied
1 Initialize $last$ to size of aggregated patterns $P_A$;
2 for $i = 1 \rightarrow last$ do
3     Calculate distance $D_i$ based on Dynamic Time Warping between $i^{th}$
        aggregated pattern $P_{A_i}$ and current pattern $P_C$;
4     Add $D_i$ in array of integer $V_{DTW}$;
5 Sort $V_{DTW}$ in ascending order;
6 Extract top-ten patterns of sorted $V_{DTW}$ to $P_S$ as the most similar patterns;
7 return $P_S$;
```

Algorithm 1 describes how to find the top ten similar patterns. After inserting the current pattern into the aggregated patterns as historical dataset, we calculate the dtw-based distance between the current pattern and a pattern generated by the sliding window method (Line 2 $\sim$ 4). Then, we select the top ten patterns with the smallest difference (Line 5 $\sim$ 6). The actual code in R is as follows.

```
$P_A$.dist <- dist($P_A$, method='DTW')
```

Step 3 (Feature selection using stepwise regression): Given a current pattern of stock price, insignificant variables from all variables composed of the price are removed.

Algorithm 2: Algorithm for feature selection of the current pattern using stepwise regression

```
  input  : A current pattern $P_C$
  output: List of remaining variables excluding the insignificant variables
          $Var$
1 Transform variables of current pattern $P_C$ to normalized $NM_C$;
2 Extract $STD_C$ from remaining variables $Var$ using stepwise regression
    method;
3 Initialize $len$ to the size of remaining variables $Var$;
4 for $i = 1 \rightarrow len$ do
5     if P-value of $i^{th}$ Var is more than 0.05 then
6         break;
7     else
8         flag = true;
9 return $Var$;
```

Algorithm 2 describes the steps for feature selection of the current pattern using stepwise regression. First, before stepwise regression, all variables are normalized as the units of total 40 variables are different (Line 2). Then, insignificant variables among the variables of the current pattern are removed using stepwise regression (Line 3). The remaining variables with p value below a specified threshold are considered to be significant variables (Line 5 $\sim$ 8). The actual code in R is as follows.

```
$Var_{final}$ ← step($Var_{initial}$, direction='both')
```

After removing insignificant variables from the current pattern, optimal historical dataset should be retrieved using them. To do that, it is necessary to calculate and compare leverage of determinants in each regression model with current and historical(=similar) patterns. If a historical pattern has the same or greater number of elements than the threshold value of leverage compared with the current pattern, we consider it to be the optimal dataset.

Algorithm 3 describes the comparison of leverage between current and historical patterns. First, beta values as leverage of the regression model are calculated using the remaining variables generated from Algorithm 2 (Line 1). Moreover, similar to Algorithm 2, all variables of each similar pattern are normalized to generate regression models (Line 4). After creating each stepwise regression model

Algorithm 3: Algorithm for optimal historical dataset selection using standardization

```
input  : List of remaining variables of current pattern $Var$ and a list of
         similar patterns $P_S$
output: A list of optimal patterns $P_O$ after beta comparison

1 Create Beta value $P_C.beta$ calculated from the remaining variables $Var$
    using beta function;
2 Initialize $len$ to size of similar patterns $P_S$;
3 for $i = 0 \to len$ do
4     Transform variables of $i^{th}$ similar pattern $P_{S_i}$ to normalized $NM_{S_i}$;
5     Create $stepmodel_i$ using stepwise regression with $NM_{S_i}$;
6     Calculate Beta value $P_{S_i}.beta$ calculated from $stepmodel_i$;
7     Compare $P_C.beta$ and $P_{S_i}.beta$;
8 return $P_O$;
```

of similar patterns, two beta values of the current pattern and a similar pattern are compared one by one (Line 5 ∼ 7).

Step 4 (Predicted data generation using artificial neural network): To create predicted data, we use an artificial neural network after feature selection. Algorithm 4 describes the steps for the generation of predicted data using an ANN method. From the input data, we determine dependent and independent variables as training data for another time zone because we predict the next day data for the current pattern (Line 1). This means that given historical time ht of similar patterns, the time of the dependent variable is $ht+1$ and the time of the independent variable is ht. After independent and dependent variables are bound, we generate an ANN-based model using the neuralnet function provided in R (Line 2). Then, independent variables at current time t in the model are input and predicted data are generated (Line 3). The actual code in R is as follows.

```
$TR$ ← cbind($TR_{dep}$, $TR_{indep}$)
colnames($TR$) ← c('output','input')
$ANN$ ← neuralnet(output∼input, training, hidden=3)
$PRD$ ← compute($ANN$, $TEST_{indep}$)
```

Step 5 (Verification): To verify our proposed model, we combine two methods such as SAX and Jaro-Winkler distance, the functions of which are also provided in R. The measure is computed from the comparisons between real and predicted data. As shown in R code below, two raw time-series data (predicted

Algorithm 4: Algorithm for the generation of predicted data

```
input : Total trading price $TR_{dep}$ as training data, reminding variables
        $TR_{indep}$ excluding the total trading price as training data, and
        reminding variables $TEST_{indep}$ excluding the total trading price as
        test data
output: A predicted trading price $PRD$ generated by an ANN

Bind dependent $TR_{dep}$ and independent $TR_{indep}$ variables;
Run artificial neural network $ANN$ with bound variables;
Create predicted trading price $PRD$ according to $ANN$ with test variables
  $TEST_{indep}$;
return $PRD$;
```

by DTW and real data) are transformed into z-normalized time-series data first. Then, after dividing them into twelve sections, each character is converted to five strings using the series_to_string as string conversion function. Finally, a similarity between the two strings is calculated using the Jaro-Winkler distance function.

```
KR7005930003_pre_znorm = znorm(KR7005930003_pre)
KR7005930003_rea_znorm = znorm(KR7005930003_rea)
paa_size=12
s1_paa = paa(KR7005930003_20141027_pre_znorm, paa_size)
s2_paa = paa(KR7005930003_20141027_rea_znorm, paa_size)
str1 = series_to_string(s1_paa, 5)
str2 = series_to_string(s2_paa, 5)
stringsim(str1, str2, method=’jw’, p=0.1)
```

6. Evaluation

In this section, we describes the test data provided by KOSCOM for three months and evaluate the accuracy of each stock item by computing SAX and Jaro-Winkler distance.

6.1. Dataset and test scenario

To prove our proposed model, we used real historical stock dataset composed of various items for three months from August 2014 to October 2014. To measure the prediction accuracy, we prepared three items (Hyundai Motor, KIA Motor, and Samsung Electronic) as companies representing the Republic of Korea, the stock data for August 1, 2014 to October 26, 2014 as training data, and the stock data for October 27 to 31, 2014 as test data. As a test scenario, first, two predicted stock data for one day were generated according to our proposed model and feature selection. Then, we checked the prediction accuracy using the Jaro-Winkler distance values and comparing the predicted and real stock data.

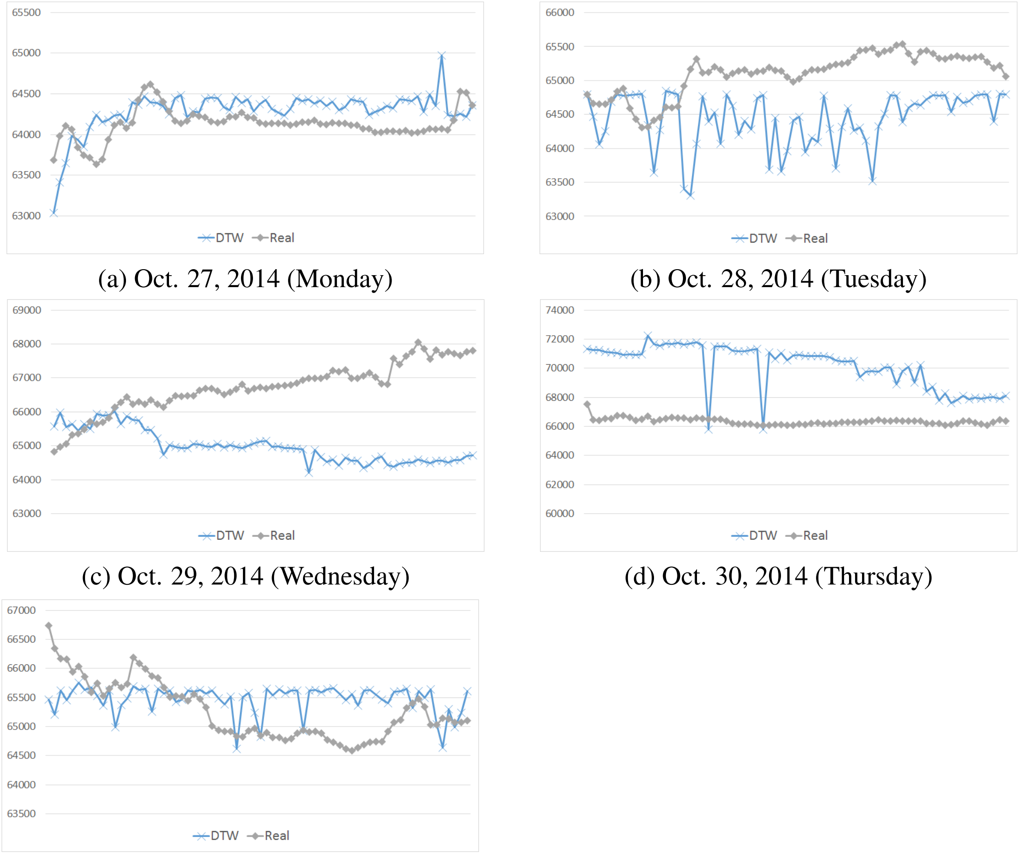
6.2. Evaluation of the prediction accuracy

We performed experiments to verify the accuracy of our proposed method. Figure 13 $\sim$ 15 describe the comparison results of actual and predicted data by our proposed model for the stocks of LG Electronics, Samsung Electronics, and Hyundai Motor for five days (October 27 to October 31, 2014), which was considered as one week. The x-axis shows the time at five-minute intervals and the y-axis shows the trading price called stock price according to the time. In addition, a final table of each figure shows the measured Jaro-Winkler Similarity and RMSE.

From Figure 13 (a), we can directly see that our predicted and real values are very similar, whereas Figure 13 (c) shows a completely different trend. Figure 13 (b) and (d) show a large difference, but a slightly similar trend. In Figure 13 (e), the predicted graph does not show a large difference between the opening and closing prices, while the real graph goes down. Comparing the Jaro-Winkler Similarity and RMSE in all graphs in Figure 13 (f), as mentioned above, the similarity results as per the Jaro-Winkler Similarity are quite good. The result of RMSE is good; however, the trends of the two graphs are different on Oct 31. On the other hand, the result of RMSE is the worst with only a slightly similar trend on Oct. 30.

Figure 14 shows the stock data derived from the real and predicted data for the Samsung Electronics Company. As shown in Figure 13, the more similar the trend, the higher the similarity in all graphs. In particular, although Figure 14 (d) shows a big difference between the predicted and actual data, the Jaro-Winkler Similarity in Figure 14 (f) is high because the trend is very similar, except for the opening time. In addition, because Figure 14 (e) shows a rise at the start and becomes flat in the subsequent part for both graphs, the similarity as per the Jaro-Winkler Similarity is good, whereas RMSE is bad due to the gap between the two graphs.

Lastly, Figure 15 depicts the stock data derived from the real and predicted data for the Hyundai Motor Company. First, in Figure 15 (a), two singularities can be seen simultaneously on both graphs, in the highest section and the middle section. For this reason, although the value of RMSE is fairly high, the similarity



(a) Oct. 27, 2014 (Monday)

(b) Oct. 28, 2014 (Tuesday)

(c) Oct. 29, 2014 (Wednesday)

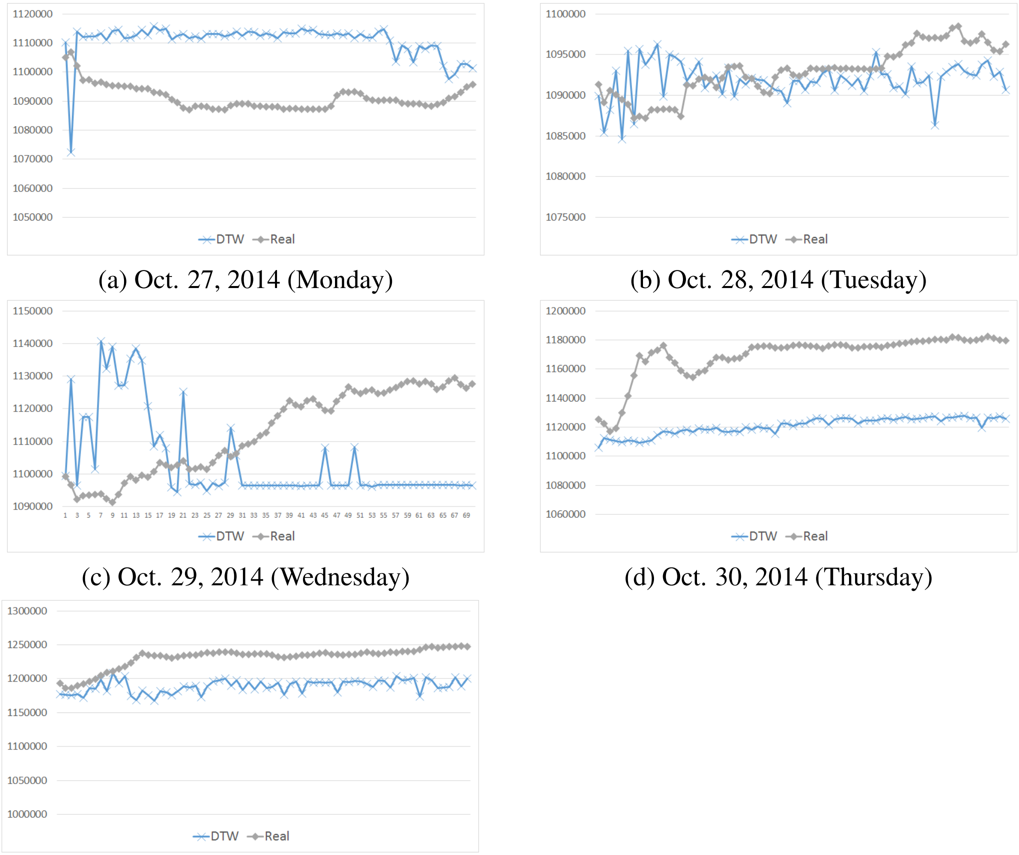
(d) Oct. 30, 2014 (Thursday)

(e) Oct. 31, 2014 (Friday)

LG Electronics Company	Jaro-Winkler Similarity	RMSE
Oct. 27	0.7071429	291.6
Oct. 28	0.6666667	1127.1
Oct. 29	0.4722222	2070.7
Oct. 30	0.6527778	4065.8
Oct. 31	0.4777778	568.7

(f) Comparison of Jaro-Winkler Similarity and RMSE

Figure 13: Comparison of real and predicted data for LG Electronics Company



(a) Oct. 27, 2014 (Monday)

(b) Oct. 28, 2014 (Tuesday)

(c) Oct. 29, 2014 (Wednesday)

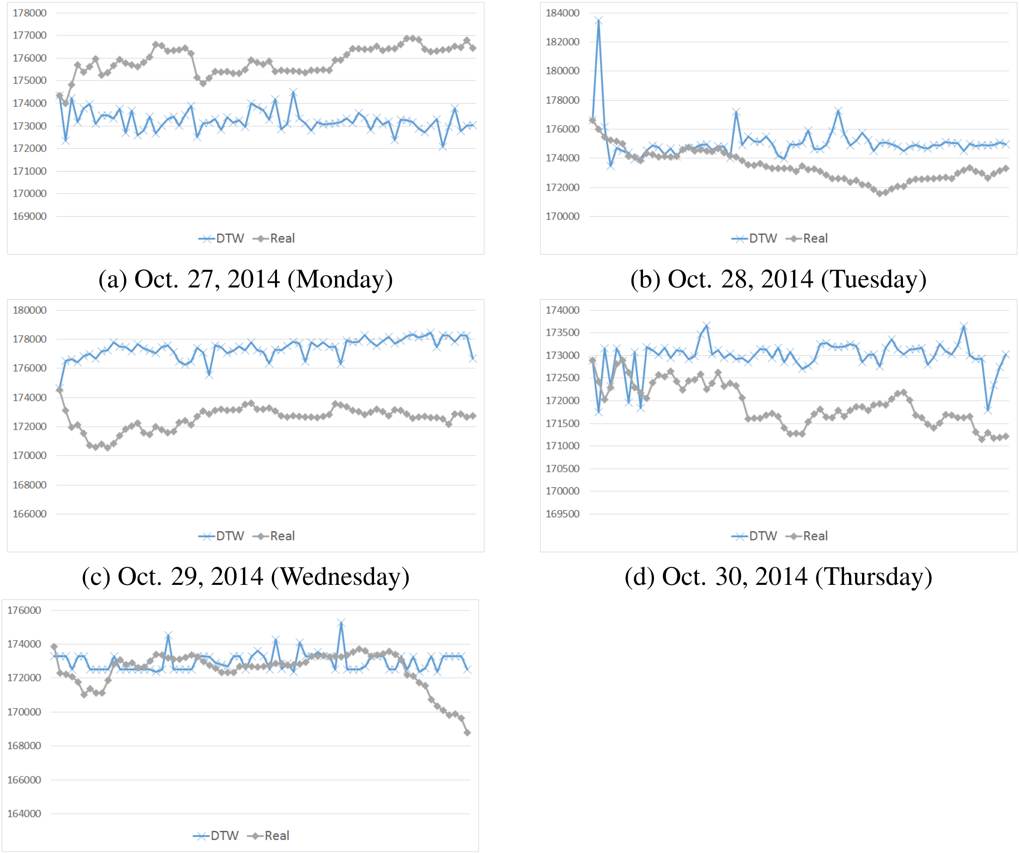
(d) Oct. 30, 2014 (Thursday)

(e) Oct. 31, 2014 (Friday)

Samsung Electronics Company	Jaro-Winkler Similarity	RMSE
Oct. 27	0.4777778	21273.5
Oct. 28	0.7071429	3858.3
Oct. 29	0.3888889	25066.9
Oct. 30	0.7071429	50016.3
Oct. 31	0.8818182	44720.1

(f) Comparison of Jaro-Winkler Similarity and RMSE

Figure 14: Comparison of real and predicted data for Samsung Electronics Company



(a) Oct. 27, 2014 (Monday)

(b) Oct. 28, 2014 (Tuesday)

(c) Oct. 29, 2014 (Wednesday)

(d) Oct. 30, 2014 (Thursday)

(e) Oct. 31, 2014 (Friday)

Hyundai Motor Company	Jaro-Winkler Similarity	RMSE
Oct. 27	0.6031746	2765.3
Oct. 28	0.6	2138.2
Oct. 29	0.7222222	4922.8
Oct. 30	0.4722222	1195.8
Oct. 31	0.5555556	1283.3

(f) Comparison of Jaro-Winkler Similarity and RMSE

Figure 15: Comparison of real and predicted data for Hyundai Motor Company

as per the Jaro-Winkler Similarity is quite good. Figure 15 (c) and (e) show totally different results; RMSE is not good and the similarity as per the Jaro-Winkler Similarity is good because the two graphs have a gradually increasing section even though the two graphs in Figure 15 (c) are quite far apart. On the other hand, RMSE is good and Jaro-Winkler Similarity is not good since the two graphs in Figure 15 (e) are closely attached with no singular points.

In conclusion, although our proposed method has low prediction accuracy, the best results were obtained for a short period of three months. To ensure achieving a higher prediction accuracy, it is necessary to secure stock data for at least on year. In addition, we should look into patterns of related stocks rather than single stocks because a stock does not change by itself, but by interacting with related stocks [6, 30].

6.3. Execution time for total processes

As mentioned above, our system consists of a total of five processes: data preprocessing, pattern matching, feature selection, prediction, and validation. Table 9 indicates average execution time for the five processes through thirty times. Among these processes, data extraction and data aggregation as a part of data preprocessing took the longest time because it requires extracting a lot of stock data from the files in HDFS. In addition, the more the amount of stock data, the longer the time required. The next most time-consuming part is to predict future stock price using an Artificial Neural Network. This is because there is a lot of stock data to learn, which requires a long time. On the other hand, the prediction time on Oct 27 and 28, as shown in Table 9 (c), is short because there is not much data to learn. The third time-consuming part is to match patterns based on dynamic time warping. This is reason why all patterns composed of five-minute intervals should be generated and matched for three months. Relatively, feature selection except the validation part took a little time because the features of top ten patterns should only be considered.

7. Related works

In this section, we introduce related works using various prediction methods such as artificial neural network, feature selection, and text mining for stock price prediction. First, an artificial neural network, which is the most widely used method since a few decades, was used by itself, and we gradually attempted to combine it with other techniques for a higher prediction accuracy. In [31], the authors have proposed a prediction system for buying and selling timing using

Table 9: Execution time of five processes

(a) Execution times of five processes for LG Electronic Company

LG Electronic Company	**Oct. 27**	**Oct. 28**	**Oct. 29**	**Oct. 30**	**Oct. 31**
Data extraction and Data aggregation	**499.51**	**499.431**	**502.758**	**501.638**	**502.878**
Pattern matching based on DTW	**11.406**	**11.494**	**11.756**	**12.239**	**12.132**
Feature selection	5.836	6.133	5.233	5.623	6.146
Prediction using ANN	7.884	3.617	6.65	**17.333**	**15.49**
Validation	0.824	1.29	0.987	0.438	1.38
Total number of stock data records	2,073,473				

(b) Execution times of five processes for Samsung Electronic Company

Samsung electronic company	**Oct. 27**	**Oct. 28**	**Oct. 29**	**Oct. 30**	**Oct. 31**
Data extraction and Data aggregation	**470.47**	**475.11**	**473.16**	**471.99**	**473.66**
Pattern matching based on DTW	**11.464**	**12.217**	**11.702**	**11.907**	**12.387**
Feature selection	5.809	6.284	5.472	5.335	6.079
Prediction using ANN	**16.046**	**10.021**	**12.697**	**59.301**	**35.62**
Validation	1.24	0.9	1.57	0.851	1.38
Total number of stock data records	1,986,010				

(c) Execution times of five processes for Hyundai Motor Company

Hyundai motor company	**Oct. 27**	**Oct. 28**	**Oct. 29**	**Oct. 30**	**Oct. 31**
Data extraction and Data aggregation	**366.13**	**362.79**	**365.2**	**367.69**	**365.562**
Pattern matching based on DTW	**11.344**	**11.645**	**11.81**	**11.941**	**12.162**
Feature selection	5.893	2.675	5.897	5.722	6.245
Prediction using ANN	7.123	1.442	**30.589**	**56.732**	**19.104**
Validation	1.8	1.908	1.766	1.228	1.13
Total number of stock data records	1,329,534				

economic indexes (foreign exchange rates) and technical indexes (vector curves) in the Tokyo Stock Exchange Prices Indexes. In another research, the authors have used an echo state network as a novel recurrent neural network to forecast the next closing price [37].

The following method is a feature selection method to select significant input attributes for supporting other methods and is recently often used. In [23], the authors have proposed combining support vector regression (SVR) with the self-organizing feature map (SOFM) technique and feature selection based on filtering for predicting the next days price index on Taiwan Index Futures (FITX). They selected the important features using the r-squared value as input data for SVR. In [34], the authors developed a prediction model based on support vector machine (SVM) with a hybrid feature selection method, which finds the original input features, for NASDAQ Index direction and unlike the above paper, f-score is used as a selection factor.

However, most of them have some limitations for short-term prediction. First, given all historical stock data as input data, because they have predicted the next closing price without removing the outliers, the error rate is high. Second, it was insufficient to consider various factors although the total trading price is determined by a variety of factors such as foreign purchase closing price and domestic selling trading amount. In other words, it is necessary to combine some significant factors.

8. Summary for our contributions from big data point of view

Since big data can be described using five Vs: **Volume**, **Variety**, **Velocity**, **Veracity** and **Value**, we explain our contributions by using five Vs in this paper [19, 25, 45].

Firstly, **Volume** means the amount of the data needed for an analysis. In this paper, we required a total 50 GB for twenty million datasets for historical stock data (Section 2.2) to forecast next days stock price. Although the size of this dataset might seem too small in comparison with other datasets such as twitter messages and Facebook messages, the amount of the stock data is never small when the number of users is compared. In addition, there are several studies conducted for predicting stock price and most researchers use small datasets based on the closing price, which is only useful for medium and long-term prediction. Besides, studies mainly focus on the prediction of stock price graph that fits perfectly well with the actual stock price graph. We find similar pattern shape as the current pattern; thus, a huge volume of historical data must be thoroughly analyzed.

Next, **Variety** refers to the numerous types of data that is structured, semi-structured, and unstructured. Unfortunately, if interpreted literally, we do not focus on this factor in-depth because stock data is structured data in the form of a file. However, if explaining this word in terms of data, it is fully satisfied since we use different types of determinants such as domestic selling high price, foreign buying opening price and domestic selling trading amount, as mentioned in Section 1 and 2.

Third, **Velocity** refers to the speed at which data is generated and processed. Because the stock data we store is historical data, the speed at which data is generated is excluded and the speed at which data is processed is the only area focused upon. Here, to quickly obtain the predicted stock price data for each item, we built a system based on big data open source programs in Section 5 and showed the results using execution times in Section 6.3. Considering this, it can be said that it is partially satisfied. Of course, since it is not a commercial-grade system, it does not seem to be as fast as a supercomputer, but it will gradually improve through enhancements in the future.

Veracity describes the fact that collected data should be good quality without biases, noise, and abnormality. In this paper, we believe that the data delivered to us will have high reliability because it is also used in business at KOSCOM. In order to deal with them that may be very rare, we are trying to eliminate them through data aggregation for generating predicted stock price data with high accuracy in Section 4.1.

Lastly, as recently added, **Value** indicates how valuable the content of the analysis is. In Section 4.6, we defined a new prediction accuracy measure by combining SAX and Jaro-Winkler distance for easily comparing the singular points of the predicted graph and the actual graph, and as a result, we achieved an accuracy of 60%. Although it is true that the accuracy is lower than in other domains, it can be considered to be somewhat reliable considering the prediction of the stock price pattern for one day.

9. Conclusion and future work

In this paper, we determined that stock prices sparsely show similar patterns and all the variables do not have a significant impact on the price. For short-term prediction, we proposed a novel method based on the combination of dynamic time warping, stepwise regression, and artificial neural network model to find similar historical datasets for each stock item and predict daily stock price using optimal significant variables through feature selection and comparison of leverage. Moreover, we dealt with the overall process using a big data processing framework composed of Hadoop, R, and RHive. Finally, we demonstrated the prediction accuracy for three stock items using SAX and Jaro-Winkler distance. In future work, we will improve the reliability of the predicted stock price by relation analysis of same field for a longer period and enhance the execution time by changing our system or file structure to use minimum search queries.

Acknowledgment

This work was supported by BK21PLUS, Creative Human Resource Development Program for IT Convergence.

References

[1] Adebiyi, A., Ayo, C., Adebiyi, M. O., Otokiti, S., 2012. Stock price prediction using neural network with hybridized market indicators. Journal of Emerging Trends in Computing and Information Sciences 3 (1), 1–9.

[2] Ané, T., Ureche-Rangau, L., Gambet, J.-B., Bouverot, J., 2008. Robust outlier detection for asia–pacific stock index returns. Journal of International Financial Markets, Institutions and Money 18 (4), 326–343.

[3] Banavas, G. N., Denham, S., Denham, M. J., et al., 2000. Fast nonlinear deterministic forecasting of segmented stock indices using pattern matching and embedding techniques. Computing in Economics and Finance 2000 64.

[4] Berndt, D. J., Clifford, J., 1994. Using dynamic time warping to find patterns in time series. In: KDD workshop. Vol. 10. Seattle, WA, pp. 359–370.

[5] Bollen, J., Mao, H., Zeng, X., 2011. Twitter mood predicts the stock market. Journal of Computational Science 2 (1), 1–8.

[6] Broadstock, D. C., Cao, H., Zhang, D., 2012. Oil shocks and their impact on energy related stocks in china. Energy Economics 34 (6), 1888–1895.

[7] Bulkowski, T. N., 2011. Encyclopedia of chart patterns. Vol. 225. John Wiley & Sons.

[8] Cao, L., Tay, F. E., 2001. Financial forecasting using support vector machines. Neural Computing & Applications 10 (2), 184–192.

[9] Cao, Q., Leggio, K. B., Schniederjans, M. J., 2005. A comparison between fama and french's model and artificial neural networks in predicting the chinese stock market. Computers & Operations Research 32 (10), 2499–2512.

[10] Chang, V., 2014. The business intelligence as a service in the cloud. Future Generation Computer Systems 37, 512–534.

[11] Chang, V., 2014. A proposed model to analyse risk and return for Cloud adoption. Lambert.

[12] Chang, V., Ramachandran, M., 2016. Towards achieving data security with the cloud computing adoption framework. IEEE Transactions on Services Computing 9 (1), 138–151.

[13] Chang, V., Walters, R. J., Wills, G. B., 2016. Organisational sustainability modellingan emerging service and analytics model for evaluating cloud computing adoption with two case studies. International Journal of Information Management 36 (1), 167–179.

[14] Charles, A., Darné, O., 2005. Outliers and garch models in financial data. Economics Letters 86 (3), 347–352.

[15] Chen, L., Ng, R., 2004. On the marriage of lp-norms and edit distance. In: Proceedings of the Thirtieth international conference on Very large data bases-Volume 30. VLDB Endowment, pp. 792–803.

[16] Chen, L., Özsu, M. T., Oria, V., 2005. Robust and fast similarity search for moving object trajectories. In: Proceedings of the 2005 ACM SIGMOD international conference on Management of data. ACM, pp. 491–502.

[17] Coelho, M. S., 2012. Patterns in financial markets: Dynamic time warping. Ph.D. thesis, NSBE-UNL.

[18] de Oliveira, F. A., Nobre, C. N., Zarate, L. E., 2013. Applying artificial neural networks to prediction of stock price and improvement of the directional prediction index–case study of petr4, petrobras, brazil. Expert Systems with Applications 40 (18), 7596–7606.

[19] Demchenko, Y., De Laat, C., Membrey, P., 2014. Defining architecture components of the big data ecosystem. In: Collaboration Technologies and Systems (CTS), 2014 International Conference on. IEEE, pp. 104–112.

[20] Fritsch, S., Guenther, F., Guenther, M. F., 2012. Package neuralnet. Training of Neural Network 1.

[21] Hsu, C.-M., 2013. A hybrid procedure with feature selection for resolving stock/futures price forecasting problems. Neural Computing and Applications 22 (3-4), 651–671.

[22] Huang, C.-F., Chang, B. R., Cheng, D.-W., Chang, C.-H., 2012. Feature selection and parameter optimization of a fuzzy-based stock selection model using genetic algorithms. International Journal of Fuzzy Systems 14 (1), 65–75.

[23] Huang, C.-L., Tsai, C.-Y., 2009. A hybrid sofm-svr with a filter-based feature selection for stock market forecasting. Expert Systems with Applications 36 (2), 1529–1539.

[24] Ince, H., Trafalis, T. B., 2007. Kernel principal component analysis and support vector machines for stock price prediction. IIE Transactions 39 (6), 629–637.

[25] Jamiy, F. E., Daif, A., Azouazi, M., Marzak, A., 2015. The potential and challenges of big data-recommendation systems next level application. arXiv preprint arXiv:1501.03424.

[26] Jeon, S., Hong, B., 2016. Monte carlo simulation-based traffic speed forecasting using historical big data. Future Generation Computer Systems 65, 182–195.

[27] Jeon, S., Hong, B., Lee, H., Kim, J., 2016. Stock price prediction based on stock big data and pattern graph analysis. In: Proceedings of the International Conference on Internet of Things and Big Data. pp. 223–231.

[28] Kim, K.-j., Han, I., 2000. Genetic algorithms approach to feature discretization in artificial neural networks for the prediction of stock price index. Expert systems with Applications 19 (2), 125–132.

[29] Kim, Y., Jeong, S. R., Ghani, I., 2014. Text opinion mining to analyze news for stock market prediction. Int. J. Advance. Soft Comput. Appl 6 (1).

[30] Kim, Y., Shin, J., 2000. Interactions among china-related stocks. Asia-Pacific Financial Markets 7 (1), 97–115.

[31] Kimoto, T., Asakawa, K., Yoda, M., Takeoka, M., 1990. Stock market prediction system with modular neural networks. In: Neural Networks, 1990., 1990 IJCNN International Joint Conference on. IEEE, pp. 1–6.

[32] Kohara, K., Ishikawa, T., Fukuhara, Y., Nakamura, Y., 1997. Stock price prediction using prior knowledge and neural networks. Intelligent systems in accounting, finance and management 6 (1), 11–22.

[33] Kuo, R. J., Chen, C., Hwang, Y., 2001. An intelligent stock trading decision support system through integration of genetic algorithm based fuzzy neural network and artificial neural network. Fuzzy sets and systems 118 (1), 21–45.

[34] Lee, M.-C., 2009. Using support vector machine with a hybrid feature selection method to the stock trend prediction. Expert Systems with Applications 36 (8), 10896–10904.

[35] Lin, B., Wehkamp, R., Kanniainen, J., 2015. Practitioner's guide on the use of cloud computing in finance. Available at SSRN 2697583.

[36] Lin, J., Keogh, E., Lonardi, S., Chiu, B., 2003. A symbolic representation of time series, with implications for streaming algorithms. In: Proceedings of the 8th ACM SIGMOD workshop on Research issues in data mining and knowledge discovery. ACM, pp. 2–11.

[37] Lin, X., Yang, Z., Song, Y., 2009. Short-term stock price prediction based on echo state networks. Expert systems with applications 36 (3), 7313–7317.

[38] Mittermayer, M.-A., 2004. Forecasting intraday stock price trends with text mining techniques. In: System Sciences, 2004. Proceedings of the 37th Annual Hawaii International Conference on. IEEE, pp. 10–pp.

[39] Nikfarjam, A., Emadzadeh, E., Muthaiyah, S., 2010. Text mining approaches for stock market prediction. In: Computer and Automation Engineering (ICCAE), 2010 The 2nd International Conference on. Vol. 4. IEEE, pp. 256–260.

[40] Pai, P.-F., Lin, C.-S., 2005. A hybrid arima and support vector machines model in stock price forecasting. Omega 33 (6), 497–505.

[41] Ramachandran, M., Chang, V., 2014. Financial software as a service–a paradigm for risk modelling and analytics. International Journal of Organizational and Collective Intelligence 4 (3), 65–89.

[42] Shen, S., Jiang, H., Zhang, T., 2012. Stock market forecasting using machine learning algorithms. Sruthi. V is currently pursuing BE computer Science and Engineering in SSN College of Engineering Chennai, India. She is doing research in the field of machine learning.

[43] Ticknor, J. L., 2013. A bayesian regularized artificial neural network for stock market forecasting. Expert Systems with Applications 40 (14), 5501–5506.

[44] Vlachos, M., Kollios, G., Gunopulos, D., 2002. Discovering similar multidimensional trajectories. In: Data Engineering, 2002. Proceedings. 18th International Conference on. IEEE, pp. 673–684.

[45] Wamba, S. F., Akter, S., Edwards, A., Chopin, G., Gnanzou, D., 2015. How big datacan make big impact: Findings from a systematic review and a longitudinal case study. International Journal of Production Economics 165, 234–246.

[46] Wang, J.-H., Leu, J.-Y., 1996. Stock market trend prediction using arima-based neural networks. In: Neural Networks, 1996., IEEE International Conference on. Vol. 4. IEEE, pp. 2160–2165.

[47] Winkler, W. E., 1990. String comparator metrics and enhanced decision rules in the fellegi-sunter model of record linkage., 778–783.

[48] Zhang, Y., Glass, J. R., 2011. A piecewise aggregate approximation lower-bound estimate for posteriorgram-based dynamic time warping. In: Interspeech. pp. 1909–1912.

[49] Zhang, Z., Jiang, J., Liu, X., Lau, R., Wang, H., Zhang, R., 2010. A real time hybrid pattern matching scheme for stock time series. In: Proceedings of the Twenty-First Australasian Conference on Database Technologies-Volume 104. Australian Computer Society, Inc., pp. 161–170.

*Biographies (Text)

Seungwoo Jeon received the B.S., M.S. and Ph.D. degrees in computer engineering from Pusan National University (PNU), Busan, Korea, in 2006, 2011 and 2016, respectively. He is going to develop traffic predictor software with Big Data Processing Platform Research Center(BDRC). Now, his research focuses on time-series prediction, statistical model and big data processing software in various fields.

Bonghee Hong received the M.S. and Ph.D. degrees in computer engineering from Seoul National University, Seoul, Korea, in 1984 and 1988. In 1987, he joined the faculty of Computer Engineering of the Pusan National University (PNU). He
is working as a professor of database in the Department of Computer Engineering at the PNU. He is a director of Big Data Processing Platform Research Center(BDRC). He is also a steering committee member of DASFAA. His research interests include theory of database systems, moving object databases, spatial databases and big data processing for traffic prediction.

Victor Chang is an Associate Professor in Information Management and Information Systems and also a Director of PhD Programme at International Business School Suzhou (IBSS), Xi'an Jiaotong Liverpool, China. Dr. Victor Chang was a Senior Lecturer in Computing at Leeds Beckett University, UK and a Visiting Researcher at the University of Southampton, UK. He has been a technical lead in web applications, web services, database, grid, cloud, storage/backup, bioinformatics, financial computing which subsequently have become his research interests. Victor has also successfully delivered many IT projects in Taiwan (place of birth), Singapore, Australia, and the UK since 1998. Victor is experienced in a number of different IT subjects and has 27 certifications with 97% on average. He completed PGCert (Higher Education, University Greenwich, 2012) and PhD (C.S, University of Southampton, 2013) within four years while working full-time, whereby the distance between his work and research is about hundreds of miles away. He has over 100 published peer-reviewed papers, including several high-quality journals up-to-date.

*Biographies (Photograph)

Click here to download high resolution image

*Biographies (Photograph)

Click here to download high resolution image

***Biographies (Photograph)**

Click here to download high resolution image