J. Korean Soc. Ind. Appl. Math. Vol.26, No.1, 23–48, 2022 https://doi.org/10.12941/jksiam.2022.26.023

PREDICTING KOREAN FRUIT PRICES USING LSTM ALGORITHM

TAE-SU PARK[1], JONGHAE KEUM[2], HOISUB KIM[2], YOUNG ROCK KIM[2,3], AND YOUNGHO MIN[3†]

[1] DEPARTMENT OF MATHEMATICS, HANKUK UNIVERSITY OF FOREIGN STUDIES, YONGIN, 17035, REPUBLIC OF KOREA
Email address: tessinu51@gmail.com

[2] SCHOOL OF MATHEMATICS, KOREA INSTITUTE FOR ADVANCED STUDY, 85 HOEGIRO DONGDAEMUN-GU, SEOUL, 02455, REPUBLIC OF KOREA
Email address: jhkeum@kias.re.kr, hskimmdial@naver.com

[3] DEPARTMENT OF MATHEMATICS EDUCATION, HANKUK UNIVERSITY OF FOREIGN STUDIES, SEOUL, 02450, REPUBLIC OF KOREA
Email address: rocky777@hufs.ac.kr, †pumpzxc@gmail.com

ABSTRACT. In this paper, we provide predictive models for the market price of fruits, and analyze the performance of each fruit price predictive model. The data used to create the predictive models are fruit price data, weather data, and Korea composite stock price index (KOSPI) data. We collect these data through Open-API for 10 years period from year 2011 to year 2020. Six types of fruit price predictive models are constructed using the LSTM algorithm, a special form of deep learning RNN algorithm, and the performance is measured using the root mean square error. For each model, the data from year 2011 to year 2018 are trained to predict the fruit price in year 2019, and the data from year 2011 to year 2019 are trained to predict the fruit price in year 2020. By comparing the fruit price predictive models of year 2019 and those models of year 2020, the model with excellent efficiency is identified and the best model to provide the service is selected. The model we made will be available in other countries and regions as well.

1. INTRODUCTION

Fruits are the oldest food of mankind and many countries around the world are conducting research on the supply and demand of fruits to analyze the market price of fruits [1, 2, 3, 4]. One of the factors that greatly influences fruit supply and price is the weather, which is measured by indicators such as temperature, precipitation and wind speed. If the appropriate temperature and precipitation are not adjusted for each fruit, problems such as lowering of sugar content or damage to the fruit occur, which adversely affects the wholesale price of fruit [5]. The uncertainty of the weather makes fruit suppliers feel anxious. Therefore, various countermeasures such as high tunnels, revenue insurance, and weather insurance have been

Received December 30 2021; Revised March 10 2022; Accepted in revised form March 24 2022; Published online March 25 2022.
1991 *Mathematics Subject Classification.* 65Yxx, 68Txx.
Key words and phrases. LSTM, Predictive model, Time series data, Fruit prices, Weather data, KOSPI.
†Corresponding author.

proposed to solve this problem [6]. Several countries have implemented insurance policies for crops, including fruits, to stabilize fruit prices and protect fruit suppliers. An example of a nationally implemented fruit supply and demand policy is the US Supplemental Nutrition Assistance Program (SNAP). As part of the consumption policy to improve the people's healthy eating habits and increase fruit consumption, it is supporting subsidies for fruit consumption by the low-income class. It was based on a study of the relationship between government subsidies and fruit consumption reported to the US Department of Agriculture (USDA) [7]. Also, for fruit price stabilization, not only the supply and demand of domestically produced fruits but also the prices of imported fruits are considered. Fruits imported indiscriminately at very low prices adversely affect the stabilization of fruit prices in the region, and excessively high tariffs also prevents the provision of fresh fruits at good prices to consumers. Rickard, BJ, & Lei, L. simulated the reduction of global tariffs and elimination of sanitary and phytosanitary (SPS) in the apple and orange markets to analyze the impact of tariff and non-tariff barriers on consumers and suppliers in the international market and suggested strategies for stabilizing fruit prices [8]. In this way, the price of fruit is determined by many variables related to supply and demand. The stable fruit price guarantees economic benefits for suppliers and provides healthy food to consumers.

Stabilization of fruit prices is a very important issue from a regional and international perspective, and various studies are being conducted for this purpose. However, there are not many studies that apply deep learning to fruit or agricultural and fishery products data. In this paper, we pay attention to the prediction of fruit prices using deep learning to help stabilize fruit prices. We can suggest several policies through accurate fruit price prediction. If fruit prices are predicted to be higher than those in previous years, the government can stabilize fruit prices by increasing imports of overseas substitute fruits and implementing policies that guarantee the profits of suppliers. Conversely, if fruit prices are expected to be low, policies to support consumers' fruit consumption can be planned [9].

As an example of the introduction of deep learning on fruit prices in Korea, there is a study on a fruit price predictive model using artificial intelligence by Im, J. M., Kim, W. Y., Byoun, W. J., & Shin, S. J. [10]. They predicted fruit price using time series data based on LSTM (Long Short Term Memory) algorithm among RNN (Recurrent Neural Network) algorithms. In their paper, the attempt to analyze fruit prices through deep learning was good, but the data used for training the deep learning algorithm were only short-term weather data, and the prediction target value was only the price of apples in a specific period. There is the other study by Shin, S., Lee, M., & Song, S. K. who studied agricultural product price prediction using LSTM network [11]. In their paper, 108 features for training was used to predict the price of agricultural products and root mean square error (RMSE) was used as a performance measurement tool. They obtained the root mean square error ranging from 0.065 to 0.121 for city/agricultural products.

The goal of our paper is to present predictive models for the market price of fruits using the LSTM algorithm, and to provide highly efficient predictive models with a small number of features by analyzing the performance of each fruit price predictive model.

This paper consists of 8 sections and proceeds in the following order. In Section 2, we explain RNN and LSTM, which are machine learning techniques used in this study. Section 3 contains a description of the workflow and an overview of the progress of this study. In Section 4, we collect the data through the Open-API, and describe the collected data. In Section 5, we explain the process of preprocessing the data obtained in Section 3 and the merging of data used for training. We construct predictive models for fruit prices in Section 6. In Section 7, we summarizes the learning process, compares and analyzes the performance of the models. Finally, we discuss the conclusions of this study in Section 8.

2. METHODOLOGY

Unlike general programming, machine learning refers to programming that allows a program to learn and develop by itself by implanting a neural network that mimics the structure of a human neuron into an algorithm. When the number of hidden layers of the neural network used in the machine learning increases, we call this deep learning. The core idea of deep learning is to find the weights and biases of the neural network that minimizes the loss function through backpropagation method [12]. In deep learning, the number of weights and biases increases exponentially as the neural network grows deeper.

The fruit prices, weather, and Korea composite stock price index data that we are going to cover in this paper are time series data, and which generally have very long sequence. Therefore we need to use long-length deep neural networks for handling time series data. However, it takes a lot of memory and time to store and update all the weights and biases to train these deep neural networks. This problem was solved by RNN using weights and biases again. In 1993, Schmidhuber, J. stated that RNN is suitable for handling data requiring more than 1000 subsequent layers [13]. For this reason, the RNN is used when dealing with time series data in various research fields, and our models are also created using LSTM that is a kind of RNN algorithm. In this section, we introduce the basic RNN algorithm and the LSTM algorithm to explain our models.

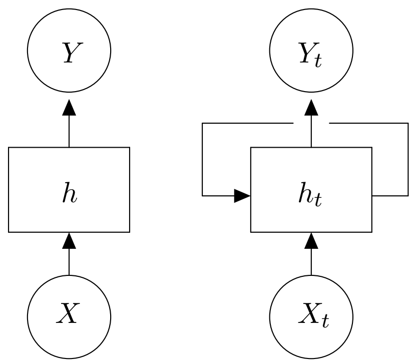

FIGURE 1. Basic feedforward neural networks and recurrent neural networks

2.1. **Recurrent Neural Network (RNN).** The RNN algorithm is one of artificial neural network and has a recurrent characteristic of applying the weights obtained through learning. Fig. 1 is a diagram comparing the basic feed-forward neural network and the recurrent neural network, in short, RNN. In the figure, the arrow direction means the network flow, X indicates the input layer, h is the hidden layer, and Y is the output layer. In the basic feed-forward neural network, input data are transmitted from the input layer to the hidden layer and then from the hidden layer to the output layer. However, in the RNN, the hidden layer has a structure that receives information from the hidden layer of the previous time step as well as information from the input layer. This flow of information is called a loop or recurrent edge, and the name RNN is derived from this.

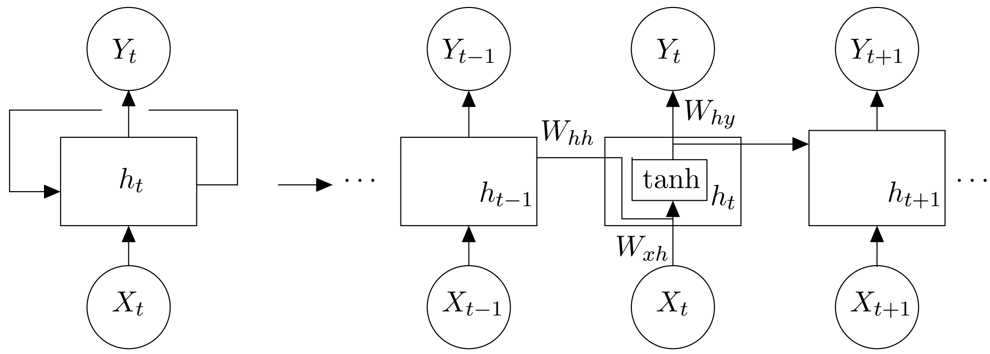

FIGURE 2. Recurrent neural network

Figure 2 shows the unfolded recurrent neural network structure. The detailed method of calculating the outputs of the hidden layer and the output layer with the input data in the t-step is as follows. Let W_{xh}, W_{hh}, W_{hy} be the weight matrix between the input X_t and the hidden layer, the weight matrix for the recursive edge, and the weight matrix between the hidden layer and the output layer, respectively. The new input data X_t and the past learning results h_{t-1} are multiplied by the corresponding weights and added, and then the bias b_h at the t-step is added. It is expressed as a linear combination

$$Z_t = W_{xh}X_t + W_{hh}h_{t-1} + b_h. \tag{2.1}$$

By putting this input value Z_t into the hyperbolic tangent activation function, the output h_t at t-step can be calculated. If the weight matrix is defined as $W_h = [W_{xh} : W_{hh}]$, then this process can be simply expressed as follows.

$$h_t = \tanh\left(W_h \begin{bmatrix} X_t \\ h_{t-1} \end{bmatrix} + b_h\right), \tag{2.2}$$
$$y_t = W_{hy} h_t + b_y,$$

where b_y is the bias for output layer. Now, the output h_t is transferred to the next hidden layer, multiplied by the weight again, added to the input value $W_{xh}X_{t+1}$ of the next layer, and then the bias b_h is added, repeating the structure. Here, the weights W_{xh}, W_{hh}, W_{hy} are used again. Because of this structural characteristic, the order of the data has a great influence on the learning result.

Although RNN is good at handling sequential data, there are some problems when we deal with very long sequences. In backpropagation phase, we calculate the gradient of the loss function at time step t as follows:

$$\frac{\partial L_t}{\partial W_{hh}} = \frac{\partial L_t}{\partial y_t} \times \frac{\partial y_t}{\partial h_t} \times \left(\sum_{k=1}^{t} \frac{\partial h_t}{\partial h_k} \times \frac{\partial h_k}{\partial W_{hh}}\right),$$

where $\dfrac{\partial h_t}{\partial h_k} = \displaystyle\prod_{i=k+1}^{t} \frac{\partial h_i}{\partial h_{i-1}}$ which is $(t-k)$ multiplicative terms. By the Eq. (2.2), we can obtain

$$\frac{\partial h_i}{\partial h_{i-1}} = \frac{\partial \tanh(Z_i)}{\partial Z_i} \frac{\partial Z_i}{\partial h_{i-1}}.$$

In the Eq. (2.1), Z_i consists of the input term (X_i), the previous hidden layer output (h_{i-1}) and bias term (b_h). Therefore, only the weight matrix W_{hh} remains in $\dfrac{\partial Z_i}{\partial h_{i-1}}$. Since the derivative of the $\tanh$ activation function is positive and smaller than 1, therefore if t is very large and $\|W_{hh}\| < 1$, then vanishing gradient problem is occurred where $\|\cdot\|$ is any matrix norm [14]. Also if t is very large and norms of W_{hh} are large enough to overpower the smaller derivative of $\tanh$, the exploding gradient problem is ocurred [15]. In next subsection, we introduce LSTM as the solution to these problems.

2.2. **LSTM (Long Short Term Memory).** Hochreiter and Schmidhuber proposed LSTM, a modification of the RNN, to solve the vanishing gradient problem caused by prolonged learning [16]. LSTM has an internal structure called a memory cell, and the memory cell contains a recurrent edge that maintains an appropriate weight $\|W\| = 1$. The output of this recurrent edge is called a cell state.

The flow of information in the memory cell is controlled through several gates. In general, there are three types of gates in an LSTM cell. These are the forget gate, the input gate, and the output gate. The forget gate (f_t) adjusts the output of the hidden unit at the $t-1$ time step (h_{t-1}) and the input at the t time step (X_t) to determine which information to pass through and which information to suppress [17]. f_t is calculated as follows:

$$f_t = \sigma(W_{xf}X_t + W_{hf}h_{t-1} + b_f),$$

where σ is the sigmoid function. The input gate (i_t) and the input node (g_t) update the cell state,

$$\begin{aligned} i_t =& \sigma(W_{xi}X_t + W_{hi}h_{t-1} + b_i) \\ g_t =& \sigma(W_{xg}X_t + W_{hg}h_{t-1} + b_g). \end{aligned}$$

The cell state at time step t (C_t) is updated by element-wise adding information received from the input gate and input node.

$$C_t = (C_{t-1} \otimes f_t) \oplus (i_t \otimes g_t)$$

where the symbol $\otimes$ means element-wise multiplication and the symbol $\oplus$ means element-wise addition, respectively. This network is designed to obtain the cell state of the next time step without directly multiplying the cell state of the previous time step with any weight. This structure of LSTM solves the vanishing gradient problem by avoiding the problem of multiplying weights by superposition. Using this cell state (C_t), the output of the hidden unit at the time step t is calculated as follows:

$$h_t = o_t \otimes \tanh(C_t),$$

where o_t is the output gate defined by $o_t = \sigma(W_{xo}X_t + W_{ho}h_{t-1} + b_o)$. Figure 3 shows the detailed structure of the LSTM.

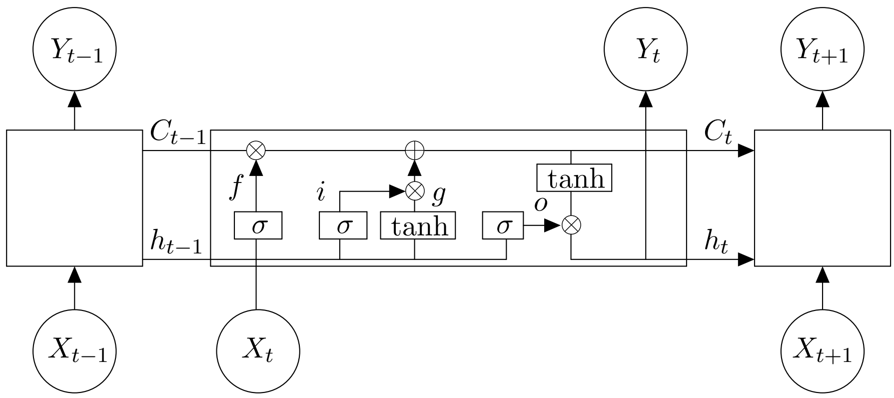

FIGURE 3. Long short term memory

3. FRUIT PRICE PREDICTION WORKFLOW

We will explain the workflow that summarizes the process of this study in Fig. 4. A brief description of each part is as follows.

First, data are collected through Open-API (Open Application Programming Interface). The collected data are fruit price data, weather data, and comprehensive stock index data. The

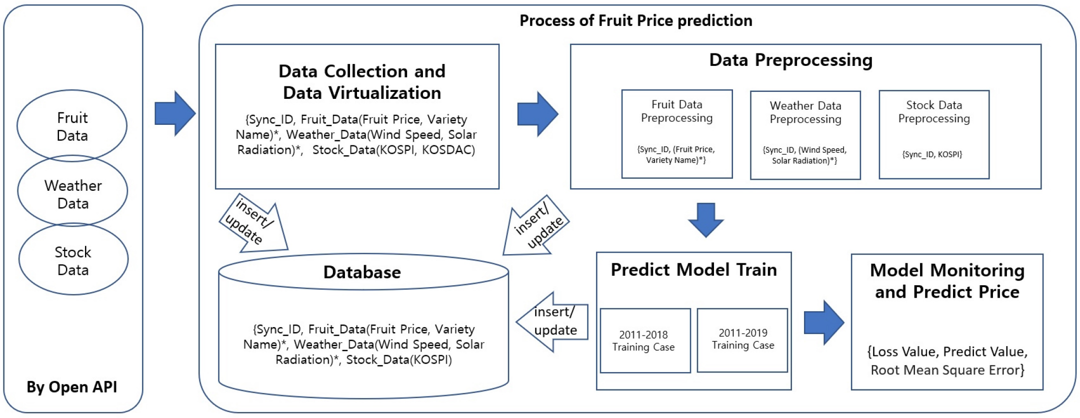

FIGURE 4. Fruit price prediction workflow

collected fruit price data contain prices of a variety of fruits including our target fruits (apples, pears, persimmons, bananas, and oranges) by 5 different regions in Korea. This raw data include quality and daily price of each fruit. Weather data have 15 features which are average temperature, minimum temperature, maximum temperature, daily precipitation, maximum instantaneous wind speed, maximum wind speed, average wind speed, average dew point temperature, minimum relative humidity, average relative humidity, average vapor pressure, hot water time, total solar time, total solar radiation and average surface temperature. KOSPI data and KOSDAQ data are collected as comprehensive stock index data. Data collection will be described in detail in Section 4.

Next, we will preprocess the collected data. Fruit price data consist of data based on the trading day. Preprocessing of weather data is performed through multiple linear regression analysis. Composite stock index data are extracted as KOSPI data on the same day as the fruit trading day, and preprocessing is performed. The preprocessing content will be described in detail in Section 5.

Based on the preprocessed data, six types of data sets are constructed according to the independent variables used for each model. The details of the independent variables used in each model will be described in detail in Section 6.

For each model, we will train the data from year 2011 to year 2018 to predict the fruit price in year 2019 and compare it with the actual fruit price. Similarly, we will train on data from year 2011 to year 2019 to predict fruit prices in year 2020. We will use the root mean square error to measure performance and select the best model. This will be described in detail in Section 7.

4. DATA COLLECTION AND VIRTUALIZATION

In this study, fruit price data, weather data related to fruit prices, and comprehensive stock index data are collected. The collected data are daily data from January 2011 to December 2020 in Korea. All data are collected using Open-API. Fruit price data are collected from Korea Agro-Fisheries & Food Trade Corporation, in short KAMIS (`www.kamis.or.kr`) Open-API and weather data in public data portal (`www.data.go.kr`) are collected. Also, we could collect data through the Open-API and pandas-datareader library for the comprehensive stock index. High price, low price, opening price, closing price, trading volume in KOSPI index, and revised stock information are standardized and collected by date through the Python library.

4.1. **Fruit price data.**

Fruit price data are collected via Open-API and are provided by KAMIS. The 'wholesale/retail price information by category' API is selected among the open APIs provided. We request the data from KAMIS using the request URL and request parameter, and we receive data as the response field.

TABLE 1. Request parameters

Request Parameter	Value	Explanation
p_cert_key	string	certification key
p_cert_id	string	requester id
p_returntype	string	Return Type (json:Json data form, xml: XML data form)
p_product_cls_code	string	division (01:retail, 02:wholesale, default:02)
p_item_category_code	string	class code (100:food crops, 200:vegetables, 300:special crop, 400:fruits, 500:livestock, 600:seafood, default:100)
p_country_code	string	retail price selectable area (1101: Seoul, 2100: Busan, 2200: Daegu, 2300: Incheon, 2401: Gwangju, 2501: Daejeon, 2601: Ulsan, 3111: Suwon, 3211: Chuncheon, 3311: Cheongju, 3511: Jeonju, 3711: Pohang, 3911 : Jeju, 3113: Uijeongbu, 3613: Suncheon, 3714: Andong, 3814: Changwon, 3145: Yongin) Wholesale price selection area (1101: Seoul, 2100: Busan, 2200: Daegu, 2401: Gwangju, 2501: Daejeon)
p_regday	string	Date: yyyy-mm-dd (default: latest survey date)
p_convert_kg_yn	string	Whether to convert in kg unit (Y: 1 kg unit display, N: information survey unit display, ex: rice 20 kg)

The request parameter is a variable requested together with the request URL when we request data from the server. Table 1 shows the data explaining the configuration of the request variable. Table 2 describes an extract of some of the output results.

TABLE 2. Response elements

field	Value	Explanation
condition	string	request message
item_name	string	Item name
itemcode	string	Item code
dpr1	string	View Date Price
⋮	⋮	⋮

4.2. **Weather data and composite stock index data.** Weather data and comprehensive stock index data are collected using Open-API provided by the public data portal (`www.data.go.kr`). In the case of weather data, the 'terrestrial (synoptic, ASOS) daily data inquiry service' is selected among the Open-APIs provided through 'Daily Weather Data Inquiry', and this is a service that inquires the daily weather data observed with the synoptic meteorological observation equipment. We requested corresponding data through a request URL and a request parameter to receive the corresponding data, and we received data as an output result field (response field). Table 3 has the contents extracted from some of the request variables. The following is the request url. `http://apis.data.go.kr/1360000/AsosDalyInfoService/getWthrDataList`

TABLE 3. Weather request parameters

Item name	Item size	Category	sample data	Item Description
serviceKey	100	1	certification key (URL Encode)	Public data portal certification key
numOfRows	4	0	10	number of results per page Default: 10
pageNo	4	0	1	page number Default: 1
⋮	⋮	⋮	⋮	⋮

Table 4 below shows some of the output values.

GDP (Gross Domestic Product) is a representative indicator that shows the overall situation of a country's economy at a glance, but in the case of GDP, it is difficult to apply to this study because the indicator GDP is calculated on a quarterly basis. Therefore, instead of GDP, the stock index is an indicator that can determine the trend of the stock market. Indices of the Korean stock market include the KOSPI (Korean version of the US Dow Jones) and the KOSDAQ (Korean version of the US NASDAQ). To collect these data, the pandas_datareader library is

TABLE 4. Weather response elements

Item name	Item size	Item classification	sample data	Item Description
numOfRows	4	1	1	display per page number of data
pageNo	4	1	1	number of pages Default: 10
totalCount	10	1	1	total number of data Default: 1
⋮	⋮	⋮	⋮	⋮

used, and the closing price information by date of the stock market trend is preprocessed and used in this study. The collected data are shown in Table 5, and the information about the columns constituting the collected data are in the order of date, high price, low price, opening price, closing price, trading volume, and adjusted close.

TABLE 5. KOSPI response elements

Date	High	Low	Open	Close	Volume	Adj Close
2010-01-04	1696.14	1681.71	1681.71	1696.14	296500	1696.14
2010-01-05	1702.39	1686.45	1701.62	1690.62	408900	1690.62
2010-01-06	1706.89	1696.1	1697.88	1705.32	426000	1705.32
2010-01-07	1707.9	1683.45	1702.92	1683.45	462400	1683.45
2010-01-08	1695.26	1668.84	1694.06	1695.26	380000	1695.26
⋮	⋮	⋮	⋮	⋮	⋮	⋮
2020-12-23	2769.08	2716.28	2737.74	2759.82	1121300	2759.82
2020-12-24	2812.16	2762.6	2762.6	2806.86	1030900	2806.86
2020-12-28	2834.59	2799.56	2820.95	2808.6	1006200	2808.6
2020-12-29	2823.44	2792.06	2810.55	2820.51	1046800	2820.51
2020-12-30	2878.21	2809.35	2820.36	2873.47	1074000	2873.47

4.3. **Data collection and virtualization results.** The collected data is in XML format and preprocessing of the data is required to utilize it. Since the Open-API output result is in XML format, data virtualization is realized by using the Pandas Python Library to standardize it in a table format. Afterwards, we will remove empty and unnecessary data in preprocessing phase.

5. DATA PREPROCESSING

5.1. **Fruit price data preprocessing.** Fruit price data of KAMIS exist by region (Seoul, Busan, Daegu, Daejeon, Gwangju). The data to be targeted here would be fruit price data in Seoul, and fruit price data in the rest of regions are used as an independent variable (feature) to train. From the collected regional fruit sales data, high quality fruit products are selected and the learning data for this fruit is extracted. For example, after selecting an apple (Fuji) in Seoul, the transaction date and transaction price are extracted at Pandas data table. After that, apple (Tsugaru), apple (Hongro), pear (Wonhang), pear (Shingo), banana, and orange data are extracted by same way for other fruits in Seoul. After completing the data extraction for the Seoul area, the price and transaction date are extracted in the same way as the method of extracting the price in the Seoul area while changing the region.

Because the sales period differs for each fruit, there is no full time-series fruit data on an annual basis. For time series data learning, we preprocess the data by adding information on the week and day of the week to the fruit data in this study. First, we construct data using the transaction price of the day before the holiday on days when no sales were made, such as holidays, and we put 0 won for a period with no transactions for more than 5 days. A year is counted as 52 weeks, and values up to 52 are assigned to the week, and values from 0 to 4 are assigned from Monday to Friday when there is a transaction to preprocess the time series data. Table 6 below shows some of the preprocessed apple (Fuji) data.

TABLE 6. Part of apple (Fuji) data

date	day of week	week of year	Pusan	Daejeon	Daegu	Gwangju	Seoul
2011-01-03	0	1	6000	6000	6000	6000	6000
2011-01-04	1	1	6000	6333	6000	6000	6000
2011-01-05	2	1	6000	6333	6000	6000	6000
2011-01-06	3	1	6000	6333	6000	6000	6000
2011-01-07	4	1	6000	6333	6000	6000	6000
2011-01-10	0	2	6000	6333	6000	6000	6000
2011-01-11	1	2	6000	6333	6000	6000	6000
2011-01-12	2	2	6000	6333	6000	6000	6000
2011-01-13	3	2	6000	6333	6000	6000	6000
2011-01-14	4	2	6000	6333	6000	6000	6000
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮

5.2. **Weather data preprocessing.** We could extract 15 columns to be used for deep learning from the weather data table standardized through Open-API. The data consist of weather data from January 2011 to December 2020. The 15 extracted columns are average temperature, minimum temperature, maximum temperature, sum of rainfall, maximum instantaneous wind speed, maximum wind speed, average wind speed, average temperature of dew point, minimum relative humidity, average relative humidity, average pressure of vapor, sunshine duration, sum of sunshine hour, sum of solar radiation and average temperature on surface. Information and meaning for each column is presented in table 7.

5.3. **Multiple linear regression analysis.** The performance of the model is not guaranteed by using all 15 columns from weather data. Some features may not be very helpful for training.

TABLE 7. Extracted weather data

Item name	Item Description
average temperature	daily average temperature
minimum temperature	daily minimum temperature
maximum temperature	daily maximum temperature
sum of rainfall	daily precipitation
maximum instantaneous wind speed	daily maximum instantaneous wind speed
maximum wind speed	daily maximum wind speed
average wind speed	daily average wind speed
average temperature of dew point	average temperature when water vapor condenses
minimum relative humidity	daily minimum relative humidity (relative humidity : the ratio of the mass of water vapor in the atmosphere divided by the amount of saturated water vapor at the current temperature)
average relative humidity	daily average relative humidity
average pressure of vapor	the pressure due to vaporization of a solid or liquid
sunshine duration	the length of time between when the sun's center rises to the horizon and sets again on the horizon
sum of sunshine hour	the amount of time direct sunlight hits the Earth's surface
sum of solar radiation	the radiant energy of the sun reaching the Earth's surface
average temperature on surface	the temperature of the air near the surface of the earth

With this possibility in mind, we extract features that are highly relevant to the target values, in other words, fruit prices in Seoul. Results for these extractions will be analyzed in section 7.

We perform feature extraction through multiple linear regression analysis. Multiple linear regression analysis is an analysis method that verifies the effect of two or more continuous independent variables on continuous dependent variables. The concept and analysis method are the same as simple regression analysis, only the number of independent variables is different. For multiple linear regression analysis, preprocessed data of apples, pears, persimmons, bananas, and oranges are prepared. Thereafter, multiple linear regression analysis is performed between each fruit prices and the weather data. We have 5 weather features that are commonly significant, that is, Significance F is less than 0.05.

TABLE 8. Multiple linear regression analysis of weather data for apple (Fuji)

	Degree of Freedom	Sum of Squares	Mean of Squares	F	Significance F
Regression	15	527591627	35172775.13	33.66018	3.88E-87
Residual	2081	2174514214	1044937.152		
Total	2096	2702105841			

The apple data used to multiple linear regression analysis are daily data of 2,096 cases for a total of 10 years from year 2011 to year 2020. Table 8 is the first part of the contents of multiple linear regression analysis based on 95% reliability of apple data. Significance F is an indicator that verifies whether it is statistically valid. As the value of F is large and the value of Significance F is less than 0.05, a more significant value can be obtained. In the case of the above apple, Significance F is less than 0.05, so it can be said to be significant.

TABLE 9. Multiple linear regression analysis of apple data

Parmeter name	Item name	Coefficients	p-value
Y-intercept	Y-intercept	2474.547	0.009537
avgTa	average temperature	7.736941	0.88927
minTa	minimum temperature	-19.4318	0.439966
maxTa	maximum temperature	-19.0545	0.481196
sumRn	sum of rainfall	4.43451	0.017636
maxInsWs	maximum instantaneous wind speed	-215.362	1.52E-23
maxWs	maximum wind speed	359.9087	4.4E-16
avgWs	average wind speed	215.7026	5.55E-05
avgTd	average temperature of dew point	16.24501	0.68126
minRhm	minimum relative humidity	0.858588	0.827902
avgRhm	average relative humidity	-9.55187	0.365955
avgPv	average pressure of vapor	57.14733	5.74E-08
ssDur	sunshine duration	197.0722	5.34E-09
sumSsHr	sum of sunshine hour	-86.9069	2.46E-09
sumGsr	sum of solar radiation	59.33607	7.58E-09
avgTs	average temperature on surface	-42.0829	0.001735

Table 9 is the second part of the multiple linear regression analysis of apple data. The Y values are as follows.

$$\begin{aligned}\text{Y value} = 2474.55 &+ 7.74\text{avg Ta} - 19.43\text{min Ta} - 19.05\text{max Ta} \\ &+ 4.43\text{sum Rn} - 215.36\text{max Ins Ws} + 359.91\text{max Ws} + 215.70\text{avg Ws} \\ &+ 16.25\text{avg Td} + 0.86\text{min Rhm} - 9.55\text{avg Rhm} + 57.15\text{avg Pv} \\ &+ 197.07\text{ss Dur} - 86.91\text{sum SsHr} + 59.34\text{sum Gsr} - 42.08\text{avg Ts}.\end{aligned}$$

The p-value is an index that verifies whether it is statistically valid, and a value of 0.05 or less based on a 95% confidence level can be considered as a significant value. Through multiple linear regression analysis of the remaining types of apples, pears, persimmons, bananas, and oranges, five factors including maximum instantaneous wind speed, average vapor pressure, heating time, total solar time, and total solar radiation, which are weather features commonly related to all fruit prices are selected.

5.4. **Comprehensive stock index data preprocessing.** We collect the comprehensive stock index data from year 2011 to year 2020. The data consist of year, KOSPI index, and KOSDAQ index columns, and it is composed of daily data. Since KOSPI data are judged to be suitable for domestic market analysis than KOSDAQ, so we use KOSPI data as the Korean stock market index. In order to analyze the fruit price, the KOSPI data are preprocessed to be the same day as the day on which the fruit was traded.

5.5. **Data merge preprocessing.** We combine all the preprocessed data to create a database. Based on the fruit data, it goes through the process of merging the weather data and the KOSPI index data on the day that the fruit was traded. Then we can sort all the data by date and each column means a feature. This database contains 45 columns of price in 5 cities of 9 fruits, 2 columns of numeric data for dates and weeks, 15 columns of weather data, and one column of KOSPI data. We extracted five features from 15 weather data using multiple linear regression analysis in Section 5.3. We will use this database as training data and test data for learning various models.

6. BUILD AND TRAIN PREDICTIVE MODELS

LSTM is mainly used in deep learning in three modes: one-to-many, many-to-one, and many-to-many. One-to-many sequence problems are sequence problems in which a single input value from t-step gives a vector of multiple time-steps. This is generally used for image captioning [18]. Many-to-one sequence problems take a vector of multiple time-steps as input and return a single output. Here, this output value is used for prediction value. We typically use many-to-one LSTM network for sentiment analysis or text classification, as well as for stock price prediction [19]. Many-to-many sequence problems take a vector of multiple time steps as input and returns a vector of multiple time steps as output. Here, the input time-steps and the output time-steps may be the same or different depending on the problem. It can be used for machine translation and video classification [20]. In this study, we train our algorithm using a many-to-one LSTM.

We construct six types of models to predict fruit prices. First of all, six models commonly include fruit price data in Seoul and numeric data for dates and weeks defined in Section 5.1. The fruit prices data in Seoul, which is included in common in all models, are used not only for a training feature, but also as a prediction target. For example, suppose we have an input sequence of length s whose time step starts at t-step and ends at $(t + s - 1)$-step. If we take this sequence that includes fruit price data in Seoul as input data, we get one prediction value as output. The weights are updated by comparing the output of this predicted value with the $(i + s)$th fruit price data in Seoul, which is the actual value. That is, the fruit price in Seoul on the next day of the sequence is used as a prediction target.

The six models are constructed using three common data columns and the merged database from Section 5.5. We use 15 weather data obtained through Section 5.2 for Model 1, and use the data of 5 weather features obtained through multiple linear regression analysis of Section 5.3 for Model 2. Model 3 is a model in which KOSPI data is added to the training data of Model 2 above. Model 4 contain regional fruit price data in addition to the training data of Model 2. Model 5 is constructed by adding the KOSPI index and regional price data to Model 2. The last model is a model using all weather data, KOSPI index, and fruit price data by region.

The number of features used for training is 18 for Model 1, 8 for Model 2, 9 for Model 3, 12 for Model 4, 13 for Model 5, and 23 for Model 6, and predicts the price of fruits using the LSTM algorithm. The model consists of relatively fewer variables than the 108 features used in the paper [11]. Table 10 below summarizes the contents of the manufactured model.

TABLE 10. Contents of each model

Predictive model	Item Description	Number of features
Model 1	Weekly, daily data, all weather variables and fruit price in Seoul	18
Model 2	Weekly, daily data, 5 weather variables and fruit price in Seoul	8
Model 3	Weekly, daily data, 5 weather variables, KOSPI and fruit price in Seoul	9
Model 4	Weekly, daily data, 5 weather variables, regional pricing data and fruit price in Seoul	12
Model 5	Weekly, daily data, 5 weather variables, KOSPI, regional price data and fruit price in Seoul	13
Model 6	Weekly, daily data, all weather variables, KOSPI, regional pricing data and fruit price in Seoul	23

7. ANALYZING THE PERFORMANCE OF PREDICTIVE MODELS

7.1. **How to measure model performance.** We measure the error between the actual fruit price and the predicted fruit price obtained through the model to evaluate the performance of the model. The root mean square error (RMSE), which is widely used in deep learning field, is used as the error measurement method [11, 21]. The RMSE equation is as follows :

$$RMSE = \sqrt{\frac{1}{T}\sum_{t=1}^{T}(Y_t^p - Y_t^a)^2},$$

where Y_t^p is a predicted value, Y_t^a is an actual value, and T is the number of observations.

7.2. **Training process of predictive models.** In preprocessing step, data are normalized in the range (0,1) using min_max_scaling and reshaped according to the training feature. Let $S = \{x_1, ..., x_N, y\}$ be the database set of training features and a target where each x_i and y is a vector with m elements for $i = 1, 2, \ldots, N$. Here, y is the target vector and the number of elements m is the length of time series data for daily. Then we can choose the elements of S to construct the feature matrix M_k, but each M_k must contain the target vector y. If M_k takes n elements in S, then M_k becomes an m-by-n matrix.

Let R_k be the many-to-one LSTM network models for $k = 1, \ldots, 6$. To train the models R_k, we need to prepare the input data for training according to the length of the input sequence. If the length of the input sequence is s, it is possible to extract input data M_{k_j} having s-length rows from the feature matrix M_k for $j = 1, \ldots, (m-s)$. Since we do not have $(m+1)$-th row of feature matrix to update the weight, $(m - s + 1)$-th input data cannot be used for training. Now, we train the models R_k using all M_{k_j}.

The trained model R_k predicts the target price for next day using the s-length sequential data. For example, if we put the s-length sequential data which do not have target value in trained model R_k, then R_k returns a predicted scalar value. Let y_p be the vector of actual values for a period we want to predict. To obtain d predicted values for the period, we need to d sequences. Using this sequences, we estimate the predicted values with the trained model R_k and measure the RMSE between the predicted value and the target value y_p for model accuracy. The following is a pseudocode for training part of algorithm.

Let $S = \{x_1, x_2, ..., x_N, y\}$ and $M_k = \{x_1, x_2, ..., x_{m_k}\}$. Here, M_k is the subset of S and the elements/vectors of M_k consist of training feature for R_k, for $k = 1, \ldots, 6$.

```
-------------------------------------------------------------
Set LSTM units, hyperparameters and optimizer to define
        LSTM Network
Normalize the dataset into values from 0 to 1 using
        min_max_scaling
Select feature_set and organize dataset
for i <- 1 to #R_k do
  for n_epochs and batch_size do
    Train the models
  end for
end for
for i <- 1 to #R_k do
  Run Predictions
  Calculate the loss function using root_mean_square_error
  Compare prediction and real price
end for
-------------------------------------------------------------
```

We train the models while changing the values of the hyperparameters and measure the RMSE of the models to find the hyperparameters most suitable for features. 80% of the prepared data are used as training data, and 20% of those is used as test data of the training model. In this study, the six types of models classified by training features are divided into two versions according to the period of the training data. One version of models is trained with data from year 2011 to year 2018, and the other version of models is trained with data from year 2011 to year 2019. Models trained with data from year 2011 to year 2018 are used to predict fruit prices in year 2019, and models trained with data from year 2011 to year 2019 are used to predict fruit prices in year 2020, respectively. The training is conducted a total of 2916 times (sequence length(3) $\times$ output size of hidden layer(3) $\times$ LSTM layers(3) $\times$ year(2) $\times$ fruit variety(9) $\times$ types of models(6)). Here sequence length varies 5, 10, 20, and output size of hidden layer is varies 1, 3, 5, and the numbers of LSTM layers are 2, 4, 6. Table 11 below is part of the table recording the training performance.

7.3. **Learning model analysis.** In this subsection, we find the optimal hyperparameter among the experimental results in Section 7.2. The hyperparameters to be adjusted are the sequence

TABLE 11. Part of RMSE according to hyperparameter

sequence length	output size of hidden dimension	the number of LSTM layers	year	fruit	Model 1	Model 2	Model 3	Model 4	Model 5	Model 6	average RMSE
5	1	2	2020	apple(Fuji)	0.063	0.069	0.066	0.121	0.066	0.068	0.075
5	1	2	2020	apple(Tsugaru)	0.061	0.061	0.061	0.106	0.080	0.190	0.093
5	1	2	2020	apple(Hongro)	0.071	0.068	0.071	0.113	0.068	0.069	0.077
5	1	2	2020	banana	0.018	0.018	0.018	0.042	0.018	0.018	0.022
5	1	2	2020	orange(Navel USA)	0.057	0.056	0.057	0.139	0.065	0.060	0.072
5	1	2	2020	orange(Valencia USA)	0.067	0.130	0.155	0.158	0.131	0.143	0.131
5	1	2	2020	pear(Shingo)	0.060	0.061	0.105	0.106	0.066	0.062	0.077
5	1	2	2020	pear(Wonhwang)	0.071	0.070	0.067	0.138	0.073	0.097	0.086
5	1	2	2020	persimmon	0.074	0.073	0.154	0.152	0.074	0.076	0.100

length, output size of hidden layer, and the number of LSTM layers. For convenience of notation, we describe the hyperparameters as $[S : \cdot\, ; H : \cdot\, ; L : \cdot]$, where S, H, and L mean sequence length, output size of hidden layer, and the numbers of LSTM layers, respectively. We calculate the average RMSE of 54 experiments (fruit variety(9) $\times$ type of models(6)) performed with the same hyperparameters in a total of 2916 experiments. The results are separately organized by year and shown in Fig. 5. The blue bar on the graph represents year 2019 and the orange bar represents year 2020. The table 12 encloses the detailed values of the average RMSE.

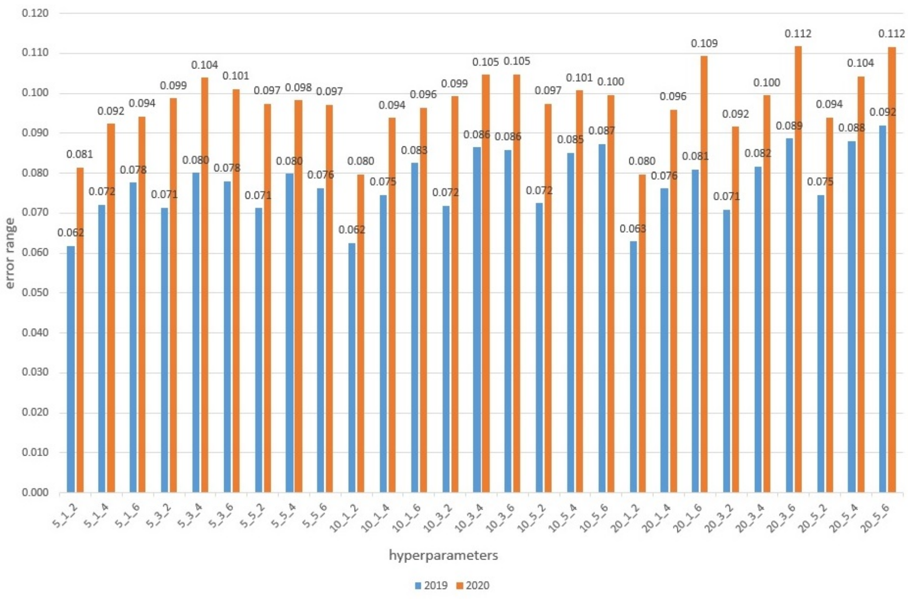

FIGURE 5. Average RMSE graph by hyperparameters

In Fig. 5 and Table 12, the hyperparameter with the smallest average RMSE in year 2019 is $[S : 5;\ H : 1;\ L : 2]$, whose value is 0.061820, and the smallest average RMSE in year

TABLE 12. Measured average RMSE value by hyperparameters

hyperparameters	2019	2020	hyperparameters	2019	2020
$[S:5; H:1; L:2]$	0.062	0.081	$[S:10; H:3; L:6]$	0.086	0.105
$[S:5; H:1; L:4]$	0.072	0.092	$[S:10; H:5; L:2]$	0.072	0.097
$[S:5; H:1; L:6]$	0.078	0.094	$[S:10; H:5; L:4]$	0.085	0.101
$[S:5; H:3; L:2]$	0.071	0.099	$[S:10; H:5; L:6]$	0.087	0.100
$[S:5; H:3; L:4]$	0.080	0.104	$[S:20; H:1; L:2]$	0.063	0.080
$[S:5; H:3; L:6]$	0.078	0.101	$[S:20; H:1; L:4]$	0.076	0.096
$[S:5; H:5; L:2]$	0.071	0.097	$[S:20; H:1; L:6]$	0.081	0.109
$[S:5; H:5; L:4]$	0.080	0.098	$[S:20; H:3; L:2]$	0.071	0.092
$[S:5; H:5; L:6]$	0.076	0.097	$[S:20; H:3; L:4]$	0.082	0.100
$[S:10; H:1; L:2]$	0.062	0.080	$[S:20; H:3; L:6]$	0.089	0.112
$[S:10; H:1; L:4]$	0.075	0.094	$[S:20; H:5; L:2]$	0.075	0.094
$[S:10; H:1; L:6]$	0.083	0.096	$[S:20; H:5; L:4]$	0.088	0.104
$[S:10; H:3; L:2]$	0.072	0.099	$[S:20; H:5; L:6]$	0.092	0.112
$[S:10; H:3; L:4]$	0.086	0.105			

2020 is $[S:20;\ H:1;\ L:2]$ and its value is 0.079671. However, the average value of the two years for the hyperparameter $[S:10;\ H:1;\ L:2]$ is 0.071096, which is smaller than 0.071625 and 0.071287 for the average value of $[S:5; H:1; L:2]$ and $[S:20; H:1; L:2]$, respectively. Therefore, hyperparameter $[S:10;\ H:1;\ L:2]$ is the optimal hyperparameter in our experiments over year 2019 and 2020.

7.4. **RMSE by model.** In this subsection, we evaluate the performance of models with hyperparameters $[S:10;\ H:1;\ L:2]$. The performance of each model is measured using the average RMSE of 9 fruits. When the average RMSE is small, we evaluate that the performance of the model is excellent. The result is shown in the Fig. 6.

Among the experimental results, Model 1 has the best performance, that is to mean that the average RMSE of Model 1 is the smallest over two years. In addition, there are noticeable performance differences between the average RMSE in year 2019 and 2020 for models except Model 1 and Model 5. Due to this performance difference, in the experiment in Section 7.3 the average RMSE in year 2020 was measured to be greater than the average RMSE in year 2019 for all hyperparameters. Here, we focus on the Model 1 and Model 5, which have small performance differences between year 2019 and 2020. The features used to train Model 1 except for common features (Weekly, Daily data, and fruit price in Seoul) are 15 weather data, and the total number of features is 18. On the other hand, the features used to train Model 5 consist of 5 weather data obtained through multiple linear regression analysis, fruit price data by region, and KOSPI, and the total number of features is 13. In terms of performance, Model 1 is the best, but Model 5 is also efficient enough. We compare the actual value and the predicted value of fruit price through Model 5.

Figures 7 and 8 show the comparison of fruit prices in 2019 and the comparison of fruit prices in 2020, respectively. The blue line on the graph is the real price and the orange line is the predicted price. The point at which the fruit prices are zero means that there is no trade in

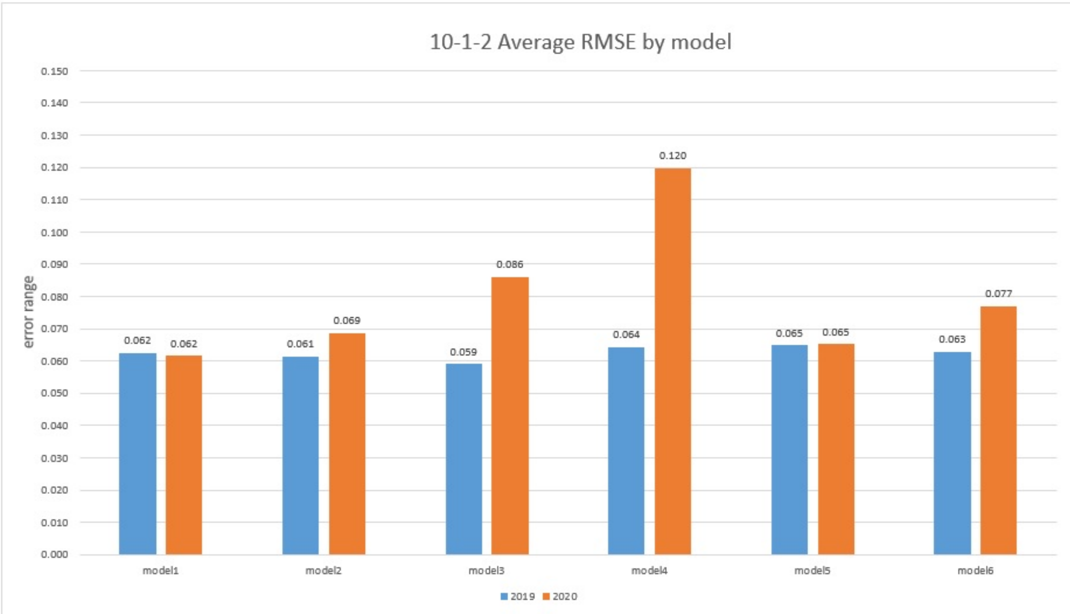

FIGURE 6. Comparison of average RMSE by model in 2019 and 2020

the market. The results show that prediction for orange (Valencia USA) in both year 2019 and 2020 does not perform well compared to other fruits. Especially for year 2020, the predictive model works well for trading day of orange (Valencia USA), but the problem is that it does not predict the duration of the trade at all.

7.5. **External factors of fruit price predictions.** In the subsection 7.4, we can see that our prediction of sales period with predictive model does not match the actual sales period of orange (Valencia USA). Therefore, this subsection will explain the reason. To do this, we look at orange (Valencia USA) sales data from year 2011 to year 2020. In the Fig. 9, the x-axis represents the sale date of the orange (Valencia USA) and the y-axis represents the price. The red line in the graph is data for 2020. In other years, orange (Valencia USA) traded from June to November on average, whereas in 2020 only traded for one month, from the third week of June to the second week of July.

We could find the reason for the low import volume and short import period of orange (Valencia USA) in 2020 in the report on Agricultural and livestock export and import trends by Korea Rural Economic Institute (KREI). According to this report, the import volume of oranges from January to September 2020 in Korea decreased compared to the previous year, due to sluggish consumption according to COVID-19 and an increase in demand for substitute fruits such as Korean mandarin [22, 23]. In particular, it is reported that the Valencia USA, one of the orange varieties selected for our model's performance test, was traded only in one

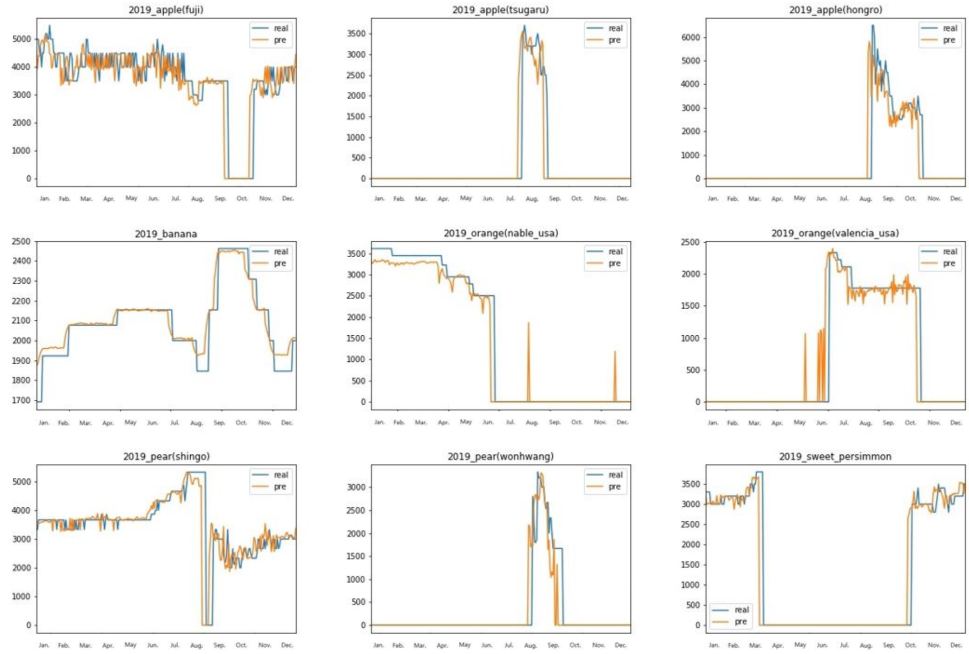

FIGURE 7. 2019 fruit price prediction graph by fruits

month, from the third week of June to the second week of July. By the result, we can see that the price of oranges in 2020 was significantly affected by unexpected external factors such as COVID-19, so the flow of fruit prices was significantly different from previous 9 years.

To investigate the effect of the 2020 orange (Valencia USA) data on the model evaluation in practice, we measure the model's performance by subtracting the 2020 orange (Valencia USA) for all hyperparameters used in Section 7.3. The orange bar in Fig. 10 represents the average RMSE in 2020, and the green bar represents the average RMSE in 2020 excluding the orange (Valencia USA). This graph shows that the RMSE for the green bar is smaller than the RMSE for the orange bar for all hyperparameters, which means that the orange price data for 2020 raises the average RMSE. Since the orange (Valencia USA) price data for 2020 is data under special circumstances, as mentioned above, we determine that the orange (Valencia USA) price for 2020 is not appropriate to evaluate our models.

7.6. **RMSE analysis excluding orange (Valencia USA).** In this subsection, we evaluate the predictive model using eight fruits except orange (Valencia USA). As in the evaluation method of Section 7.4, we measures the average RMSE of fruits, excluding the RMSE of oranges (Valencia USA) in both 2019 and 2020. The blue and orange bars in Fig. 11 represent the average RMSE for all fruits in 2019 and 2020, respectively, as shown in Fig. 6 in Section 7.4. Also, the gray and yellow bars in Fig. 11 represent the average RMSE for fruits excluding

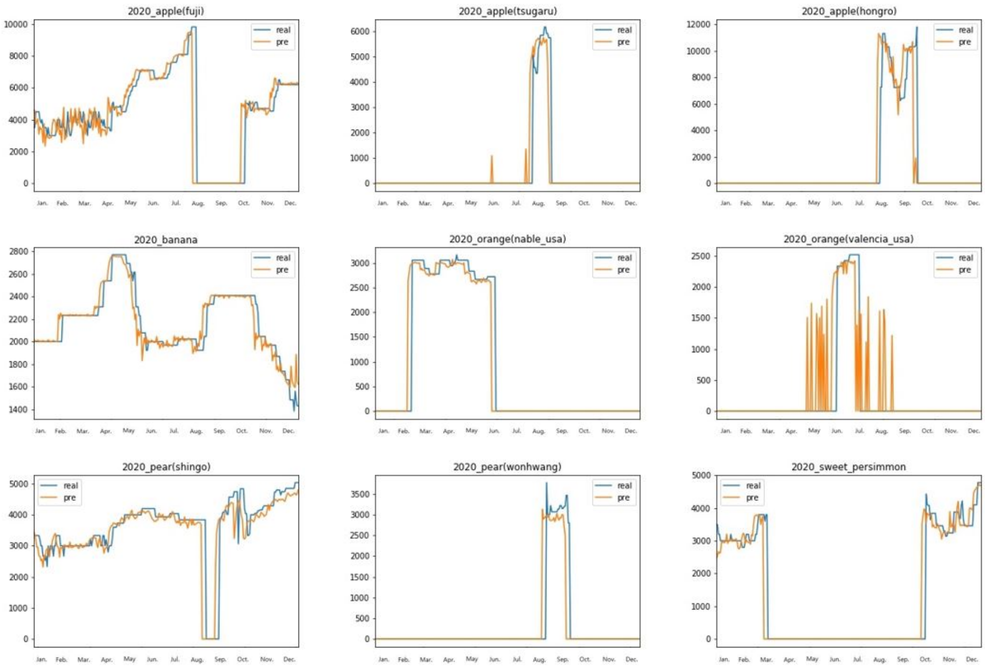

FIGURE 8. 2020 fruit price prediction graph by fruits

oranges (Valencia USA) in 2019 and 2020, respectively. In the 2019 predictive model, the average of the difference between the blue bars and the gray bars is 0.0018. This means that the average RMSE is reduced by 0.0018 by excluding orange (Valencia USA). Also, in the 2020 predictive models, if orange (Valencia USA) is excluded from orange bars, the average RMSE decreases by 0.0065, which is about 3.58 times the result of 2019. This shows that, as analyzed in Section 7.5, the orange (Valencia USA) price in 2020 was formed under special circumstances and it is more difficult to predict compared to the orange (Valencia USA) price in other years.

Now, we consider the models with excellent performance among the six models. The RMSE of 2020 (yellow bar in Fig. 11) is used for the evaluation because it used the most data for training, and we assumes that the smaller the RMSE, the better the performance. Models 1, 2, 5, and 6 have the average RMSE under 0.070 in 2020, yellow bars in Fig. 11.

First, Model 6 uses the most features. However, since the performance of Model 1 using sub-features of the features used in Model 6 is better than that of Model 6, Model 6 is excluded from the good predictive model. The average RMSEs of predictive Model 1, 2, and 5 are 0.061, 0.059, and 0.060, respectively. Also, the number of features used for training are 18, 8, and 13, respectively. The features used for training Model 2 are obtained through multiple linear regression analysis of the features used in Model 1 with the target. Through this performance

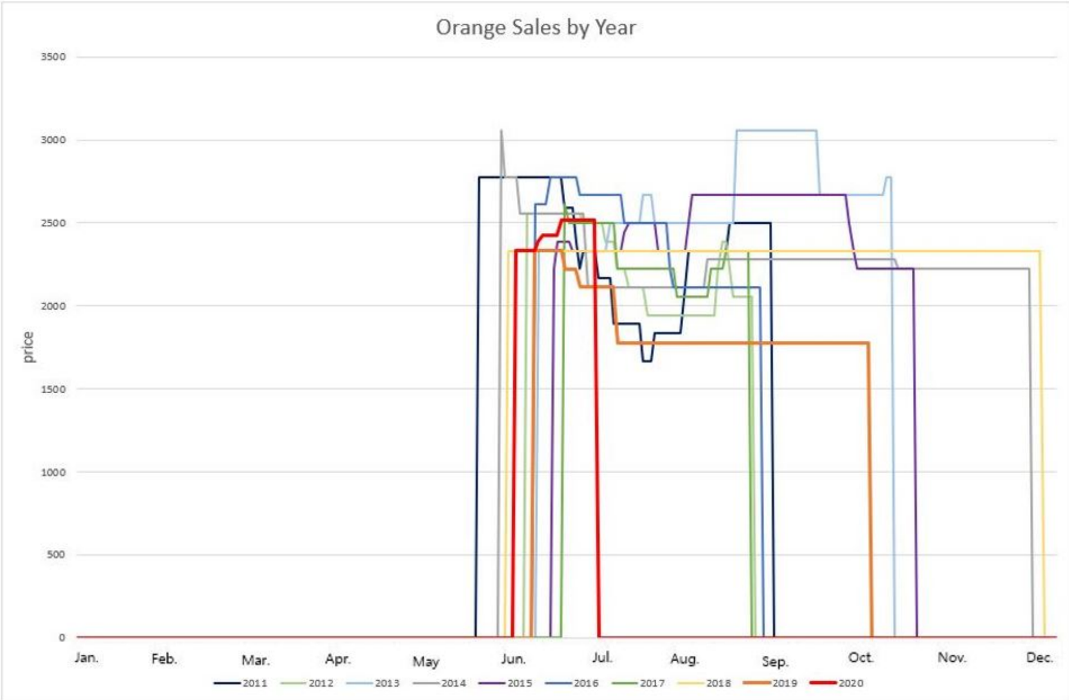

FIGURE 9. Orange (Valencia USA) annual time series graph (10 years)

comparison of Model 1 and Model 2, it is revealed that multiple linear regression analysis can exclude features that do not significantly improve performance. Therefore, in terms of average RMSE, Model 2 shows the best performance with the fewest number of features. However, the difference in the average RMSE of the three models is not noticeably large. Therefore, we want to check the distribution of RMSE by model.

Figure 12 is box plots showing the RMSE distribution of the models. Box plots are used for the purpose of quickly checking the range and median of a data set using a picture when it is difficult to visually check a lot of data. Here, the boxes mean the distribution of the models according to the fruits, and box plots show that the smaller the vertical size of the box, the more uniform performance of the model. Models with uniform performance are analyzed to be stable. The vertical sizes of the boxes of Model 1, 2, and 5 are 0.010, 0.013, and 0.008, respectively. Therefore, we evaluate that Model 5 shows the most uniform performance regardless of the fruits. Also, Model 2 has the largest performance deviation among the three models.

In conclusion, Model 1 has the largest average RMSE among the three models, and the performance deviation is intermediate between Model 2 and Model 5. In addition, the number of features used for training Model 1 is the largest. Model 2 use the smallest number of features. It has the smallest average RMSE, and has a large performance deviation. At last, Model 5

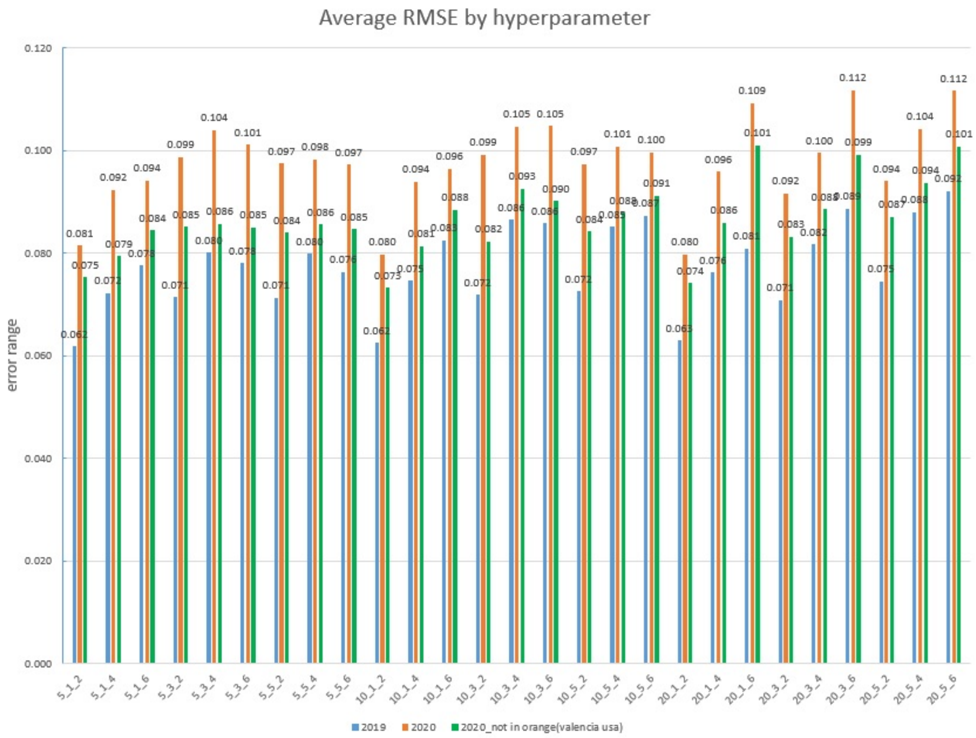

FIGURE 10. Average RMSE of fruits by hyperparameter

shows the most uniform performance although the average RMSE is greater than Model 2. Therefore, we evaluate Model 5 as the best model for application to various fruits.

8. CONCLUSION AND DISCUSSION

This paper provides a fruit price prediction algorithm using deep learning. LSTM network is employed as deep learning methods to create predictive models of time series data. As for the training features, KOSPI data and regional fruit price data are added to the weather data that have been used in other papers [10, 11]. Here, we set the weather data as a feature that affects the supply of fruit, and the KOSPI and regional fruit price data are set as the feature that affects the demand for fruit. We find the hyperparameters of the LSTM network suitable for fruit price prediction and evaluate six models composed of the prepared training features. The characteristic of our models is that the number of features is small (the model with the most features uses 23 features, and the model with the fewest features uses 8 features). During evaluation, we find an unsuitable fruit to evaluate our model, and analyze this fruit from the data. As a result, we find out that the price of the fruit was greatly affected by the influence of

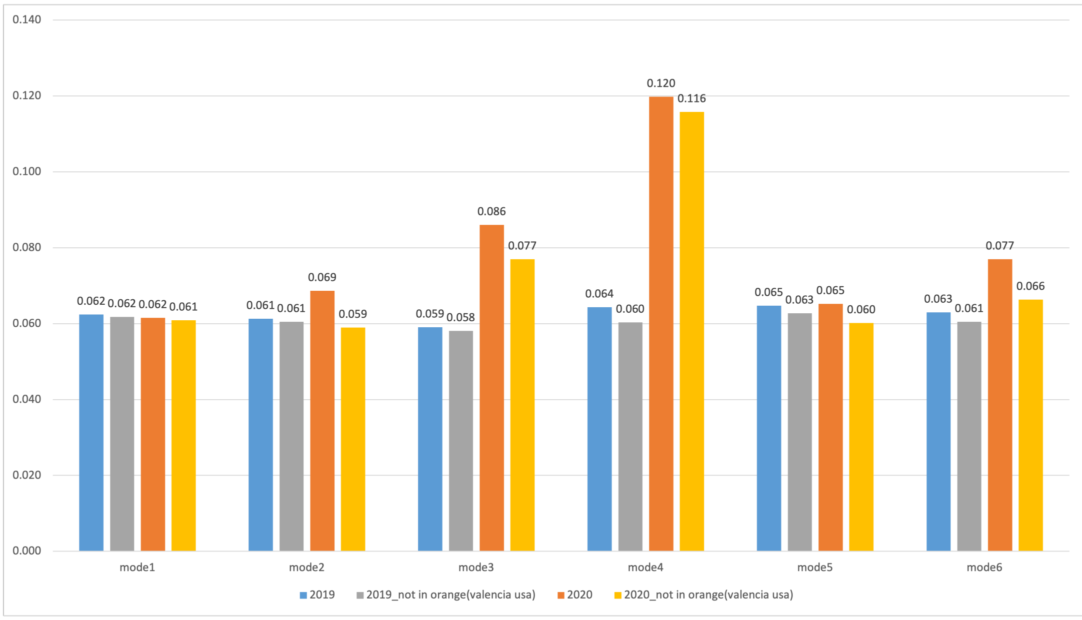

FIGURE 11. Average RMSE by model excluding orange (Valencia USA)

unexpected external factors such as COVID-19. So, we measure the performance of the models except for that fruit in order to evaluate the models except under special circumstances.

As comparing the average RMSE of Model 1 and Model 2, it is revealed that feature selection through multiple linear regression analysis in the preprocessing step do not increase the average RMSE of the model. Therefore, it is more advantageous to use Model 2 which has fewer features than to use Model 1 when we predict the fruit price with LSTM. Among the remaining models, the models with the smallest average RMSE are Model 2 and Model 5, which means that Model 2 and Model 5 have the best performance. We adds features related to the demand for fruit to the features of Model 2 in Model 5. Since Model 5 and Model 2 have similar average RMSE, then, in terms of performance, this result seems that there is no need to include the feature of demand for fruit in model. However, as a result of comparing the performance distribution of the models, Model 5 has more uniform performance than Model 2. These results show that adding features related to fruit demand increases the stability of the model. Therefore, when we use our predictive model to predict the prices of other fruits that we have not tested, we judge that using Model 5 will give more stable results.

In conclusion, this paper demonstrates that training the LSTM network using the features of supply and demand of fruits is superior in performance and stability. In addition, we show that feature selection through multiple linear regression analysis reduces the number of features used in the model while maintaining the model's performance. Also we show that weather

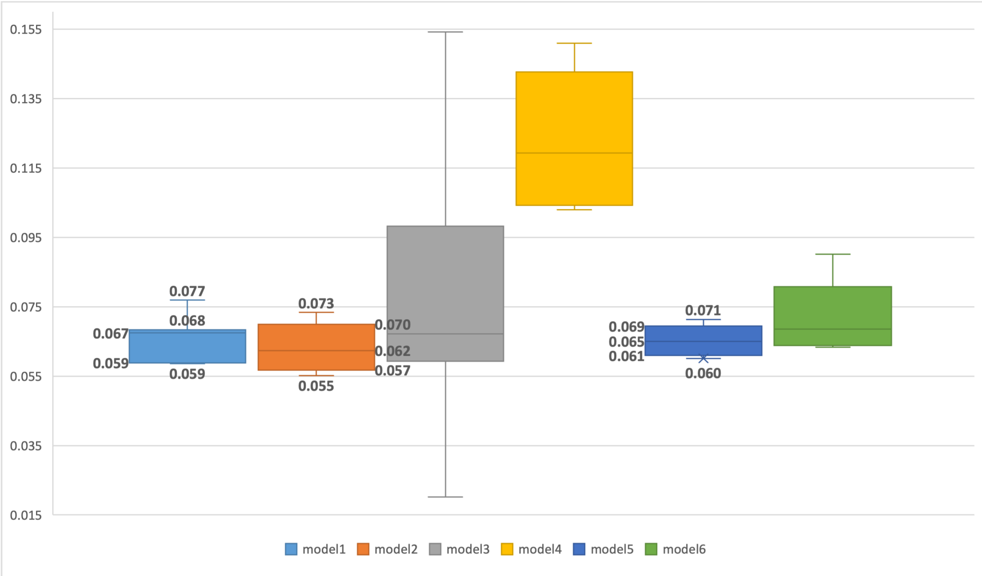

FIGURE 12. Box plots for models

data, regional fruit price data, and KOSPI data used in Model 5 are the features optimized for the fruit price predictive model.

For future work, we will use many-to-many LSTM algorithm to get weekly or long-term predictions. Also we can replace the LSTM algorithm with a bidirectional LSTM [19] or a transformer attention mechanism to get better performance.

ACKNOWLEDGMENTS

Tae-Su Park and Young Rock Kim were supported by Hankuk University of Foreign Studies Research Fund of 2021. JongHae Keum and Hoisub Kim were supported by KIAS Individual Grant (MG008512) at Korea Institute for Advanced Study. Young Rock Kim and Youngho Min were supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. 2021R1A2C1011467).

REFERENCES

[1] Weisong, M., Xiaoshuan, Z., Lingxian, Z., & Zettan, F., *A structural model for analysis of fruit supply and demand applied to grapes in China*, New Zealand Journal of Agricultural Research, **50(5)** (2007), 1359–1365.

[2] Czyżewski, A., Bieniek-Majka, M., & Czakowski, D., *Factors shaping supply-demand relations on the fruit and vegetable market in the light of the behavior of groups and producer organizations*, Management, **22(1)** (2018).

[3] Oeasby, R. C., Darroch, M. A., & Dushmaniteh, V. Y., *The demand for and supply of South African deciduous fruit exports: a dynamic analysis.*, Agrekon, **30(4)** (1991), 241–243.

[4] Weerahewa, J., Rajapakse, C., & Pushpakumara, G., *An analysis of consumer demand for fruits in Sri Lanka*, Appetite, **60** (1981–2010), 252–258.
[5] Chang, J. H., Kim, J. W., Kwak, D. E., & Aziz, N., *A Correlation Study Between Fruit Wholesale Price And Weather Factor. In Proceedings of the Korea Information Processing Society Conference.*, Korea Information Processing Society, (2017), 706–708.
[6] Ho, S. T., Ifft, J. E., Rickard, B. J., & Turvey, C. G. , *Alternative strategies to manage weather risk in perennial fruit crop production.*, Agricultural and Resource Economics Review, **47(3)** (2018), 452–476.
[7] Dong, D., & Lin, B. H., *Fruit and vegetable consumption by low-income Americans: would a price reduction make a difference?*, **60**, (2009), No. 1477-2016-121112.
[8] Rickard, B. J., & Lei, L. , *How important are tariffs and nontariff barriers in international markets for fresh fruit?.*, Agricultural Economics, **42** (2011), 19–32.
[9] Cummings Jr, R., Rashid, S., & Gulati, A. , *Grain price stabilization experiences in Asia: What have we learned?*, Food Policy, **31(4)** (2006), 302–312.
[10] Im, J. M., Kim, W. Y., Byoun, W. J., & Shin, S. J., *Fruit price prediction study using artificial intelligence.*, Journal of the convergence on culture technology, Vol. 4, No. 2, pp.197–204, May 31, (2018), pISSN 2384-0358, eISSN 2384-0366.
[11] Shin, S., Lee, M., & Song, S. K., *A Prediction Model for Agricultural Products Price with LSTM Network*. The Journal of the Korea Contents Association, **18(11)** (2018), 416-429.
[12] Schmidhuber, J., *Deep learning in neural networks: An overview. Neural networks*, New Zealand Journal of Agricultural Research, **61** (2015), 85–117.
[13] Schmidhuber J., WebCite, *Demonstrates Credit Assignment Across the Equivalent of 1,200 Layers in an Unfolded RNN.*, URL: `http://www.webcitation.org/71i6G4Jaw` [WebCite Cache ID 71i6G4Jaw] (1993).
[14] Bengio, Y., Frasconi, P., & Simard, P., *The problem of learning long-term dependencies in recurrent networks,* IEEE international conference on neural networks (pp. 1183–1188). IEEE. (1993).
[15] Raschka, S., & Mirjalili, V., *Python Machine Learning: Machine Learning and Deep Learning with Python. Scikit-Learn, and TensorFlow. Second edition ed.* (2017).
[16] Hochreiter, S., & Schmidhuber, J., *Long short-term memory*, Neural computation, **9(8)** (1997), 1735–1780.
[17] Gers, F. A., Schmidhuber, J., & Cummins, F., *Learning to forget: Continual prediction with LSTM*, Neural computation, **12(10)** (2000), 2451–2471.
[18] Vinyals, O., Toshev, A., Bengio, S., & Erhan, D. *Show and tell: A neural image caption generator,* In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 3156–3164). IEEE. (2015).
[19] Joo, I. T., & Choi, S. H., *Stock prediction model based on bidirectional LSTM recurrent neural network*, The Journal of Korea Institute of Information, Electronics, and Communication Technology, **11(2)** (2018), 204–208.
[20] Wu, Yonghui, et al., *Google's neural machine translation system: Bridging the gap between human and machine translation*, arXiv preprint arXiv:1609.08144, (2016).
[21] Kyoung-woo Cho, Yong-jin Jung, Chul-gyu Kang, Chang-heon Oh, *Conformity Assessment of Machine Learning Algorithm for Particulate Matter Prediction*, Journal of the Korea Institute of Information and Communication Engineering,(2019), Vol. 23, No. 1: 20–26.
[22] FTA Implementation Support Center, *Agricultural and livestock export and import trend Vol. 8, No. 3*, Korea Rural Economic Institute, (2020).
[23] FTA Implementation Support Center, *Agricultural and livestock export and import trend Vol. 8, No. 4*, Korea Rural Economic Institute, (2020).