JDCS
디지털콘텐츠학회논문지
Journal of Digital Contents Society
Vol. 23, No. 12, pp. 2445-2455, Dec. 2022

Check for updates

머신러닝 기반 임대주택 가격 예측 서비스 개발

이 권 우[1] · 사왈리우 니그남제 제엠바움[2] · 김 정 동[3*]
[1]선문대학교 컴퓨터융합전자공학과 석사과정
[2]선문대학교 컴퓨터융합전자공학과 박사과정
[3*]선문대학교 컴퓨터공학부 교수

Development of Rent House Price Prediction Service based on Machine Learning

Kwon-Woo Lee[1] · Soualihou Ngnamsie Njimbouom[2] · Jeong-Dong Kim[3*]

[1]Master's Course, Department of Computer Science and Engineering, Sunmoon University, Asan 31460, Korea
[2]Ph. D's Course, Department of Computer Science and Engineering, Sunmoon University, Asan 31460, Korea
[3*]Professor, Department of Computer Science and Engineering, Sunmoon University, Asan 31460, Korea

[요 약]

오늘날 공유 경제는 우리의 일상생활에 밀접하게 통합되어 있습니다. 임대 주택 플랫폼은 독특한 가격 전략으로 여러 소규모 비즈니스를 만들었다. 가격 전략을 이해하면 새로운 비즈니스에 대한 통찰력을 얻었다. 본 논문에서는 데이터 수집, 전처리, 예측의 3단계로 구성된 머신 러닝 기반 임대 주택 가격 예측 모델을 제안하고 웹 기반 플랫폼의 개발 및 구현에 대해 설명하였다. Airbnb의 Seoul, Tokyo, World 데이터셋을 이용하여 모델에 대한 비교평가를 수행하였다. 랜덤 포레스트 회귀 모델을 사용하여 평균 절대 오차, 평균 제곱 오차 및 평균 절대 오차로 성능을 보였다. 또한 임대 주택 가격 예측 웹 페이지에서는 소유자가 임대 주택 정보를 입력하고 정확한 예측 결과를 얻었다.

[Abstract]

The today sharing economy is closely integrated into our daily life. The rental house platform has created several small businesses with a unique pricing strategy. Understanding the pricing strategy can provide insight into new business. This paper proposes a machine learning-based rental house price prediction model consisting of three primary steps: data collection, pre-processing, and prediction, and describes the development and implementation of the web-based platform. A comparative evaluation was performed on the model using Airbnb's Seoul, Tokyo, and World dataset. We used a random forest regression model and showed performance with Mean Absolute Error, Mean Squared Error, and Root Mean Absolute Error. In addition, the rental house price prediction webpage allows owners to input rental house information and get accurate forecast results.

색인어 : 머신러닝, 임대 주택, 랜덤포레스트 회귀, 서포트 벡터 회귀, 다층 퍼셉트론 회귀

Keyword : Machine Learning, Rental House, Random Forest Regression, Support Vector Regression, Multi Layer Perceptron Regression.

http://dx.doi.org/10.9728/dcs.2022.23.12.2445

Received 04 November 2022; **Revised** 22 November 2022
Accepted 13 December 2022

***Corresponding Author; Jeong-Dong Kim**

Tel: +82-41-530-2221
E-mail: kjdvhu@gmail.com

Ⅰ. Introduction

The online market for a rental house is growing. One of the online marketplace platforms for the real estate rental sector is Airbnb, with over 150 million users, 650,000 property owners, and over 6 million properties registered in 2019. Airbnb serves as an alternative lodging place for tourists as well as an additional source of income for property owners [1].

Airbnb was founded in 2008 to share homes and experiences, and today has become one of the largest online accommodation booking platforms. Today, Airbnb offers more than 700 million properties in more than 220 countries in almost every country worldwide. Morgan Stanley predicts that Airbnb's users will continue to grow, and more people will use Airbnb instead of hotels. Hosts release free space online for guests looking for accommodation. The main reason for successful bookings is the right price, which is essential in balancing supply and demand in a two-sided market. Airbnb's current market share in China is 8.7%, much lower than other online platforms [2].

The Airbnb platform must help hosts set custom pricing effectively to expand market share. Thus, hosts and guests must be able to accommodate the price. Effective pricing offers help hosts earn more and entice guests to book through Airbnb rather than hotels or other platforms, which is an essential and challenging task for the Airbnb platform. Airbnb is an estate-sharing marketplace where property owners and tenants can publish properties online to allow guests to pay for accommodation [3]. As Airbnb is a platform that makes reservations in advance, unlike in-kind payments for hotels, it is essential to develop an intelligent pricing strategy that can accurately predict future house prices and offer reasonable prices to attract more tenants. Traditionally, Airbnb suggests that landlords first fix based on the price of a nearby or similar home, then increase or lower the cost if no one has booked at check-in or double the price during the holiday period. These strategies do not consider the flexible real market value of the home and can lead to difficulties in renting a home. Airbnb house prices can be affected by many factors [4]. Airbnb launched a price suggestion tool based on home properties in 2012 and improved its accuracy and flexibility in 2015 [5], but its price prediction model still needs improvement. Nowadays, hosts decorate their purchased or rented apartments to make money in the long run with Airbnb. Therefore, needs more suggestions for equipment, location, decorations, etc. [6]. The more users a site like Airbnb has, the more things to consider when pricing the rental house it offers.

Determining competitive rental rates is, thus, a complicated matter. This research addresses the challenge of creating a credible Rental House Price (RHP) prediction model using machine learning to assist the host in offering price evaluation and housing. This research's main contribution is to propose an approach for price prediction that uses multiple machine learning methods and has the advantage of learning patterns from the features of a dataset and making accurate predictions. We used three datasets: Airbnb's Seoul, Tokyo, and world. As for the machine learning method [7], experiments were performed through Linear Regression (LR), Decision Tree Regression (DTR), eXtreme Gradient Boosting Regression (XGBR), Random Forest Regression (RFR), Support Vector Regression (SVR), Multi-Layer Perceptron Regression (MLPR). The performance of each model was compared using Mean Absolute Error (MAE), Mean Squared Error (MSE), and Root Mean Squared Error (RMSE). In addition, via the commonly used machine learning methods, this work implements a web-based platform to predict the price quickly. The implemented web application can compare the information of different rental houses by predicting the price.

The rest of this research work is structured as follows. Section 2 discusses related work. Section 3 describes the proposed price prediction model. Section 4 outlines the experimental evaluations conducted on the model. Section 5 describes the implementation of the proposed model as a web. Finally, Section 6 concludes this paper.

Ⅱ. Related Work

Previous research work on price prediction models for Airbnb hosts was performed. A Gradient Boosting Machine (GBM) [8] was applied to predict the booking probability, which is the lodging demand curve. Then created a strategic model to provide a price offer. Finally, it provides accurate and personal advice, tailoring the proposal to the host's personal goals [9].

Researchers at Stanford University did a similar study. Pouya Rezazadeh Kalehbasti uses various methods such as machine learning and neutral language processing techniques [10] to develop price-prediction models. Pouya Rezazadeh Kalehbasti also uses several algorithms, including LR [11], SVR [12], K-means Clustering (KMC) [13], and Neural Networks (NNs) [14]. The main contribution of Pouya Rezazadeh Kalehbasti is the use of a tunable feature selection [15] technique that provides the 22 best features and a neutral network [16] to increase the model's accuracy. Pouya Rezazadeh Kalehbasti also added sentiment analysis to review customer reviews [17]. However, Pouya Rezazadeh Kalehbasti did not present a complete price prediction model.

Laura Lewis has studied pricing on London Airbnb listings, primarily through XGBoost and *NNs*. Laura Lewis also has over one year of host experience and performed exploratory data analysis [18].

In addition, some studies have worked on the influence characteristics of price. The features and information provided by the host are considered host-controlled variables. Features that can be controlled outside the host include variables managed externally. According to Brando MaNeil, both host-controlled and non-host-controlled variables are essential in determining Airbnb listing prices. Brando MaNeil also found that it significantly impacted listing price accuracy more when combined host-controlled with non-host-controlled variables. However, host-controlled variables have more significant implications for listing prices than non-host-controlled variables [19].

The prior studies show that renting an entire house has a higher average return than renting a room. According to Robbin Deboosere, the instant booking feature is another feature that affects average revenue. Moreover, commercial operators with multiple rentals can lower prices to get higher reviews and higher returns. These conclusions could potentially lead to the fact that the number of reviews can significantly impact booking probability and revenue. The development of nearby accommodation facilities also affects average income. As a result, hosts can earn higher revenue on average if they treat their house as a hotel rather than a temporary shared house [20].

Finally, Ang Zhu used regression and other machine learning models trained on New York City listings and attribution information from the Airbnb website, collected by Denis Gomonov and published on Kaggle. Robbin Deboosere's model analyzes the determinants of listing prices and predicts future prices for listings with publicly available information. Robbin Deboosere's model provides valuable information on the pricing strategies of hosts and other stakeholders, as well as insights into the entire hospitality industry in the context of the sharing economy [21].

Ⅲ. Proposed Approach

This study focused on predicting Airbnb prices by proposing a prediction model. The RHP prediction consists of data collection, pre-processing [22], and a prediction model. The proposed RHP prediction model, consisting of machine learning and deep learning [23] models, uses features from the pre-processed dataset obtained from online resources [24]. Using various machine learning methods, our model predicts the RHP by taking the relationships between parameters in the dataset. Figure 1 schematically depicts the proposed model.

3-1 Data Collection

The Airbnb dataset was provided from [25]. Our dataset was collected from Seoul, Tokyo, and World. The independent variable consisted of features such as accommodation, bedrooms, bathrooms, etc., and the dependent variable consisted of continuous labels.

3-2 Data Preprocessing

We first removed the expensive data to predict the price and cleaned up the dataset by removing samples with useless features and some empty features. Data transformation uses one hot encoding because there are features that require encoding. One hot encoding technique involves transforming categorical variables into a form that could help machine learning algorithms make more accurate predictions. After that, we normalized the range of the independent variable by performing data scaling to standardize the functional scope of the input dataset; therefore, the standard deviation is one, and the mean value is zero.

3-3 Prediction Model

The prediction model of our approach, called RHP, is designed to predict price by considering features. The basic assumption is that we have independent data. The studied features were put together and used as input data for a regression model to predict Airbnb's price. Machine learning is a computational algorithm automatically enhancing learning through experience with sample data.

Supervised [26], unsupervised, and reinforcement learning are the most known types of machine learning. We have experimented using conventional machine learning methods such as RFR [27], LR, XGBR [28], SVR, and DTR [29], and deep learning methods such as MLPR [30]. This machine learning algorithms learn patterns from the input data and create rules for mine in the dataset to predict Airbnb prices. We then evaluated the effectiveness of the different methods in predicting price by comparing the predicted output of this algorithm with the actual value. The test dataset was used to compute MAE, MSE, and RMSE. The characteristics of each machine learning and deep learning method used in this study will be described in more detail.

1) Linear Regression

LR is the most used prediction model for determining variables' relationships. The idea is linear, as opposed to univariate or multivariate data kinds. LR is analyzed using a probability distribution, focusing on conditional probability distributions and multivariate analysis [31].

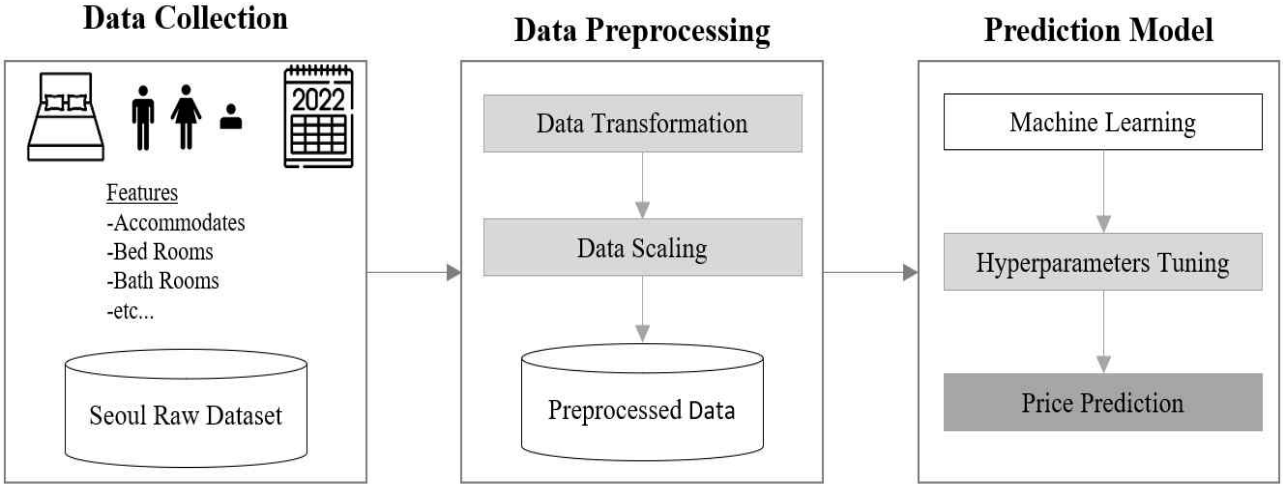

그림 1. 임대주택 가격 예측 기법의 개념 모델
Fig. 1. Conceptual model fo the rental house price prediction method

2) Decision Tree Regression

DTR is a tree-based structure that predicts the dependent variable's numerical results. DTR implements Quinlan's M5 algorithm, also called the M5P algorithm. M5P is also a tree-based structure, much like CART (classification and regression tree). Still, unlike CART, where the regression tree includes values at the leaves, the M5P-based tree has multivariate linear models. In addition, compared to the tree produced by the CART algorithm, the model trees built by the M5P algorithm are typically smaller. The working mechanism of DTR is given below. First, a tree is created using a conventional decision-tree algorithm. In this decision tree, the intra-subset variation in the class values of instances that go down each branch is minimized using a splitting criterion. The attribute, which maximizes the expected error reduction, is chosen as the root node.

After that, each leaf on the tree is pruned back. The sharp discontinuities between adjacent linear models at the pruned tree's leaves will inevitably occur. Hence a smoothing procedure is used to compensate for sharp discontinuities. DTR can predict the dependent variable's numerical results in contrast to the traditional decision tree. In addition, DTR can handle datasets with extremely high levels of dimension, and the tree generated by DTR is much smaller than the CART [32].

3) eXtreme Gradient Boosting Regression

Friedman's gradient boosting decision was used to improve XGBR. The XGBR can generate boosted trees and operate in parallel, efficiently handling regression challenges. The XGBR, like many optimization methods, employ machine learning techniques to define the optimal variables of a stated objective function. XGBR can upgrade parallel trees and provide fast and reliable models for various engineering simulations. XGBR is famous for 'regularized boosting' technology, and no such regularization step is present in the conventional gradient boosting implementation. XGBR combines the new technique with the gradient boosting method to improve the model's accuracy [33].

4) Random Forest Regression

Breiman's random forest approach is an enhanced regression tree method that has gained popularity for its robustness and flexibility in modeling the input-output functional relationship. A method of this type consists of a collection of regression trees that have been trained using distinct bootstrap samples of the training data. The final result is the average of the multiple tree outputs, and each tree functions independently as a regression function. Furthermore, the RFR's built-in cross-validation capability using out-of-bag samples offers actual prediction error estimates during the training process. Hence, real-time implementation is suitable. Additionally, RFR effectively handles high-dimensional data, unlike NNs. RFR is a non-parametric regression algorithm. The final output is calculated to obtain the average of all tree predictions [34].

5) Support Vector Regression

SVR is frequently used for curve fitting and prediction in linear and nonlinear regression models. Support Vectors Machine (SVM) is closer points to the generated hyperplane in an n-dimensional feature space that segregates the data points around the hyperplane, and SVR is based on SVM. The generalized equation for hyperplane is $y = wX + b$, where w represents weights and b represents the intercept at X = 0. Epsilon defines the margin of tolerance. The SVR model is imported from the scikit-learn python library's SVM class [35].

6) Multi-Layer Perceptron Regression

Past researchers proposed many types of Artificial Neural Networks (ANNs) [35]. These algorithms could be combined with various time-series models, including multivariable regressions, linear regressions, autocorrelations, and other statistical analysis methods. ANNs multilayer perceptron (MLP) and radial basis function (RBF) processes are suitable for regression modeling. The MLP structure consists of three layers of neurons. Each layer consists of several processing units linked to one another by weights in the subsequent layer. The perceptron forms a linear combination based on its input weights to compute a single output from several real-valued inputs. Therefore, the MLP uses several nonlinear inputs to generate a single output.

MLP network is a supervised learning approach that trains its neurons using either a standard gradient descent technique or a nonlinear optimization algorithm, such as the conjugated gradients algorithm. MLP network training can significantly accelerate the convergence rate of a weighted gradient descent algorithm. A MLPR is a variant of an MLP with no activation function in the output layer or an identity function as an activation function, MLPR implements an MLP that trains using backpropagation. Therefore, the square error is the loss function in MLPR, and the output is a set of continuous values. MLPR also enables multi-output regression, which allows a sample to have many targets [36].

Ⅳ. Experimentation

This section describes the experimental setup, dataset, hyperparameter tuning of the machine learning model, and the experiment's results. In this study, extensive experiments were performed to compare the performance of various well-known machine learning methods.

4-1 Dataset

Our dataset collected rental house information in Seoul, Tokyo, and the world. The Seoul, Tokyo, and World dataset provides 8,519, 6,371, and 494,954 data, respectively. To predict the price, nine features were used, including accommodates, bedrooms, and bathrooms, by removing outliers, missing values, and useless features. The pre-processed dataset has 3,623, 3,117, and 315,115 data, respectively.

4-2 Experimental Setup

Deep learning and machine learning necessitate platforms with excellent computing performance. Table 1 shows the details of the experimental setup used in the experiments. TensorFlow, an open-source machine learning framework, was used to build the model. All experiments were performed on a personal computer with 62.7 GB RAM, an Intel Core i9-9900K CPU, an NVIDIA 1080 Ti GTX GPU, and running 64-bit Ubuntu 18.04 as the operating system. Python version 3.9.0 was used to train the proposed approach, and the TensorFlow 2.5.0 library and CUDA-Toolkit 11.2 were used on the GPU.

표 1. 성능을 위한 실험 설정
Table 1. Experimental setup for performance

Name	Description
RAM	62.7
CPU	Intel Core i9-9900k (3.60Hz)
GPU	NVIDIA GTX 1080 Ti ×4
OS	Ubuntu 18.04 64 bit
Python Version	3.9.0
TensorFlow	2.5.0
CUDA Version	11.2.0

4-3 Hyperparameter Tuning for Machine Learning Models

The random search algorithm was used to adjust the hyperparameters of conventional machine learning methods. The random search algorithm exhaustively searches for a set of hyperparameters that optimize the model's results by taking a user-defined range of values as input. This process is performed during the training stage on the training set. The hyperparameters with the highest performance for each ML model were selected for the final models.

Table 2 describes the various optimizable parameters used during price prediction using traditional machine learning. In Table 2 the selected hyperparameter values of each algorithm can be seen after the random search algorithm that carefully determined the best value from the range of values as shown in the experimental source code. Table 2 is hyperparameter values for the Seoul, Tokyo, and World dataset, respectively.

표 2. 실험의 머신러닝 모델 하이퍼파라미터
Table 2. Hyperparameters for the various machine learning models

Machine Learning	Parameters	Seoul	Tokyo	World
LR	fit_intercept	True	True	True
	Normalize	False	False	False
DTR	min_samples_split	10	100	100
	min_samples_leaf	10	10	10
	max_features	auto	log2	log2
	max_depth	15	10	10
	Criterion	squared_error	absolute_error	squared_error
XGBR	n_estimators	500	500	1000
	max_depth	6	7	6
	learning_rate	0.01	0.01	0.1
	Nthread	4	4	4
	Object	reg:linear	reg:linear	reg:linear
RFR	n_estimators	1000	500	500
	min_samples_split	2	5	5
	min_samples_leaf	2	2	2
	max_features	sqrt	sqrt	sqrt
	max_depth	15	30	30
	Criterion	squared_error	absolute_error	absolute_error
SVR	Kernel	rbf	rbf	rbf
	Gamma	scale	scale	scale
	C	10	10	10
MLPR	hidden_layer_size	(10,)	(10,)	(10,)
	Activation	tanh	logistic	logistic
	solver	sgd	sgd	sgd
	learning_rate	constant	constant	constant
	learning_rate_init	0.001	0.001	0.001

4-4 Result

Our dataset was split into a training set and a test set to evaluate the prediction model's performance and avoid overfitting. This study compared commonly used regression evaluation metrics of machine learning tested in LR, DTR, XGBR, RFR, SVR, and MLPR. The mathematical equations for the regression evaluation metrics are defined as:

$$MAE = \frac{1}{n}\sum_{i=1}^{n}\left|\hat{y}_i - y_i\right| \quad (1)$$

MAE: Eq. (1) measures the average magnitude of the errors in a set of predictions without considering their direction. It's the average over the test sample of the absolute differences between prediction and actual observation where all individual differences have equal weight.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \quad (2)$$

MSE: Eq. (2) measures the amount of error in statistical models. MSE evaluates the average squared difference between the observed and predicted values. When a model has no error, the MSE equals zero.

$$RMSE = \sqrt[2]{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \quad (3)$$

RMSE: Eq. (3) is an evaluation that measures the error's average magnitude. Eq. (3) is the square root of the average square differences between prediction and actual observation. Since the errors are squared before averaged, the RMSE gives a relatively high weight to significant errors. This means the RMSE should be more useful when significant errors are undesirable.

표 3. 서울 데이터셋의 머신러닝 모델의 성능 비교
Table 3. Performance comparison of machine learning models (Seoul dataset)

Models	MAE	MSE	RMSE
LR	18.39	830.90	28.82
DTR	16.41	783.69	27.99
XGBR	18.34	831.84	28.84
RFR	14.10	583.02	24.14
SVR	15.22	776.17	27.85
MLPR	16.27	722.55	26.88

표 4. 도쿄 데이터셋의 머신러닝 모델의 성능 비교
Table 4. Performance comparison of machine learning models (Tokyo dataset)

Models	MAE	MSE	RMSE
LR	20.65	1003.14	31.67
DTR	18.49	923.45	30.38
XGBR	20.56	1003.66	31.68
RFR	17.90	835.19	28.88
SVR	17.91	895.77	29.92
MLPR	19.56	887.38	29.78

표 5. 세계 데이터셋의 머신러닝 모델의 성능 비교
Table 5. Performance comparison of machine learning models (World dataset)

Models	MAE	MSE	RMSE
LR	26.65	1185.64	34.43
DTR	25.88	1297.01	36.01
XGBR	26.66	1185.67	34.43
RFR	19.80	727.55	26.97
SVR	24.53	1083.00	32.90
MLPR	23.80	969.13	31.13

Performance evaluation is suitable for selecting the best machine learning for prediction models after training. After regression analysis of price prediction performed in this work, tables 3, 4, and 5 show the performance results of all machine learning methods applied on the test set of the different collected datasets. We observed that RFR performed better than the other used machine learning methods. The proposed RFR approach displayed MAE, MSE, and RMSE of 14.10, 583.02, and 24.14, respectively as shown in Table 3. Table 4 shows the MAE, MSE, and RMSE of 17.90, 835.19, and 28.88, respectively. In addition, Table 5 shows the MAE, MSE, and RMSE of 19.80, 727.55, and 26.97, respectively. Therefore, the presented RFR results can be considered competitive compared to other applied prediction methods. RFR is the most appropriate algorithm for our work out of the machine learning methods assessed on the test set.

We split the dataset into the training of 80% and the test of 20% sets for our experiments, trained the model, and the hyperparameters were tuned using K-fold cross-validation on the validation set during the randomized search step. In our case, K = 5, training and validation were split into K subsets for validation and K-1 subsets for training during each fold. These results demonstrate that our model performed equally well on the training set and the unobserved data.

Features of each dataset were passed to the RFR model, and feature importance was calculated according to the function. Figure 2, 3, and 4 are graphs for arbitrary RFR. Since the RFR algorithm is random, different feature importance were obtained in each calculation. Figure 2 shows that the most important feature is bedrooms, with a score of 0.32.

In Figure 2, the importance order of each feature is bedrooms > latitude > longitude > accommodates > bathrooms > room type entire home/apt > minstay > room type shared room > room type private room. Figure 3 shows that the most important feature is latitude, with a score of 0.28.

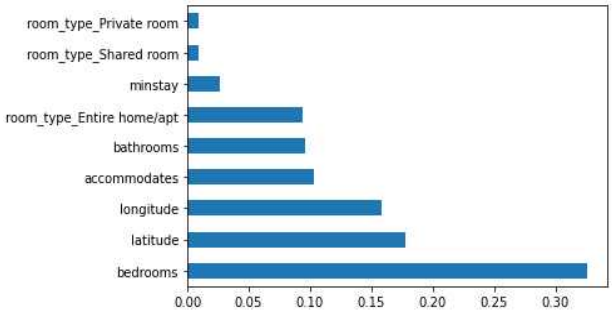

그림 2. RFR의 특성중요도 분석 결과 (서울 데이터셋)
Fig. 2. Results of feature importance to RFR (Seoul dataset)

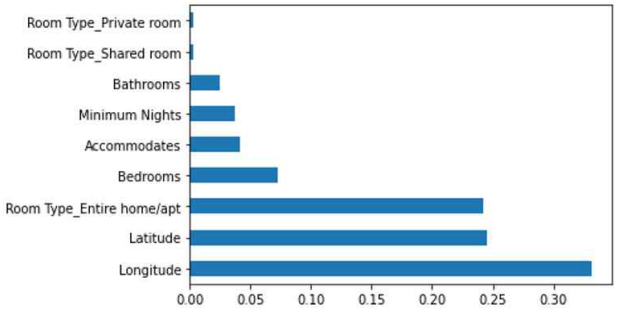

그림 3. RFR의 특성중요도 분석 결과 (도쿄 데이터셋)
Fig. 3. Results of feature importance to RFR (Tokyo dataset)

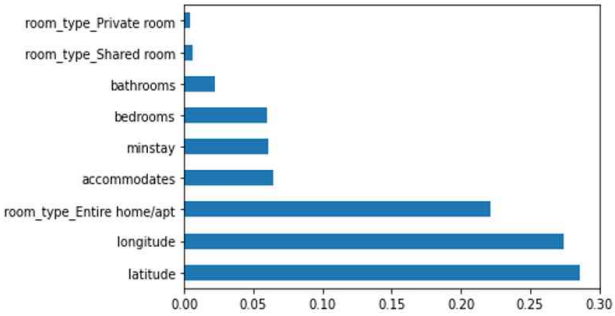

그림 4. RFR의 특성중요도 분석 결과 (세계 데이터셋)
Fig. 4. Results of feature importance to RFR (World dataset)

V. Implementation

This section discusses the model's actual web implementation and the test results obtained. The RHP prediction consists of data pre-processing and prediction. The data pre-processing phase performs data scaling. The prediction phase accurately predicts the price using the input data. During the deployment, a wireless local area network was set up to enable wireless communication between the web client and the server. In this paper, one PC was used as the server, and the other was used as the client side. The

PC was running on Windows 10 64-bit and running on Chrome. Python's Flask web framework is used to build the server. Figure 5 provides an overview of the proposed implementation. First, the user inputs data about the rental house into the interface. Then data goes through pre-processing (data cleaning and normalization) to be further used by the RFR model to predict the owner's rental house price.

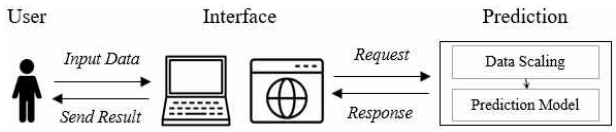

그림 5. 웹 기반 RHP 예측 개요
Fig. 5. Web-based RHP prediction conceptual overview

그림 6. 사용자 입력 양식 스냅샷
Fig. 6. Snapshot of the user input form

Figures 6, 7, and 8 show the snapshot of the RHP application. Figure 6 shows a snapshot of the rental house information entered by the user. Users can predict RHP by entering values for Accommodation, bedrooms, bathrooms, latitude, longitude, minstay, and room type. Figure 7 is a snapshot of the results for Figure 6. First, the user can know the rental house information and the predicted value of RHP, and the user's surrounding rental house information and the surrounding RHP were displayed for comparison.

그림 7. 위치별 RHP 예측의 스냅샷 결과
Fig. 7. Snapshot result of RHP prediction by location

그림 8. 선택한 위치에서 임대 주택 정보 비교 스냅샷
Fig. 8. Snapshot of the rental house's information comparison from the selected location

Figure 8-① is the rental housing data in the dataset and Figure 8-② is the data entered by the user. Figure 7 displays the list of houses as markers on the map based on the user-selected preference. Figure 8 shows rental houses' information as a map marker, implemented to allow users to compare the rental house's prices in the same area. The user can insert room preference information (accommodates, bedrooms, bathrooms, latitude, longitude, and mainstay) on this page and select a room type.

Ⅵ. Conclusion

This paper proposes a web-based RHP prediction approach integrated with machine learning technology. The data pre-processing phase allows the system to make more accurate predictions. For the same purpose, this work compared the performance of the proposed algorithm with LR, DTR, XGBR, RFR, SVR, and MLPR. According to the experimental results on the Seoul Airbnb Dataset, the RFR model showed the best performance with MAE, MSE, and RMSE of 14.10, 583.02, and 24.14, respectively. Furthermore, in the Tokyo dataset, MAE, MSE, and RMSE were 17.90, 835.19, and 28.88, respectively, and in the world dataset, MAE, MSE, and RMSE were 19.80, 727.55, and 26.97, respectively. Data transformation and scaling were used to convert the data from text format to categorical format and perform normal distribution of the different selected features. The current implementation uses the owner's house data as input data to display price prediction results to the user. Users can also compare information from different houses on the map. Our dataset is constructed by removing all noise values from the Airbnb dataset. However, our model has limitations in not predicting accurate prices when noise values are input, which can be solved by adequately adding noise and proceeding with learning. Also, our model cannot determine the peak and off-peak times for forecast prices. Therefore, we plan to implement an updated version of the current prediction model to provide appropriate renting periods with lower prices to users.

Acknowledgement

This work was supported by the National Research Foundation of Korea(NRF) grant funded by the Korea government (MSIT) (No. 2019R1F1A1058394).

References

[1] Kokasih, M. F., et al, “Property Rental Price Prediction Using the Extreme Gradient Boosting Algorithm”, *International Journal of Informatics and Information Systems*, Vol. 3, No. 2, pp. 54-59, September 2020. https://doi.org/10.47738/ijiis.v3i2.65

[2] Hill, D, et al. (2020). “How much is your spare room worth?”, *Ieee Spectrum*, Vol. 52, No. 9, pp. 32-58, August 2015. https://doi.org/10.1109/MSPEC.2015.7226609

[3] Priambodo, F. N., & Sihabuddin, A., “An extreme learning machine model approach on airbnb base price prediction”, *International Journal of Advanced Computer Science and Applications*, Vol. 11, No. 11, pp. 179-185, 2020. https://doi.org/10.14569/IJACSA.2020.0111123

[4] Liu, Y., “Airbnb Pricing Based on Statistical Machine Learning Models”, in *2021 International Conference on Signal Processing and Machine Learning (CONF-SPML)*, pp. 54-59, November 2021. https://doi.org/10.1109/CONF-SPML54095.2021.00042

[5] Feng, L., “A scorecard breaking down everyone from Xiaozhu, Tujia to Airbnb”, *EqualOcean*, June 2019.

[6] Yang, S., “Learning-based airbnb price prediction model”, in *2021 2nd International Conference on E-Commerce and Internet Technology*, pp. 283-288, March 2021. https://doi.org/10.1109/ECIT52743.2021.00068

[7] Kang, I. A., Ngnamsie Njimbouom, S., et al., “DCP: Prediction of Dental Caries Using Machine Learning in Personalized Medicine”, *Applied Sciences*, Vol. 12, No. 6, pp. 1-15, March 2022. https://doi.org/10.3390/app12063043

[8] Raghunath, K. K., Kumar, V. V., et al., “XGBoost Regression Classifier (XRC) Model for Cyber Attack Detection and Classification Using Inception V4”, *Journal of Web Engineering*, Vol. 21, No. 4, pp. 1295-1322, April 2022. https://doi.org/10.13052/jwe1540-9589.21413

[9] Ye, P., et al., “Customized regression model for airbnb dynamic pricing”, in *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*, pp. 932-940, July 2018. https://doi.org/10.1145/3219819.3219830

[10] Luo, T. T., et al., “Crystal Engineering: Toward Intersecting Channels from a Neutral Network with a bcu-Type Topology”, *Angewandte Chemie*, Vol. 117, No. 32, pp. 6217-6221, September 2005. https://doi.org/10.1002/ange.200462674

[11] Chattrairat, K., Wongseree, W., et al., “Comparisons of Machine Learning Methods of Statistical Downscaling Method: Case Studies of Daily Climate Anomalies in

Thailand", *Journal of Web Engineering*, Vol. 20, No. 5, pp. 1397-1424, July 2021. https://doi.org/10.13052/jwe1540-9589.2057

[12] Moon, C. B., Lee, J. Y., et al., (2020). "How to Retrieve Music using Mood Tags in a Folksonomy", *Journal of Web Engineering*, Vol. 20, No. 8, pp. 2335-2360, November 2021. https://doi.org/10.13052/jwe1540-9589.2086

[13] Duo, J., Zhang, P., et al., "A K-means Text Clustering Algorithm Based on Subject Feature Vector", *Journal of Web Engineering*, Vol. 20, No. 6, pp. 1935-1946, October 2021. https://doi.org/https://doi.org/10.13052/jwe1540-9589.20612

[14] Zare, A., et al., "A Hybrid Recommendation System Based on the Supply Chain in Social Networks", *Journal of Web Engineering*, Vol. 21, No. 3, pp. 633-660, February 2022. https://doi.org/10.13052/jwe1540-9589.2133

[15] Vandic, D., Frasincar, F., et al., "A framework for product description classification in e-commerce", *Journal of Web Engineering*, Vol. 17, No. 1, pp. 001-027, October 2017.

[16] Guo, H., Li, et al., "Research on indoor wireless positioning precision optimization based on UWB", *Journal of Web Engineering*, Vol. 19, No. 7, pp. 1017-1048, December 2020. https://doi.org/10.13052/jwe1540-9589.19785

[17] Rezazadeh Kalehbasti, et al., "Airbnb price prediction using machine learning and sentiment analysis", in *International Cross-Domain Conference for Machine Learning and Knowledge Extraction*, pp. 173-184, August 2021. https://doi.org/10.1007/978-3-030-84060-0_11

[18] Lewis, L., "Predicting Airbnb prices with machine learning and deep learning", *Medium-Toward Data Science*, 2019.

[19] McNeil, B. (2020). "Price Prediction in the Sharing Economy: A Case Study with Airbnb data", *University of New Hampshire*, pp. 1-16, May 2020. https://scholars.unh.edu/honors/504/

[20] Deboosere, R., et al., "Location, location and professionalization: a multilevel hedonic analysis of Airbnb listing prices and revenue", *Regional Studies, Regional Science*, Vol. 6, No. 1, pp. 143-156, April 2019. https://doi.org/10.1080/21681376.2019.1592699

[21] Zhu, A., et al., "Machine learning prediction of new york airbnb prices", in *2020 Third International Conference on Artificial Intelligence for Industries (AI4I)* , pp. 1-5, September 2020. https://doi.org/10.1109/AI4I49448.2020.00007

[22] Elbasani, E., & Kim, J. D., "LLAD: Life-log anomaly detection based on recurrent neural network LSTM", *Journal of Healthcare Engineering*, Vol. 2021, No. 8829403, pp. 1-7, February 2021. https://doi.org/10.1155/2021/8829403

[23] Elbasani, E., Njimbouom, S. N., et al., "GCRNN: graph convolutional recurrent neural network for compound–protein interaction prediction", *BMC bioinformatics*, Vol. 22, No. 5, pp. 1-14, January 2022. https://doi.org/10.1186/s12859-022-04560-x

[24] opendatasoft Available: https://public.opendatasoft.com/explore/

[25] Elbasani, E., & Kim, J. D., "AMR-CNN: Abstract Meaning Representation with Convolution Neural Network for Toxic Content Detection", *Journal of Web Engineering*, Vol. 21, No. 3, pp. 677-692, February 2022. https://doi.org/10.13052/jwe1540-9589.2135

[26] Rodriguez-Galiano, V., et al., "Machine learning predictive models for mineral prospectivity: An evaluation of neural networks, random forest, regression trees and support vector machines", *Ore Geology Reviews*, Vol. 71, pp. 804-818, December 2015. https://doi.org/10.1016/j.oregeorev.2015.01.001

[27] Fan, J., Wang, X., et al., "Comparison of Support Vector Machine and Extreme Gradient Boosting for predicting daily global solar radiation using temperature and precipitation in humid subtropical climates: A case study in China", *Energy conversion and management*, Vol. 164, pp. 102-111, May 2018. https://doi.org/10.1016/j.enconman.2018.02.087

[28] Pekel, E., "Estimation of soil moisture using decision tree regression", *Theoretical and Applied Climatology*, Vol. 139, No. 3, pp. 1111-1119, November 2020. https://doi.org/10.1007/s00704-019-03048-8

[29] Murtagh, F., "Multilayer perceptrons for classification and regression", *Neurocomputing*, Vol. 2, No. 5-6, pp. 183-197, July 1991. https://doi.org/10.1016/0925-2312(91)90023-5

[30] Kavitha, S., et al., "A comparative analysis on linear regression and support vector regression", in *2016 online international conference on green engineering and technologies (IC-GET)*, pp. 1-5, November 2016. https://doi.org/10.1109/GET.2016.7916627

[31] Rathore, S. S., & Kumar, S., "A decision tree regression based approach for the number of software faults prediction", *ACM SIGSOFT Software Engineering Notes*, Vol. 41, No. 1, pp. 1-6, January 2016. https://doi.org/10.1145/2853073.2853083

[32] Zhou, J., Qiu, Y., et al., "Developing a hybrid model of Jaya algorithm-based extreme gradient boosting machine to estimate blast-induced ground vibrations", *International Journal of Rock Mechanics and Mining Sciences*, Vol. 145, No. 104856, pp. 1-12, September 2021. https://doi.org/10.1016/j.ijrmms.2021.104856

[33] Adusumilli, S., Bhatt, D., et al. “A low-cost INS/GPS integration methodology based on random forest regression”, *Expert Systems with Application*, Vol. 40, No. 11, pp. 4653-4659, September 2013.
https://doi.org/10.1016/j.eswa.2013.02.002

[34] Parbat, D., & Chakraborty, M., “A python based support vector regression model for prediction of COVID19 cases in India”, *Chaos, Solitons & Fractals*, Vol. 138, No. 109942, pp. 1-5, September 2020.
https://doi.org/10.1016/j.chaos.2020.109942

[35] Yang, G. R., & Wang, X. J., “Artificial neural networks for neuroscientists: A primer”, *Neuron*, Vol. 107, No. 6, pp. 1048-1070, September 2020.
https://doi.org/10.1016/j.neuron.2020.09.005

[36] Chen, T. L., et al., “A causal time-series model based on multilayer perceptron regression for forecasting taiwan stock index”, *International Journal of Information Technology & Decision Making*, Vol. 18, No. 6, pp. 1967-1987, November 2019.
https://doi.org/10.1142/S0219622019500421

이권우(Kwonwoo Lee)

2021년 : 선문대학교 대학원 (공학석사, 컴퓨터융합전자공학과)

2021년～현 재: 선문대학교 컴퓨터융합전자공학과 석사과정
※관심분야: 머신러닝, 인공지능, 데이터분석 등

Soualihou Ngnamsie Njimbouom

2021년 : 선문대학교 대학원 (공학석사, 컴퓨터융합전자공학과)
2022년 : 선문대학교 대학원 (공학박사, 컴퓨터융합전자공학과)

2020년～2022년: 선문대학교 컴퓨터융합전자공학과 석사과정
2022년～현 재: 선문대학교 컴퓨터융합전자공학과 박사과정
※관심분야 : 머신러닝, 인공지능, 바이오인포메틱스 등

김정동(Jeong-Dong Kim)

2008년 : 고려대학교 대학원 (공학석사, 컴퓨터공학부)
2012년 : 고려대학교 대학원 (공학박사, 컴퓨터공학부)

2012년～2015년: 고려대학교 정보대학 연구교수
2015년～2015년: 호주 시드니대학교 정보기술대학 방문교수
2016년～현 재: 선문대학교 컴퓨터공학부 부교수
※관심분야: 해석가능한 인공지능, 바이오인포메틱스, 멀티모달 데이터 데이터분석 등