Towards effective health management for future smart cities: Understanding and forecasting bike-sharing demand in Seoul

Subeen Leem [1], Jisong Oh [1], Jihoon Moon [1,2(*)],
Mucheol Kim [3], and Seungmin Rho [4(*)]

[1] Department of Medical Science, Soonchunhyang University, Asan, South Korea

[2] Department of AI and Big Data, Soonchunhyang University, Asan, South Korea

[3] Department of Software, Chung-Ang University, Seoul, South Korea

[4] Department of Industrial Security, Chung-Ang University, Seoul, South Korea

* Corresponding authors
Jihoon Moon: jmoon22@sch.ac.kr
Seungmin Rho: smrho@cau.ac.kr

Abstract

Active bike sharing is an excellent method to address crucial challenges, such as expanding bike communities, lowering transportation costs, alleviating traffic congestion, reducing greenhouse gas emissions, and enhancing public health in cities. This study proposes an online learning-based two-stage forecasting model based on a low-computing environment with insufficient data for robust and fast multistep-ahead prediction for bike-sharing demand in Seoul, South Korea. The model was applied to a Seoul bike-sharing demand dataset subjected to exploratory data analysis (EDA). We split the dataset into insufficient training and sufficient testing sets. In the first stage, we generated the prediction values for the random forest, extreme gradient boosting, and Cubist methods in the training and testing sets. In the second stage, we used the ranger package (a fast implementation of random forest) trained with the external factors and prediction values using time-series cross-validation for multistep-ahead prediction for one hour to one day later. We compared the prediction performance of the proposed model with that of 10 existing Seoul bike-sharing prediction models to verify its superiority. In addition, we reported the relationships between external factors and bike-sharing demand using interpretability methods and EDA.

Keywords

bike-sharing demand, online learning, multistep-ahead prediction, ensemble learning, time-series cross-validation

Abbreviations

Bagging	bootstrap aggregating
CART	classification and regression tree
Ctree	conditional inference tree
CVRMSE	coefficient of variation of the root mean square error
DT	decision tree
EDA	exploratory data analysis
EL	ensemble learning
GBM	gradient boosting machine
GD	gradient descent
KNN	k-nearest neighbor
MAE	mean absolute error
MIMO	multiple-input multiple-output
ML	machine learning
MLR	multiple linear regression
MRF	multivariate random forest
PDP	partial dependence plot
RABOLA	ranger-based online learning approach
RF	random forest
RMSE	root mean square error
RRF	regularized random forest
SD	standard deviation
SVM	support vector machine
TSCV	time-series cross-validation
UCI	University of California, Irvine
XGBoost	extreme gradient boosting

1. Introduction

Currently, bike-sharing systems are gaining popularity worldwide [1,2]. These public transport systems provide people with bicycles for free or a minimum fee for short-term use [2,3]. Most bike-sharing systems provide a service that allows users to rent and replace bikes from one bicycle station to another on the same network [4]. Bike-sharing systems have received widespread attentiveness in current years as part of an initiative to encourage the usage of bicycles and minimize the negative influence of transport activities by replacing other modes of transportation, such as cars and buses [5]. The main motivation for this system is its focus on environmental and social welfare [6,7]. For example, active bike sharing can be a great way to address critical issues, such as building larger bike communities, reducing transportation costs, resolving traffic congestion, minimizing greenhouse gas emissions, and improving public health.

Bike-sharing platforms have been rapidly spreading worldwide with the remarkable advances in intelligent transportation systems and information technology since the first decade of the 2000s [8]. These systems simplify public bike rental. Mobile applications based on the global positioning system allow people to gather information about the nearby bike stations to rent bikes [9]. To date, many countries have implemented bike-sharing systems [6,7,10]. They have become important elements of transport systems in principal cities because of improving various aspects, such as environmental conditions, health problems, and traffic jams. After using a bicycle, people can lock it at any docking replacement in the metropolis. To expand the public use of bicycles, the managers of these systems allocate trucks to collect parked bicycles at various stations and gradually move the bicycles to their original stations [11,12].

Known in English as Seoul Bike, Ddareungi is a public rental bike-sharing platform launched in 2015 in Seoul, Republic of Korea [13]. Ddareungi was started to overcome such problems as high oil prices, traffic jams, and environmental pollution and create a healthy environment for Seoul citizens [13]. These public bicycles are available for those 13 years of age or older and manufactured using durable and lightweight materials [14]. Ddareungi provides users with driving stability and convenience. Users can rent and store bicycles at any station. The bike stations are installed in areas with heavy traffic, such as banks, schools, government offices, bus stops, residential complexes, and subway entrances and exits. In addition, with the benefit of internet-capable devices or portable phones, users can determine the stops at which they can rent and check their travel history (distance and time required) and physical activity (calories burned).

Due to these smart technologies and comfort, the utilization of bicycle rentals is growing daily in Seoul [14,15]. Therefore, the demand for bicycle rentals must be managed, and constant and suitable benefits for users must be maintained [4]. One key to these benefits is to predict the number of rental bicycles required to keep the bike-sharing platform working continuously with the growing number of users [11,12]. Sathishkumar and Cho [15,16] collected data on bike-sharing demand, weather data, and timestamp information in Seoul from December 2017 to November 2018. The evaluation method for these studies was to separate the training and testing sets based on random sampling. They constructed forecasting models based on machine learning (ML) and data mining approaches from the training set and predicted the bike-sharing demand per hour in Seoul on the testing set. They disclosed their dataset as a publicly available dataset [17].

However, while these studies are of great value, some areas need improvement. The random sampling method is difficult to generalize from unseen (out-of-sample) data in time-series forecasting because it is possible to predict a time in the past by learning future information according to sampling. Thus, these methods can only learn the bike-sharing demand pattern of in-sample data [18]. In addition, multistep-ahead forecasting is commonly used to prepare for uncertainty [19,20]. Because these studies focused on only a single forecast, they are not appropriate to prepare for uncertainty. Finally, depending on the policy of the state or local government, it may be necessary to construct an accurate forecasting model from a small dataset. However, the ratio of the training set was high, so good prediction performance when trained with only a small dataset may be difficult with these methods [21].

This paper proposed a novel multistep-ahead online learning forecasting model to address these problems. Generalizable performance can be expected because the proposed model can make accurate multistep-ahead predictions even on a small in-sample dataset in low-performance computing environments. First, we divide the Seoul bike-sharing demand dataset into short training set and long testing set periods, respectively. In the first stage, we generated the prediction values for the random forest (RF), extreme gradient boosting (XGBoost), and Cubist models, which are decision tree (DT)-based ensemble learning (EL) methods, from the training and testing sets. The DT-based methods are the most potent ML approaches used for regression and classification because they provide easy-to-understand interpretation (via their feature importance functions), adequate performance, and high accuracy [22,23]. We use an RF model to perform multistep-ahead bike-sharing demand through time-series cross-validation (TSCV) in the second stage. The RF model is trained with external elements, such as timestamp information and weather data, and the prediction values of the RF, XGBoost, and Cubist models based on the ranger package, a fast implementation of RF [24].

The principal contributions of this study are as follows:

1) We employ exploratory data analysis (EDA) using histograms, boxplots, and scatter plots for an in-depth analysis of the relationship between Seoul bike-sharing demand and weather/timestamp information to facilitate decision-making in Seoul public policy.

2) We propose a novel online learning method that can derive excellent prediction performance even in low-performance computing environments. By doing so, Seoul city could expect excellent prediction performance without incurring costs for a high-performance computing environment.

3) We use two interpretability methods to provide the interpretation of the DT-based EL methods on the training set. In addition, we compare our method and existing Seoul bike-sharing demand models in-depth to establish the excellence of the proposed process.

The rest of this article is arranged as follows. Section 2 describes the methodology for the DT-based EL method. Next, Section 3 presents an EDA between Seoul bike-sharing demand and weather/timestamp information. Section 4 offers the general procedure of the proposed method. In addition, Section 5 explains the experimental outcomes to verify the validity of the proposed scheme. Finally, Section 6 represents the conclusion and future research directions.

2. Methodology

Tree structure modeling and rule-based search are two aspects of DT [25]. The DT divides the dataset into numerous segments based on the variable's specified boundary points. Respective subsets of the dataset are constructed due to splitting, with each sample belonging to one of these subsets. The leaf or terminal nodes and split or internal nodes are the last and intermediate subsets, respectively. The DT is easier to grasp than other ML techniques because it represents a regression or classification procedure using an inference rule. However, because DT considers continuous variables noncontinuous, it may generate a large prediction error at the border point. Thus, EL approaches, such as bootstrap aggregating (bagging) and boosting, including many DTs, have been designed to overcome this problem [23,26].

2.1. Random Forest

Bagging [27] is a simple strategy to reduce result variation. Bagging takes a random dataset sample and creates a weak (single) model from each sampled dataset. Then, using voting or averaging for classification or regression, it aggregates the prediction values of the weak models to provide an output.

Thus, the bagging approach can reduce variation and noise. A DT is commonly used as a weak model in the bootstrap approach.

The RF [27,28] is a well-known example of bagging, and it works by training numerous weak DTs on an unsystematically chosen subset of all variables (features), voting or averaging the results of all DTs. Because it handles significant input features without destroying the features, RF can drive effectively on extensive datasets and produce impressive results. Furthermore, RF needs more nominal fine-tuning of its hyperparameters than other ML approaches, and it performs well with their default values. Therefore, it excels at regression and classification tasks. For model training, two typically used RF hyperparameters are considered: the number of variables available for splitting at each tree node (mTry) and the number of trees to grow (nTree) [28,29]. The resulting tree form becomes more similar as the mTry increases. Because of the various forms of each DT, the mTry, less overfitting occurs. The right value for nTree is often unimportant because growing the number of trees in the RF does not greatly enhance the model performance and increases the computational cost.

2.2. XGBoost

The gradient boosting machine (GBM) [30,31] is a well-known boosting approach that uses a gradient descent (GD) technique to adapt a new weak DT to remaining mistakes. The goal of the GD is to optimize a random cost (loss) function by iteratively altering the weight of the parameters. The GBM calculates the local gradient for a given dataset and iterates in the downward gradient direction. This method has the disadvantage of requiring a long time to develop due to repetitive training.

The XGBoost [32] was created to design an efficient and scalable GBM. To deal with the time-consuming difficulty of GBM, XGBoost uses the column subsampling method to generate more expedited training and scalability based on distributed and parallel computing. Furthermore, because missing values are automatically supplemented and recognized in order to enable boosting, XGBoost can efficiently manage them. Furthermore, to lower and balance the depth of the tree, XGBoost increases horizontally by first passing through nodes close to the root node. In addition, to effectively prevent overfitting, XGBoost can improve model performance using regularization, which applies to objective functions and the loss function at iteration t, as presented in Equations (1) and (2):

$$L^{(t)}= \sum_{i} l(y_i,\hat{y}_i) + \sum_{k} \Omega\,(f_k)\ , \tag{1}$$

$$\text{where } \Omega\,(f) = \gamma T + \frac{1}{2}\lambda \| \omega \|^{2}. \tag{2}$$

In Equation (1), l is the loss function of weak learners (classification and regression tree) used to sum the current (y_i) and previous $(\hat{y}_i)$ additive trees, and Ω is the regularization. In Equation (2), γ indicates the number of terminal leaves or nodes in a tree, and T minimizes the forecast insensitivity for respective observations. Furthermore, ω depicts the leaf weights, which can be considered the leaf output value, and λ depicts the user-definable penalty to encourage pruning.

2.3. Cubist

Cubist [22,29], like boosting, is a rule-based EL approach. After the first specimen is created, the second specimen examines whether the prior specimen overpredicted or underpredicted the outcome using an altered version of the outcome data. Quinlan's M5 model tree was used to create Cubist. The Cubist approach generates a set of "if-after-after" rules. Although it firstly creates a tree form, it tumbles each route based on the tree into a rule. Separately rule produces many sequential specimens, each with its own multivariate linear model. The appropriate specimen is used to calculate the anticipated value if the covariate set fits the rule constraints. The Cubist output offers data on variable usage, including the percentage of time each feature was used in a linear or condition model.

The following is an explanation of the broad notion of a Cubist regression model:

1. As the tree grows, it produces many leaves and branches.
2. Terminal leaves stay in a related multivariate linear model, whereas branches can be considered a set of "if-then" rules.
3. The appropriate model measures the anticipated value, assuming the covariate set fulfills the rule constraints.

The numbers of committees and neighbors are the two fundamental hyperparameters in Cubist [33]. Like the boosting strategy, the Cubist model generates a sequence of trees with altered weights and reinforces them using training committees (typically one or more). The number of neighbors is employed to adjust rule-based prediction values, and the final prediction value is a procedure of all linear models from the beginning to the end. The percentages in the Cubist outcome echo all models associated with the anticipated value (not just the terminal models).

3. Exploratory Data Analysis

The University of California, Irvine (UCI) Machine Learning Repository website [17] has a publicly available dataset we collected. This dataset provides the number of public bicycles rented per hour in Seoul, Republic of Korea, and related weather data and holiday information. The dataset period is 12 months (365 days) from Dec. 2017 to Nov. 2018, and the attribute information of the dataset is presented in Table 1.

Table 1. Information on the Seoul bike-sharing demand prediction dataset.

No.	Name	Description	Data Type
1	Date	Year-Month-Day	Timestamp
2	Rented Bike Count	Count of Bikes Rented Per Hour	Dependent
3	Hour	Hour of the Day	Independent (Timestamp)
4	Temperature	Celsius	Independent (Weather)
5	Humidity	%	Independent (Weather)
6	Wind Speed	m/s	Independent (Weather)
7	Visibility	10 m	Independent (Weather)
8	Dew Point Temperature	Celsius	Independent (Weather)
9	Solar Radiation	MJ/m^2	Independent (Weather)
10	Rainfall	mm	Independent (Weather)
11	Snowfall	cm	Independent (Weather)
12	Seasons	Spring, Summer, Autumn, Winter	Independent (Timestamp)
13	Holiday	Holiday, No Holiday (Workday)	Independent (Timestamp)
14	Functioning Day	Yes (Functional Hours), No (Nonfunctional Hours)	Independent (Timestamp)

The seasons, holidays, and functioning days in the dataset are composed of character variables. We converted these variables into numeric variables: seasons (1: spring, 2: summer, 3: autumn, and 4: winter), holidays (1: workdays and 0: national holidays), and functioning days (1: functional hours, 0: nonfunctional hours). The number of rental bicycles rented per hour is a dependent variable from the dataset.

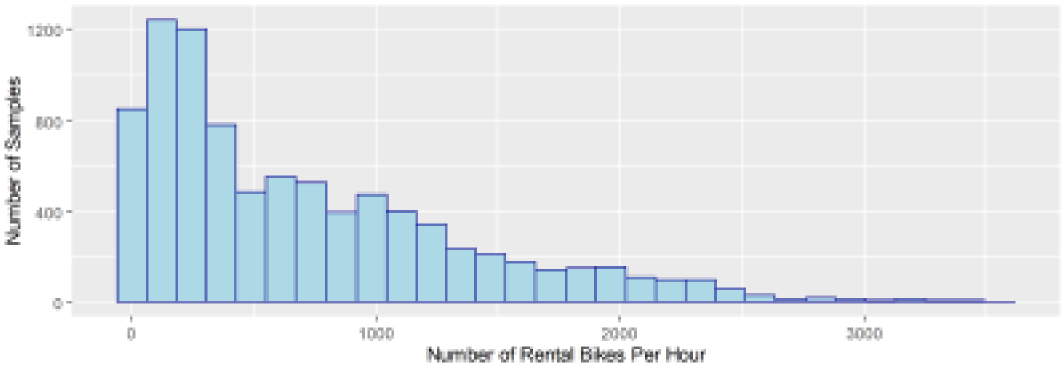


Figure 1. Histogram for Seoul bike-sharing demand.

Figure 1 illustrates a histogram of the bike-sharing demand for Seoul Bike. The histogram plot depicts the number of rental bikes per hour, and one bar covers a range of about 120. Data samples with a rental volume of fewer than 500 units per hour are concentrated, and the rental volume samples of 500 units or more per hour gradually decrease.

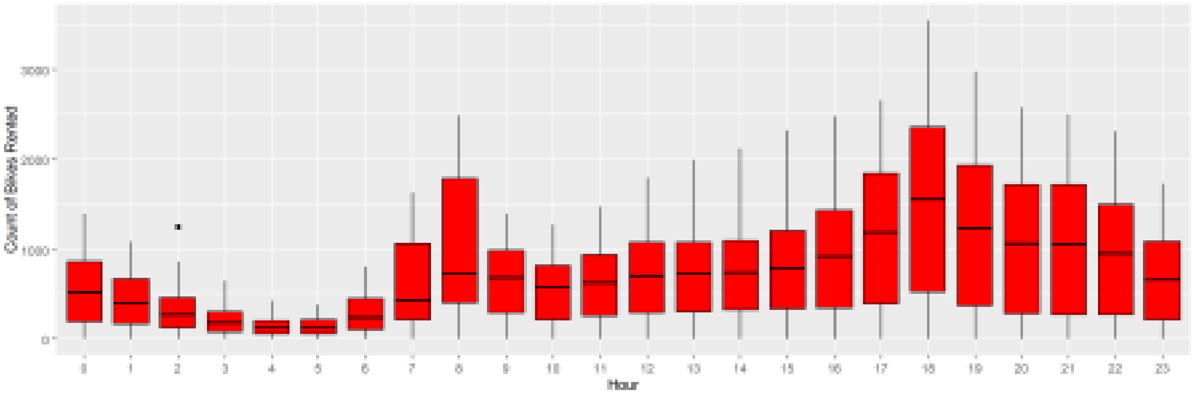
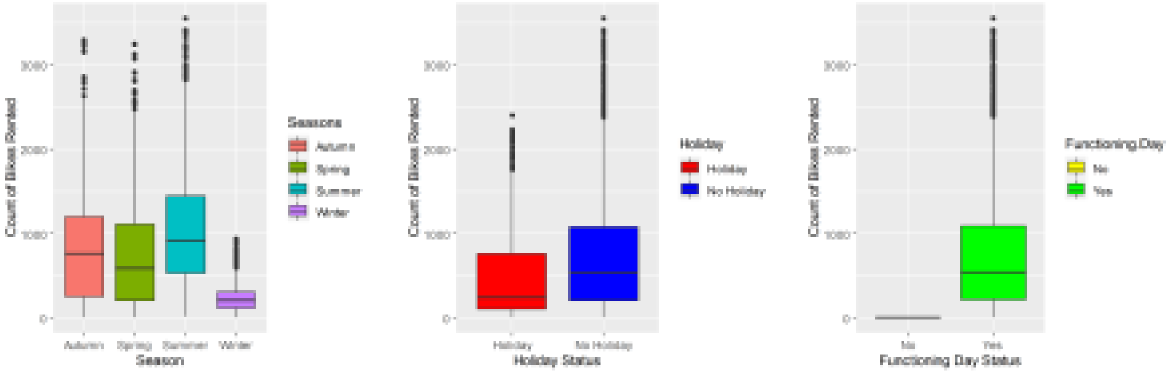


Figure 2. Boxplot for hourly Seoul bike-sharing demand.



(a) Season (b) Holiday status (c) Function day status

Figure 3. Boxplots for Seoul bike-sharing demand by (a) season, (b) holiday status, and (c) function day status.

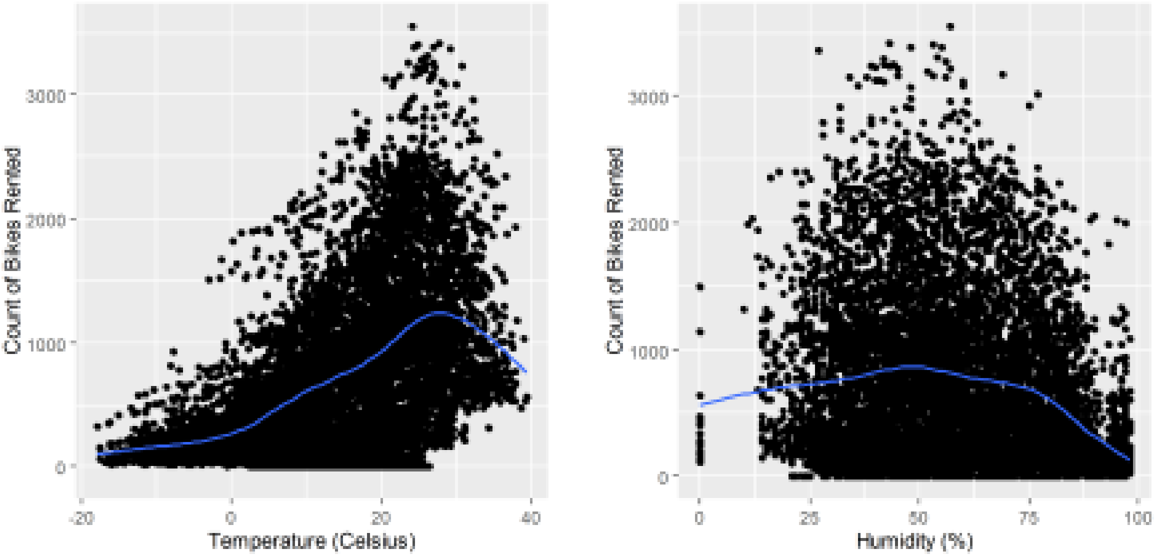
Figures 2 and 3 provide boxplots for bike-sharing demand based on timestamp information. The role of the median value at the black line is depicted in the boxplots. Figure 2 displays a boxplot of the rental bike count by hour, and Figure 3 provides boxplots of the rental bike count by season, holiday status, and function day status.

In Figure 2, the distribution of the boxplots is large at 8 a.m. and 6 p.m., suggesting that individuals choose to ride bicycles during commute hours rather than use public transportation, such as buses or vehicles. In addition, except for commute hours, the boxplots from 7 p.m. to midnight exhibit a larger distribution than the other time zones. It can be deduced from this that people ride their bikes after work as a hobby. This figure also indicates that an event occurred at 2 a.m. on a specific day because of one outlier.

In Figure 3(a), it was confirmed that, compared to other seasons, winter has a very modest distribution of boxplots. This figure depicts the influence of weather on the utilization of a rented bike, corroborating that people are less likely to ride their bikes when the weather becomes cooler. In Figure 3(b), the median value and distribution of working days are higher than those of holidays. This figure indicates that people use their bikes frequently on public transport during commute hours. This figure supports the reasoning and the high boxplot distribution of commute hours in Figure 2. Because bicycles cannot be rented on days when the system is down, all values in Figure 3(c) are zero; hence, the boxplot distribution is not shown. However, on functioning days, the median is depicted by a thick black line inside the green rectangle and has a value of 542. The lower whisker has a value of 214, and the top whisker has a value of 1084. Above the median number, there are more dispersed data. Many outliers marked with circles occur over the top whisker.

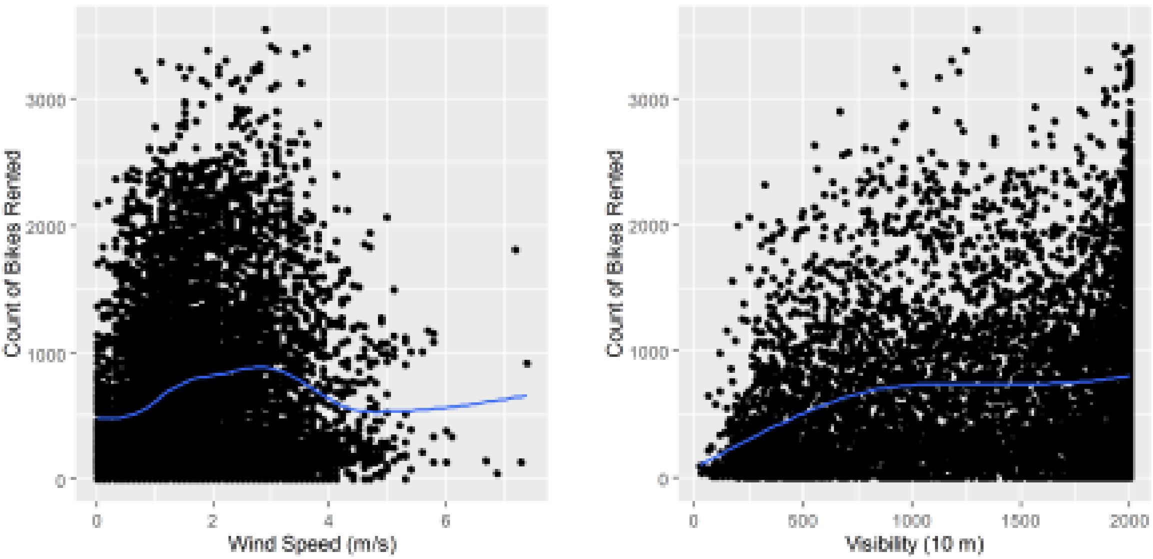
Figures 4 and 5 provide scatter plots of bike-sharing demand and weather conditions in Seoul. Figure 4 presents scatter plots of the hourly rental bike count by temperature, humidity, wind speed, and visibility, whereas Figure 5 depicts scatter plots of the hourly rental bike count and the dew point temperature, solar radiation, rainfall, and snowfall.

Figure 4(a) exhibits the moderate favorable correlation between the rental bike count and temperature. The demand for rental bikes increases as the weather warms. As the temperature rises, so does the utilization of rental bikes, but this reduces after 30°C. Thus, cycling is not recommended in extremely hot conditions. Figure 4(b) depicts a weak negative relationship between the rental bike count and humidity. This outcome may attest that adequate humidity is conducive to riding. Figures 4(c) and 4(d) depict the weak positive associations between the rental bike count and wind speed and visibility, respectively. High visibility indicates that the weather is pleasant, implying that people are frequently riding, whereas high wind speed indicates that people are not biking because it is difficult to ride in the wind.



(a) Temperature

(b) Humidity

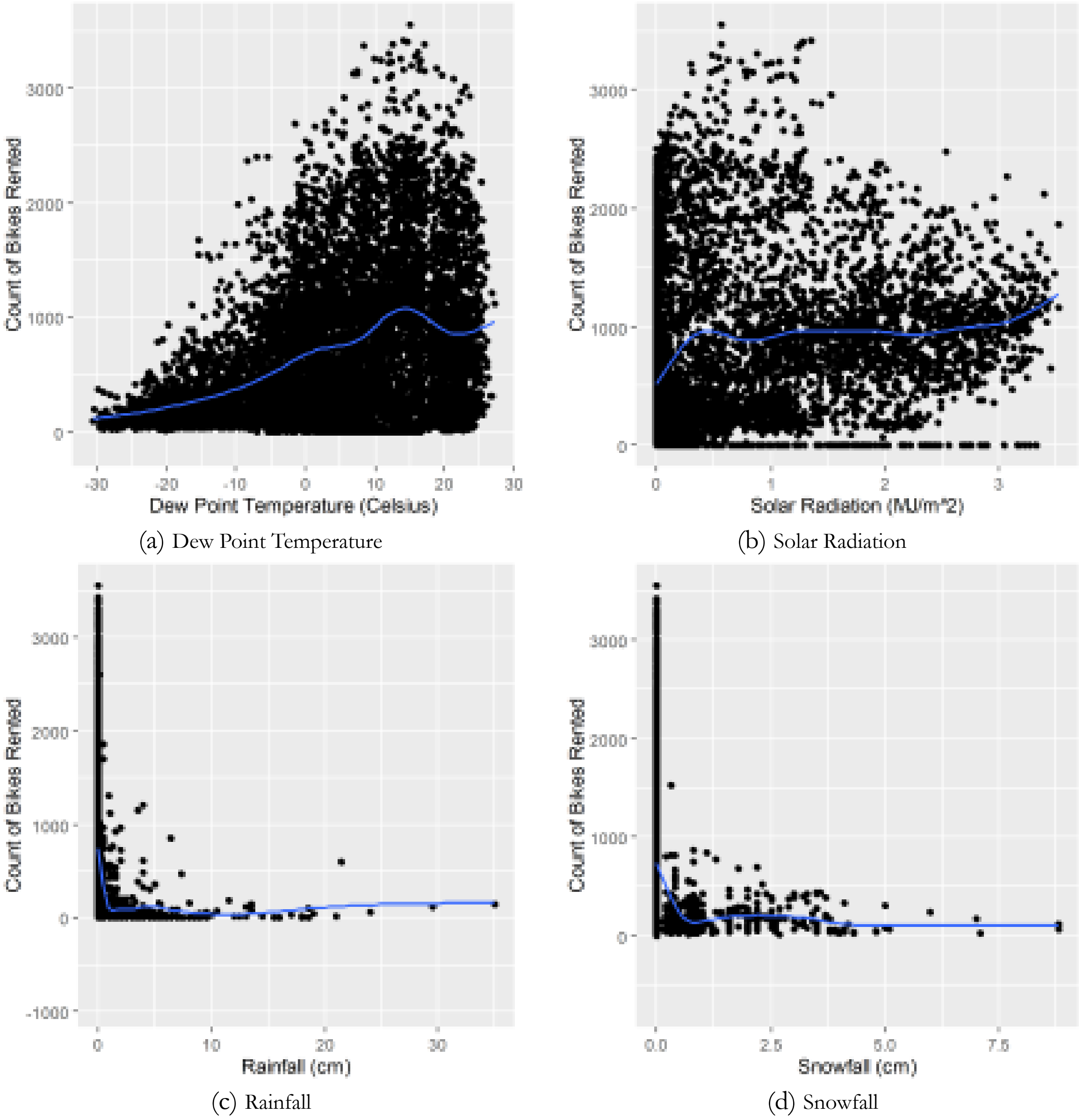


(c) Wind Speed

(d) Visibility

Figure 4. Scatter plots of bike-sharing demand in Seoul for (a) temperature, (b) humidity, (c) wind speed, and (d) visibility.

In Figure 5, weak positive correlations exist between the rental bike count and dew point temperature and solar radiation. In contrast, weak negative associations exist between the rental bike count and rainfall and snowfall, indicating that these elements negatively influence rental bike utilization. When the rainfall and snowfall values are both zero, the use of bicycles is extremely high. In other words, as rainfall and snowfall levels rise, bicycle usage does not. High correlations exist between the rental bike count and timestamp and weather conditions via EDA.



(a) Dew Point Temperature

(b) Solar Radiation

(c) Rainfall

(d) Snowfall

Figure 5. Scatter plots of Seoul bike-sharing demand and (a) dew point temperature, (b) solar radiation, (c) rainfall, and (d) snowfall.

4. Model Construction

This section introduces the construction of a novel time-series forecasting model for multistep-ahead bike-sharing demand prediction in Seoul. The comprehensive architecture of the proposed modeling is presented in Figure 6. As the goal is to obtain strong prediction performance from a limited dataset, we partitioned the data into an insufficient training and sufficient testing sets rather than a sufficient training set and insufficient testing set. Hence, we split the dataset into a training set and testing set in a 25:75 ratio. The training set period is three winter months (December 2017 to February 2018).

The testing set period is nine months, from March 2018 to November 2018, during the spring, summer, and autumn seasons. The training set was used to construct the RF, XGBoost, and Cubist models on the R environment, a programming language popular among data scientists [29].

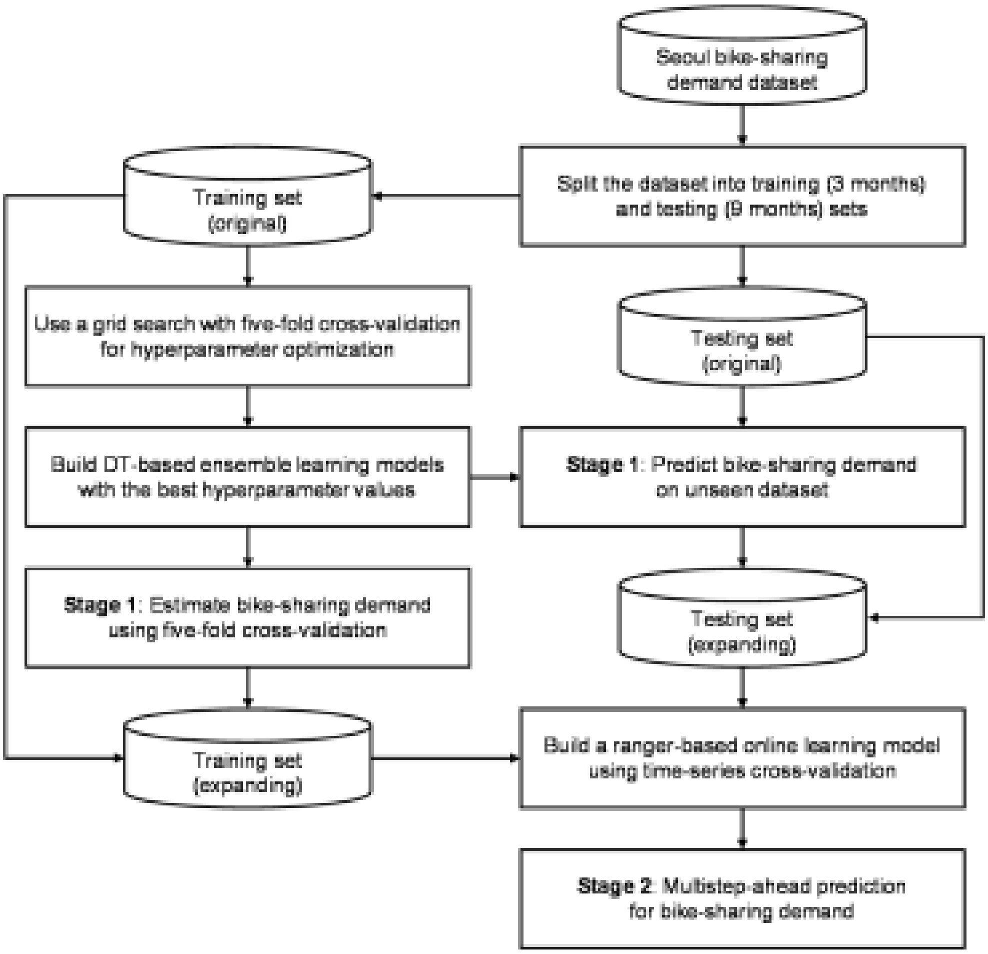

Seoul bike-sharing demand dataset
Split the dataset into training (3 months) and testing (9 months) sets
Training set (original)
Use a grid search with five-fold cross-validation for hyperparameter optimization
Testing set (original)
Build DT-based ensemble learning models with the best hyperparameter values
Stage 1: Predict bike-sharing demand on unseen dataset
Stage 1: Estimate bike-sharing demand using five-fold cross-validation
Testing set (expanding)
Training set (expanding)
Build a ranger-based online learning model using time-series cross-validation
Stage 2: Multistep-ahead prediction for bike-sharing demand

Figure 6. Overall architecture of the proposed model construction.

The implementation of randomForest [34] in R is commonly used due to its wide range of features. However, it has not been optimized for using data with many dimensions. Furthermore, the most significant flaw is the slowness with which the model is implemented. The ranger (short for RANdom forest GEneRator) [24], a creative software package, was designed to address these flaws and make it suitable for high-dimensional data. First, efficient memory management was often executed by minimizing duplicate data, keeping node information in straightforward data forms, and clearing the memory as soon as possible. The ranger package employs two distinct splitting techniques [24,26]. The first technique retrieves the feature values by index after presorting them, and the second technique splits the raw values and retrieves and arranges them. The first and second techniques are employed in the optimized runtime node on major and minor nodes, respectively.

We employed the caret (short for Classification And REgression Training) [35] package for XGBoost and Cubist modeling, which can simply implement them using the xgboost [36] and Cubist [33] packages. The ideal hyperparameters for the DT-based EL models must be determined to improve prediction performance. Several studies have been conducted to develop an appropriate RF process. For instance, the best mTry values for regression ($x/3$) and classification ($sqrt(x)$) [29], where x is the number of independent features, are well known. Furthermore, in testing with 29 datasets [28], no significant improvement in nTree occurred after 128 trees. For the RF model construction, we set nTree and mTry to 128 and 4, respectively. In addition, we performed a grid search in the training set to determine the best hyperparameters for the XGBoost and Cubist models using five-fold cross-validation. All DT-based EL models had the random seed set to 1234. Table 2 lists the hyperparameters for each model, the R library used, and the sources that provide information on the ideal values or hyperparameter ranges. The values in bold font denote the final values used for the optimal model construction.

Table 2. List of hyperparameters for optimal DT-based ensemble learning model construction.

Models	Libraries	Hyperparameters
RF	ranger	*mTry*: **4** [29]
		nTree: **128** [28]
XGBoost	xgboost,	*nround*: 10, 20, 30, 40, 50, 60, 70, 80, 90, **100** [37]
	caret	*max_depth*: 2, 3, 4, **5**, 6, 7, 8 [37]
		eta: 0.1, **0.2**, 0.3 [37]
		gamma: 0.1, 1, 10, 100, **1000** [37]
		colsample_bytree: 0.2, 0.4, 0.6, 0.8, **1** [37]
		min_child_weight: **1** [37]
		subsample: **1** [37]
Cubist	cubist,	*committees*: 1, 10, 50, **100** [33]
	caret	*neighbors*: 0, 1, **5**, 9 [33]

The main idea behind stacking EL is to construct heterogeneous weak learners in parallel and then aggregate their prediction values for the final prediction using a meta-model [38]. However, since most heterogeneous weak learners hold based on ML approaches with high data dependency, the stacking EL technique may have restrictions in enhancing the prediction performance of bike-sharing demand [26]. Put more simply; ML approaches may exhibit inadequate prediction accuracy on unseen (out-of-sample) datasets with different attributes from the training (in-sample) dataset [18]. For instance, Figure 3 reveals that the rented bicycle count is substantially lower in the winter than in other seasons. If the rental bicycle count of the testing set exhibits substantially higher than that of the training set, the ML approach built on the training set may not accurately echo the rising trend. Furthermore, when the dataset is insufficient [21], obtaining satisfactory prediction performance with scarce training data is difficult.

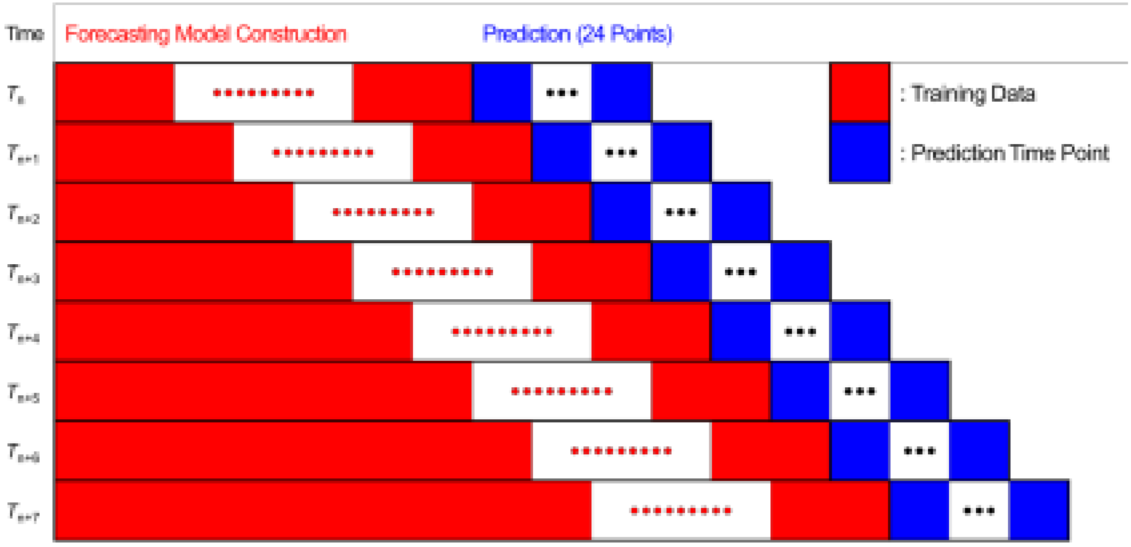


Figure 7. Time-series cross-validation for multistep-ahead prediction.

We used R ranger to create a TSCV-based online learning model to address these issues. The TSCV [39] focuses on many prediction horizons for each testing set. We employed various additional training sets depending on the schedule period, each including one observation not used in the preceding training set. We carried out the testing sets from one hour following the current period to one day to perform multistep-ahead bike-sharing demand prediction, as illustrated in Figure 7. The prediction accuracy was computed at each time point, and the results were averaged to test the forecasting model performance. As a meta-model, the ranger package takes less time to train and needs more nominal fine-tuning of hyperparameters than other well-known ML approaches, such as deep learning or boosting methods [24,26]. We employed 15 input variables for the ranger model training, including predicted values from the RF, XGBoost, and Cubist models, and weather/timestamp information, which is known to generate fluctuations in the current trends and patterns for bike-sharing demand.

To generate the prediction values of RF, XGBoost, and Cubist, which are input variables required for training the ranger model during both a training set and testing set, we first generated the prediction values of the RF, XGBoost, and Cubist models for the training set through five-fold cross-validation using the optimal hyperparameter values. Then, we constructed the RF, XGBoost, and Cubist models trained with optimal hyperparameters using the training set and performed bike-sharing demand prediction on the testing set.

The proposed model can predict bike-sharing demand more accurately than the online learning-based stacking EL method trained solely on the prediction values of the RF, XGBoost, and Cubist models. More additional data can be employed over time than in traditional time-series forecasting model evaluation. Thus, the data shortage problem can be solved, and RF can more effectively understand recent patterns and trends in bike-sharing demand from external elements. The RF can also expect good prediction results because it successfully learns the nonlinear relationship between bike-sharing demand and external elements to alter the weights of the input features in the forecasting model to match recent bike-sharing demand patterns.

5. Results and Discussion

This section first offers measures for comparing the forecasting model prediction performance. Then, we review the experimental findings in detail to demonstrate the validity of this experiment. We finally demonstrate the interpretability of the RF, XGBoost, and Cubist models on the training set.

5.1 Performance Indicators

The prediction performance of the regression models is considered using a variety of criteria; namely, the root mean square error (RMSE), mean absolute error (MAE), R-squared (R^2), and coefficient of variation of the root mean square error (CVRMSE). The RMSE is a scale-dependent indicator that produces values with the same measurement units by calculating the contrast between the actual and prediction values of the residuals. This method detects severe errors and determines the volatility of the model response in terms of variance. The MAE indicator is employed to determine prediction accuracy. Like the RMSE method, it is a scale-dependent indicator that virtually captures prediction error levels by avoiding the offset between negative and positive errors. The R^2 is the determination coefficient, which normally runs from 0 to 1, signifying the superiority of fit. A high value indicates that the anticipated values perfectly correspond to the actual values. The CVRMSE is employed to determine the relative variability measure. The CVRMSE specifies the variation of the widespread prediction error regarding the target mean. A high CVRMSE score indicates a high number of errors in the regression model.

Equations (3) to (6) are used to determine the RMSE, MAE, R^2, and CVRMSE, respectively, where n denotes the number of prediction time points and t denotes the prediction time point. Moreover, A_t and F_t signify the actual and forecasted values at time t, respectively, whereas $\bar{A}$ denotes the average of all actual values.

$$\mathrm{RMSE} = \sqrt{\frac{\sum_{t=1}^{n} (A_t - F_t)^2}{n}}, \tag{3}$$

$$\mathrm{MAE}=\frac{1}{n}\sum_{t=1}^{n}\left|A_t-F_t\right|, \quad (4)$$

$$R^2=1-\frac{\sum_{t=1}^{n}\left(A_t-F_t\right)^2}{\sum_{t=1}^{n}\left(A_t-\bar{A}\right)^2}, \quad (5)$$

$$\mathrm{CVRMSE}=\frac{100}{\bar{A}}\sqrt{\frac{\sum_{t=1}^{n}\left(A_t-F_t\right)^2}{n}}. \quad (6)$$

5.2 Experimental Design

We conducted many trials with an Intel Core i7-8565U CPU @ 1.80 GHz with 16 GB DDR4 RAM for multistep-ahead bike-sharing demand prediction. In RStudio v. 1.3.1073 with R v. 4.1.2, we performed an EDA and developed a tree-based EL model. The XGBoost and Cubist models were executed with seven threads to perform parallel processing. The run times of the DT-based EL models with optimal hyperparameter values for the training set are presented in Figure 8.

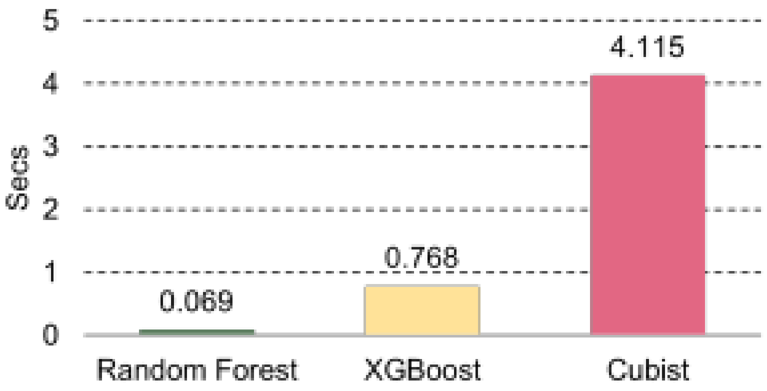


Figure 8. Run times for each model.

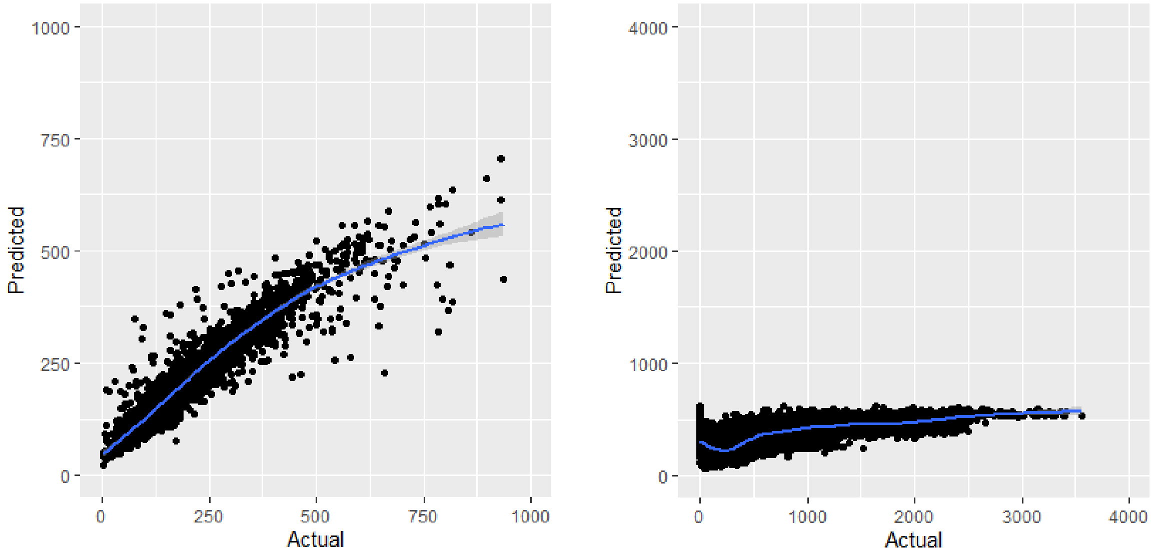
In Figure 8, XGBoost outperformed Cubist in terms of training time. The training times for RF without parallel processing and for Cubist were the quickest and slowest, respectively. Therefore, RF implemented using the ranger package is suitable for online learning even in a computing environment with limited performance. We used five-fold cross-validation using the optimal hyperparameter values to generate the RF, XGBoost, and Cubist prediction values for the training set. Then, we performed bike-sharing demand prediction on the testing set using the RF, XGBoost, and Cubist models trained with optimal hyperparameters to generate input variables.

Table 3. Performance comparison of DT-based ensemble learning models.

Models	**Training Set**				**Testing Set**			
	RMSE	MAE	R^2	CVRMSE	RMSE	MAE	R^2	CVRMSE

RF	61.09	38.33	0.862	27.08	770.96	564.44	0.454	89.50
XGBoost	50.55	29.83	0.890	22.41	698.51	513.68	0.479	81.09
Cubist	60.31	37.96	0.844	26.74	610.44	455.13	0.437	70.87

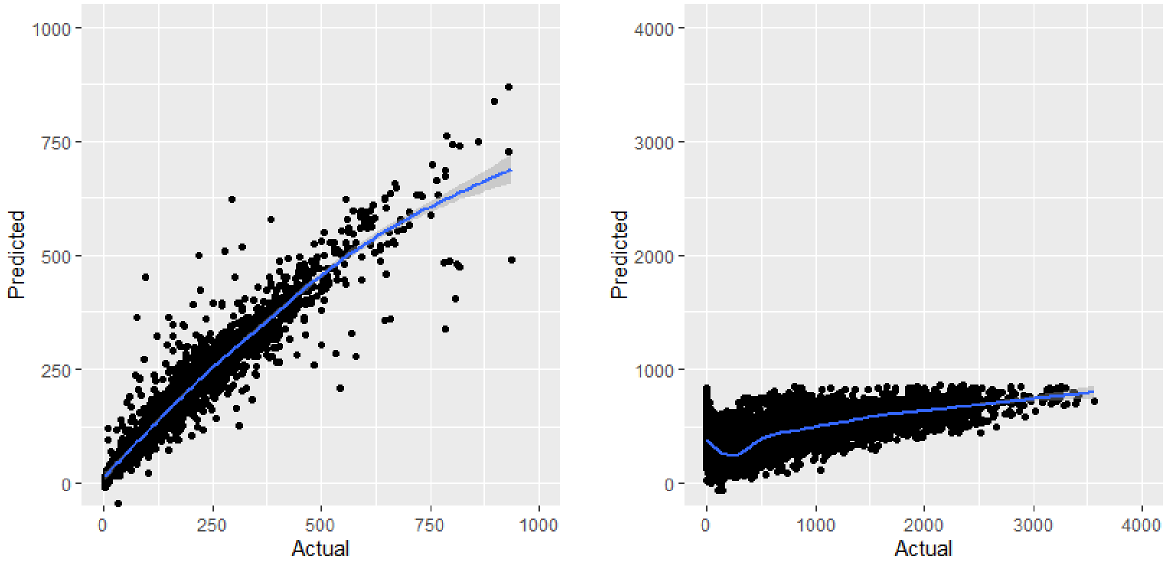
Table 3 lists the performance comparison of the DT-based EL model on the training and testing sets, and Figures 9 to 11 illustrate the actual and predicted values for each model through a scatter plot for the training and testing sets. In Table 3, the DT-based EL models achieved good performance in the training set but poor performance in the testing set (unseen) with different characteristics from the training set. In addition, the XGBoost and Cubist models derive better performance than the other models on the training and testing sets, respectively. These figures explain that the prediction values in the testing set were derived within the range of actual values of the training set rather than those of the testing set.



(a) Training set

(b) Testing set

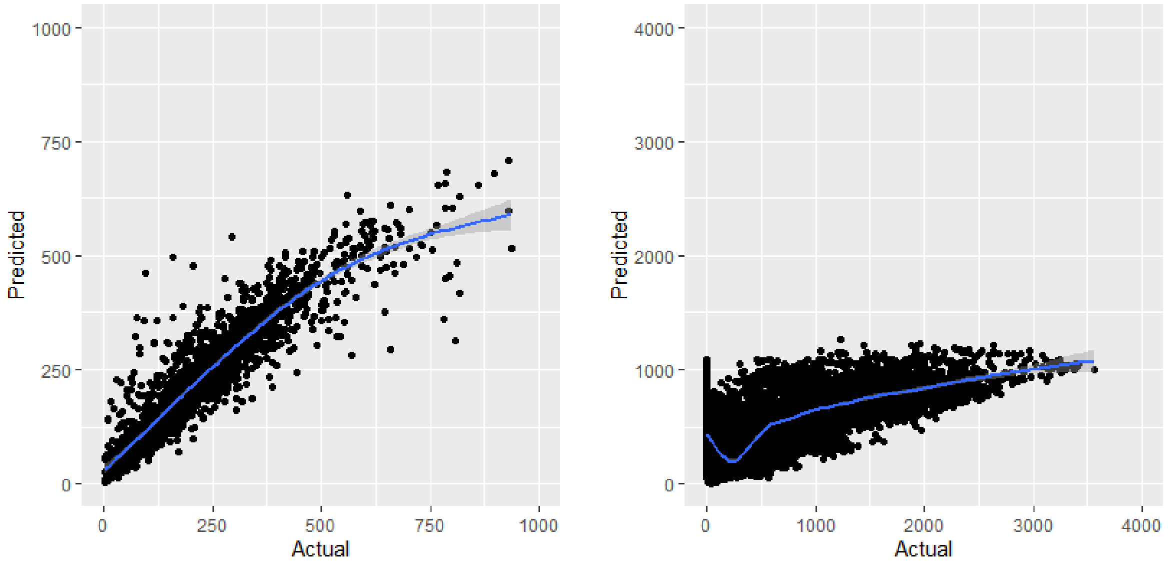
Figure 9. Scatter plots of the RF model on (a) training (in-sample) and (b) testing (unseen) sets.



(a) Training set

(b) Testing set

Figure 10. Scatter plots of the XGBoost model on the (a) training (in-sample) and (b) testing (unseen) sets.



(a) Training set

(b) Testing set

Figure 11. Scatter plots of the Cubist model on the (a) training (in-sample) and (b) testing (unseen) sets.

Interpretability approaches have received substantial attention to ensure model reliability [41]. In the training set that comprised the winter season of the DT-based EL models, we examined feature importance to establish the most influential factors and used a partial dependence plot (PDP) to represent the change in projected values based on the change in their factors, as depicted in Figures 12 and 13. As presented in Figure 12, because RF and XGBoost/Cubist employ different packages to calculate the feature importance, they display a varied range of importance values. Hour was the most influential variable, whereas temperature was the most influential weather condition and the second most influential variable for model construction. Next, for RF, the third significant variable was considered solar radiation, and for XGBoost and Cubist, the third significant variable was the dew point temperature.

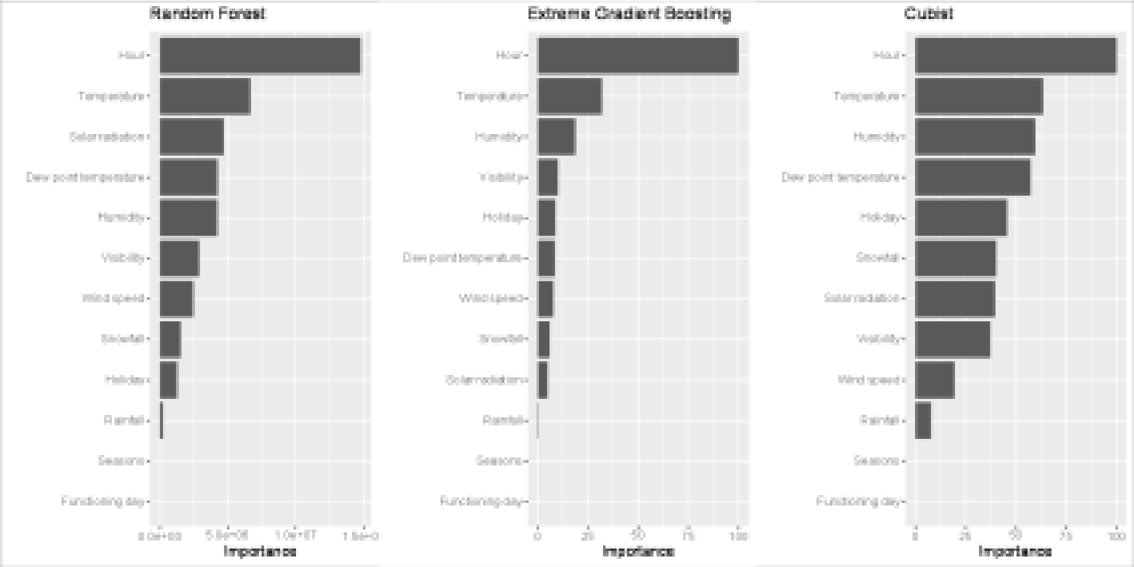


Figure 12. Feature importance for each DT-based ensemble model.

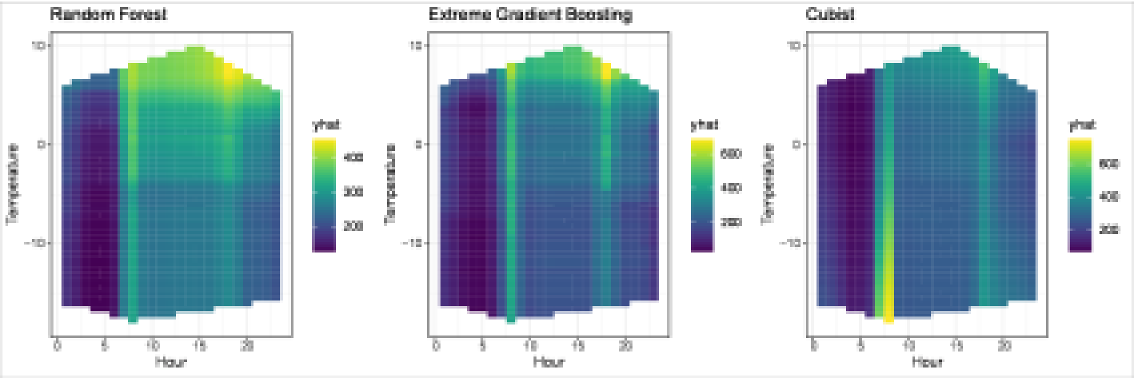


Figure 13. Partial dependence plots for each DT-based ensemble learning model.

We used multipredictor PDPs to express the change in prediction values when the significant factors changed. In Figure 13, we used hour and temperature as the most important timestamp and weather elements to display the PDPs. The prediction value is indicated by "*yhat*" in this figure. In winter, the bike-sharing demand in Seoul is high during commuting hours, and a higher temperature results in higher bike-sharing demand. The RF model had more down prediction values than the XGBoost and Cubist models since the RF model was trained with the input variables distributed uniformly.

5.3 Experimental Results

To explain the validity of the proposed model in terms of multistep-ahead time-series forecasting, we developed four multistep-ahead bike-sharing demand prediction models based on the multivariate random forest (MRF), ranger-based online learning approach (RABOLA), TSCV-based online learning model trained with only external elements (TOM 1), and TSCV-based online learning model trained with the prediction values of RF, XGBoost, and Cubist (TOM 2).

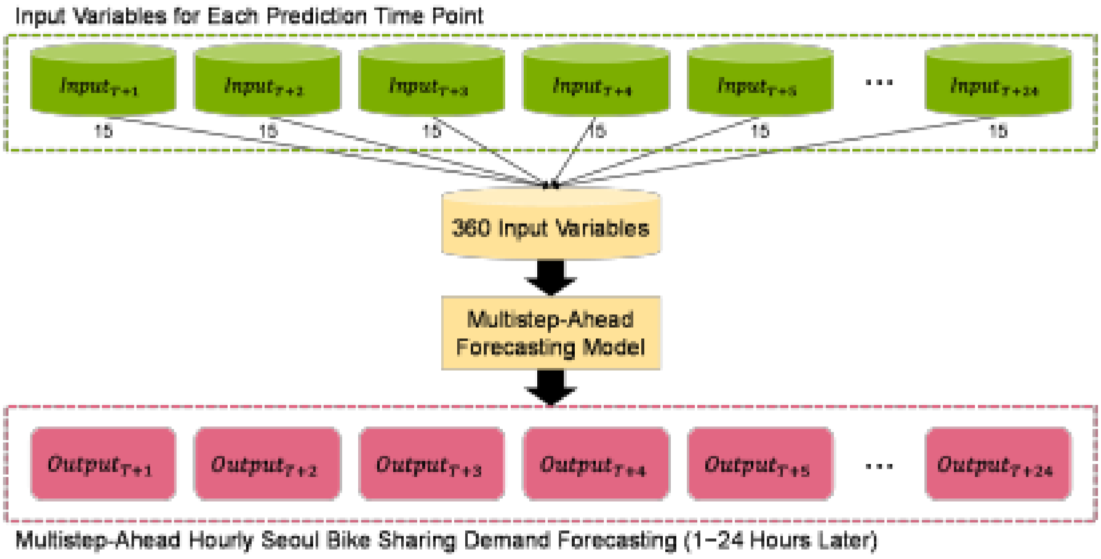


Figure 14. Conceptualization of the MIMO strategy for MRF.

As demonstrated in Figure 14, the MRF [40] uses a multiple-input and multiple-output (MIMO) strategy [19,20] to forecast multiple dependent variables in a single RF through a linear connection between the dependent variables. The MRF models were built with 360 input variables (15 input variables x 24 prediction time points). The RABOLA [26] model is a ranger-based time-series forecasting procedure with a seven-day sliding window that employs the prediction values of DT-based EL models and external elements (i.e., weather and timestamp information) as independent variables on the testing set. We used the MRF and RABOLA models as the baseline model in these experiments due to their excellent prediction performance in multistep-ahead time-series forecasting.

Table 4. Performance comparison of multistep-ahead prediction.

Models	RMSE (SD)	MAE (SD)	R^2 (SD)	CVRMSE (SD)
MRF	801.48 (16.28)	598.08 (13.87)	0.394 (0.027)	92.90 (1.86)
RABOLA	352.29 (15.60)	241.92 (10.96)	0.723 (0.023)	40.83 (1.79)
TOM 1	316.29 (6.25)	219.14 (4.96)	0.777 (0.009)	36.66 (0.71)
TOM 2	430.58 (1.31)	308.97 (1.22)	0.595 (0.002)	49.91 (0.13)
Ours	310.85 (4.81)	213.32 (3.56)	0.783 (0.007)	36.03 (0.54)

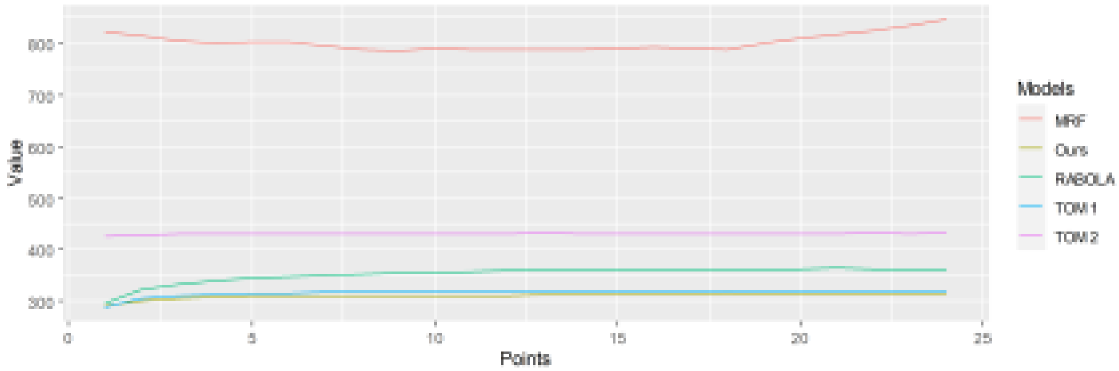


Figure 15. RMSE comparison of multistep-ahead prediction.

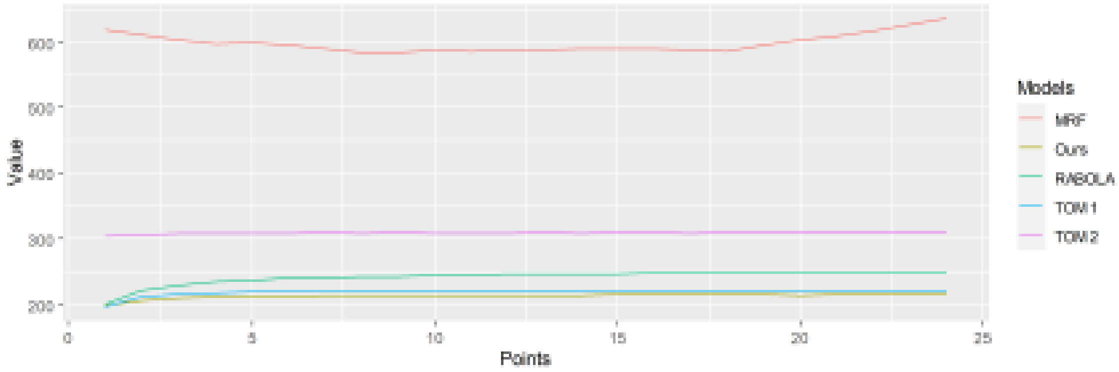


Figure 16. MAE comparison of multistep-ahead prediction.

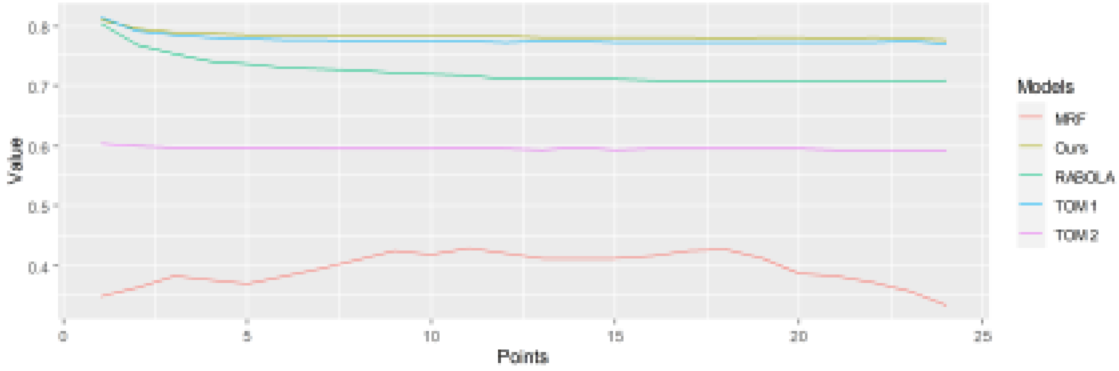


Figure 17. R-squared comparison of multistep-ahead prediction.

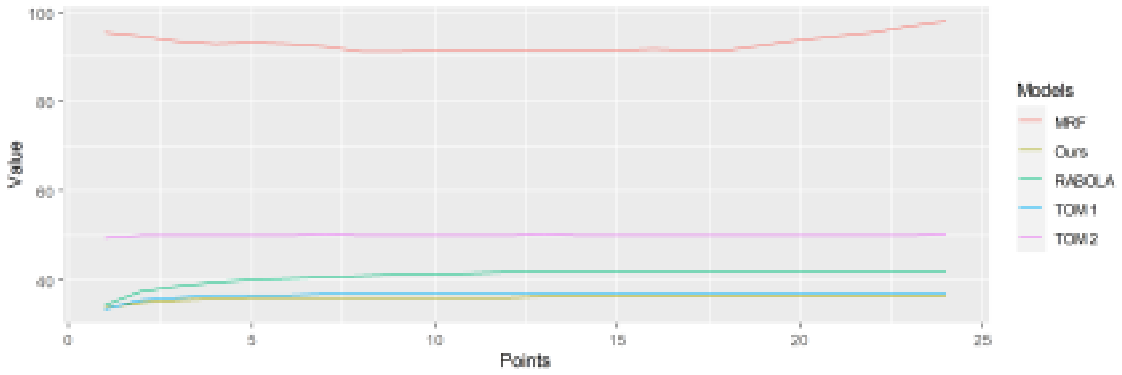


Figure 18. CVRMSE comparison of multistep-ahead prediction.

Figures 15 to 18 present the RMSE, MAE, R^2, and CVRMSE values, respectively, by step for the hourly bike-sharing demand. Table 4 lists the average and standard deviation (SD). The proposed model achieved the best prediction performances in the average RMSE, MAE, R^2, and CVRMSE. From a step-by-step perspective, most online learning models demonstrated progressively larger RMSE, MAE, R^2, and CVRMSE values with increased disparity between the present and prediction time points. The proposed model produced consistent and superior prediction performance throughout all steps.

To compare the proposed model performance in a single forecast, we implemented several existing demand prediction models for Seoul Bike as baseline models, including Cubist, regularized RF (RRF), classification and regression tree (CART), k-nearest neighbor (KNN), conditional inference tree (CTree), multiple linear regression (MLR), GBM, support vector machine (SVM) with a radial basis function kernel, boosted tree, and XGBoost [15,16]. We implemented the prediction models because the implementation environment these models considered (e.g., input variable, model construction, and computer language) could be used in the same way in this study. Table 5 displays the information on existing bike-sharing demand prediction models for Seoul Bike.

Table 5. Information on existing bike-sharing demand prediction models for Seoul Bike.

Models	Reference	Hyperparameters
Cubist	[15]	committees: 41 neighbors: 3
RRF	[15]	mtry (number of randomly selected predictors): 14 coefReg (regularization value): 0.505
CART	[15]	cp (complexity parameter): 0.0001
KNN	[15]	k (number of neighbors): 3
CTree	[15]	maxdepth (max tree depth): 19 mincriterion (1 − p-value threshold): 0.01
MLR	[16]	automatic identification
GBM	[16]	n.trees (number of boosting iterations): 301 interaction.depth (max tree depth): 12
SVM	[16]	sigma: 0.1 C (cost): 40
Boosted tree	[16]	mstop (number of trees): 41 maxdepth (max tree depth): 10
XGBoost	[16]	nrounds: 1100 max_depth 3

Sathishkumar and Cho [15,16] configured 26 input variables, which differ from the experimental input variables we employed in that they added timestamp variables that can represent weekdays/weekends and the days of the week. Then, they generated binary variables representing each characteristic by performing one-hot encoding on all variables except for the hour variable (i.e., seasons, holiday indicators, functioning day indicators, weekday indicators, and days of the week).

Table 6. Performance comparison between the proposed and existing models.

Models	RMSE	MAE	R^2	CVRMSE
Cubist [15]	570.36	423.37	0.483	66.21
RRF [15]	752.15	554.10	0.502	87.32
CART [15]	786.52	573.25	0.305	91.31
KNN [15]	839.17	622.53	0.138	97.42
Ctree [15]	763.56	570.71	0.108	88.64
MLR [16]	711.04	521.67	0.304	82.55
GBM [16]	718.27	521.18	0.472	83.39
SVM [16]	717.91	522.36	0.483	83.34
Boosted tree [16]	760.74	561.11	0.372	88.32
XGBoost [16]	683.56	500.19	0.498	79.36
Ours	315.00	216.44	0.777	36.47

We applied the 24-step-ahead (day-ahead) forecasted values of the proposed model and measured the prediction performance. Table 6 displays the prediction performance of various ML models over the entire period of the testing set. As listed in the table, the proposed scheme produced the best prediction performance for the RMSE, MAE, R^2, and CVRMSE. We established that applying the online learning method can deliver a steadier performance even if the term between the training set and testing set expansions.

6 Conclusions

This paper presents a two-stage online learning approach for bike-sharing demand prediction in Seoul, South Korea. We collected the Seoul bike-sharing demand dataset from the UCI Machine Learning Repository website and performed EDA for the dataset. We converted character variables, such as seasons, holidays, and functioning days, into numeric variables for model construction. We split the datasets into insufficient training (3 months) and sufficient testing (9 months) sets and adopted DT-based EL methods, including RF, XGBoost, and Cubist. We used a grid search through five-fold cross-validation to select optimal hyperparameter values for these methods in the training set. In the first stage, we generated the prediction values of the RF, XGBoost, and Cubist models in both the training and testing sets. In the second stage, we used the ranger package, a fast implementation of RF, to construct the TSCV-based online learning model for multistep-ahead (one hour to one day later) bike-sharing demand prediction. The proposed model was trained with external elements, such as weather and timestamp information, and the prediction values of the RF, XGBoost, and Cubist models. In multistep-ahead prediction, the proposed model performed admirably. Thus, the online learning method can deliver steadier prediction performance even as the term between training and testing sets increases.

We confirmed a significant number of bicycle rentals during commuting time, and the models trained with the training set during the winter had a significant influence during commuting time. Variable importance and PDP were analyzed to identify the hidden links between the external components. Hour and temperature were ranked as the most significant variables in predicting rental bicycle demand at each hour throughout the winter period for all DT-based EL models. In addition, this study revealed correlations between variables based on EDA.

The proposed method forecasted bicycle rental demand for the entire Seoul area to aid in developing urban policies that provide various options for better health, a cleaner environment, and climate action. Future research will concentrate on projecting rental bicycle demand by region while accounting for seasonal fluctuations. Forecasts based on regional rental bicycle demand will be significantly more helpful in delivering public rental bicycles on a constant basis, which we believe will play an essential role in the physical, social, and mental well-being of the population.

Declarations

Ethical Approval: Not applicable.

Competing Interests: The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Authors' Contributions: Subeen Leem: Conceptualization, Methodology, Writing – Original Draft. Jisong Oh: Data Curation, Visualization, Validation. Jihoon Moon: Software, Formal analysis, Writing – Review & Editing. Mucheol Kim: Project administration, Funding acquisition. Seungmin Rho: Investigation, Resources, Supervision.

Funding: This research was supported by a grant (2021-MOIS37-004) of Intelligent Technology Development Program on Disaster Response and Emergency Management funded by the Ministry of Interior and Safety (MOIS, Korea) and the BK21 FOUR program (Education and Research Center for Securing Cyber-Physical Space) through the National Research Foundation (NRF) funded by the Ministry of Education of Korea (5199990314137). This research was also supported by the Soonchunhyang University Research Fund.

Availability of Data and Materials: The data that support the findings of this study are openly available in UCI Machine Learning Repository: Seoul Bike Sharing Demand Data Set at https://archive.ics.uci.edu/ml/datasets/Seoul+Bike+Sharing+Demand. These data were derived from the following resources available in the public domain: https://data.seoul.go.kr/index.do (Seoul bike sharing demand), https://data.kma.go.kr/resources/html/en/aowdp.html (weather data), and https://publicholidays.co.kr/ (holiday information).

References

[1] Zhang, L., Zhang, J., Duan, Z. Y., & Bryde, D. (2015). Sustainable bike-sharing systems: characteristics and commonalities across cases in urban China. *Journal of Cleaner Production*, 97, 124–133. https://doi.org/10.1016/j.jclepro.2014.04.006.

[2] Shaheen, S. A., Guzman, S., & Zhang, H. (2010). Bikesharing in Europe, the Americas, and Asia: Past, Present, and Future. *Transportation Research Record*, 2143(1), 159–167. https://doi.org/10.3141/2143-20.

[3] Buck, D., Buehler, R., Happ, P., Rawls, B., Chung, P., & Borecki, N. (2013). Are Bikeshare Users Different from Regular Cyclists?: A First Look at Short-Term Users, Annual Members, and Area Cyclists in the Washington, D.C., Region. *Transportation Research Record*, 2387(1), 112–119. https://doi.org/10.3141/2387-13.

[4] Raviv, T., & Kolka, O. (2013). Optimal inventory management of a bike-sharing station. *IIE Transactions*, 45(10), 1077–1093. https://doi.org/10.1080/0740817X.2013.770186.

[5] Macioszek, E., Świerk, P., & Kurek, A. (2020). The Bike-Sharing System as an Element of Enhancing Sustainable Mobility—A Case Study based on a City in Poland. *Sustainability*, 12(8), Article 3285. https://doi.org/10.3390/su12083285.

[6] Song, J., Zhang, L., Qin, Z., & Ramli, M. A. (2021). A spatiotemporal dynamic analyses approach for dockless bike-share system. *Computers, Environment and Urban Systems*, 85, Article 101566. https://doi.org/10.1016/j.compenvurbsys.2020.101566.

[7] Yang, Y., Heppenstall, A., Turner, A., & Comber, A. (2020). Using graph structural information about flows to enhance short-term demand prediction in bike-sharing systems. *Computers, Environment and Urban Systems*, 83, Article 101521. https://doi.org/10.1016/j.compenvurbsys.2020.101521.

[8] Sun, S., & Ertz, M. (2021). Contribution of bike-sharing to urban resource conservation: The case of free-floating bike-sharing. *Journal of Cleaner Production*, 280, Article 124416. https://doi.org/10.1016/j.jclepro.2020.124416.

[9] García-Palomares, J. C., Gutiérrez, J., & Latorre, M. (2012). Optimizing the location of stations in bike-sharing programs: A GIS approach. *Applied Geography*, 35(1–2), 235–246. https://doi.org/10.1016/j.apgeog.2012.07.002.

[10] Zhu, R., Zhang, X., Kondor, D., Santi, P., & Ratti, C. (2020). Understanding spatio-temporal heterogeneity of bike-sharing and scooter-sharing mobility. *Computers, Environment and Urban Systems*, 81, Article 101483. https://doi.org/10.1016/j.compenvurbsys.2020.101483.

[11] Hua, M., Chen, X., Zheng, S., Cheng, L., & Chen, J. (2020). Estimating the parking demand of free-floating bike sharing: A journey-data-based study of Nanjing, China. *Journal of Cleaner Production*, 244, Article 118764. https://doi.org/10.1016/j.jclepro.2019.118764.

[12] Yang, Y., Heppenstall, A., Turner, A., & Comber, A. (2019). A spatiotemporal and graph-based analysis of dockless bike sharing patterns to understand urban flows over the last mile. *Computers, Environment and Urban Systems*, 77, Article 101361. https://doi.org/10.1016/j.compenvurbsys.2019.101361.

[13] Kim, E. J., Kim, J., & Kim, H. (2020). Does Environmental Walkability Matter? The Role of Walkable Environment in Active Commuting. *International Journal of Environmental Research and Public Health*, 17(4), Article 1261. https://doi.org/10.3390/ijerph17041261.

[14] The Korea Bizwire. (2020). *Seoul City Introduces Sturdier Models of Public Bike.* Retrieved from http://koreabizwire.com/seoul-city-introduces-sturdier-models-of-public-bike/173869/. Accessed April 1, 2022.

[15] Sathishkumar, V. E., & Cho, Y. (2020). A rule-based model for Seoul Bike sharing demand prediction using weather data. *European Journal of Remote Sensing*, 53(sup1), 166–183. https://doi.org/10.1080/22797254.2020.1725789.

[16] Sathishkumar, V. E., Park, J., & Cho, Y. (2020). Using data mining techniques for bike sharing demand prediction in metropolitan city. *Computer Communications*, 153, 353–366. https://doi.org/10.1016/j.comcom.2020.02.007.

[17] [dataset] Sathishkumar, V. E., & Cho, Y. (2020). *Seoul Bike Sharing Demand Data Set.* UCI Machine Learning Repository. https://archive.ics.uci.edu/ml/datasets/Seoul+Bike+Sharing+Demand.

[18] Potgieter, P. H. (2020). Machine Learning and Forecasting: A Review. In: Alleman, J., Rappoport, P., & Hamoudia, M. (Eds.), *Applied Economics in the Digital Era* (pp. 193–207). Palgrave Macmillan. https://doi.org/10.1007/978-3-030-40601-1_8.

[19] Taieb, S. B., Bontempi, G., Atiya, A. F., & Sorjamaa, A. (2012). A review and comparison of strategies for multi-step ahead time series forecasting based on the NN5 forecasting competition. *Expert Systems with Applications*, 39(8), 7067–7083. https://doi.org/10.1016/j.eswa.2012.01.039.

[20] Yang, B. S., & Tan, A. C. C. (2009). Multi-step ahead direct prediction for the machine condition prognosis using regression trees and neuro-fuzzy systems. *Expert Systems with Applications*, 36(5), 9378–9387. https://doi.org/10.1016/j.eswa.2009.01.007.

[21] Schmidt, J., Marques, M. R., Botti, S., & Marques, M. A. (2019). Recent advances and applications of machine learning in solid-state materials science. *npj Computational Materials*, 5(1), 1–36. https://doi.org/10.1038/s41524-019-0221-0.

[22] Zhou, J., Li, E., Wei, H., Li, C., Qiao, Q., & Armaghani, D. J. (2019). Random Forests and Cubist Algorithms for Predicting Shear Strengths of Rockfill Materials. *Applied Sciences*, 9(8), Article 1621. https://doi.org/10.3390/app9081621

[23] Haggag, M., Yosri, A., El-Dakhakhni, W., & Hassini, E. (2022). Interpretable data-driven model for Climate-Induced Disaster damage prediction: The first step in community resilience planning. *International Journal of Disaster Risk Reduction*, 73, Article 102884. https://doi.org/10.1016/j.ijdrr.2022.102884.

[24] Wright, M. N., & Ziegler, A. (2017). ranger: A fast implementation of random forests for high dimensional data in C++ and R. *Journal of Statistical Software*, 77, 1–17. https://doi.org/10.18637/jss.v077.i01.

[25] Lee, J., Jeong, J., Jung, S., Moon, J., & Rho, S. (2022). Verification of De-Identification Techniques for Personal Information Using Tree-Based Methods with Shapley Values. *Journal of Personalized Medicine*, 12(2), Article 190. https://doi.org/10.3390/jpm12020190.

[26] Moon, J., Park, S., Rho, S., & Hwang, E. (2022). Robust building energy consumption forecasting using an online learning approach with R ranger. *Journal of Building Engineering*, 47, Article 103851. https://doi.org/10.1016/j.jobe.2021.103851.

[27] Altman, N., & Krzywinski, M. (2017). Ensemble methods: bagging and random forests. *Nature Methods*, 14(10), 933–935. https://doi.org/10.1038/nmeth.4438.

[28] Oshiro, T. M., Perez, P. S., & Baranauskas, J. A. (2012, July). *How Many Trees in a Random Forest?*. In: Perner, P. (Eds.), Machine Learning and Data Mining in Pattern Recognition (pp. 154–168). Springer. https://doi.org/10.1007/978-3-642-31537-4_13.

[29] Kuhn, M., & Johnson, K. (2013). *Applied Predictive Modeling.* Springer.

[30] Vartholomaios, A. (2019). A machine learning approach to modelling solar irradiation of urban and terrain 3D models. *Computers, Environment and Urban Systems,* 78, Article 101387. https://doi.org/10.1016/j.compenvurbsys.2019.101387.

[31] Feng, C., & Jiao, J. (2021). Predicting and mapping neighborhood-scale health outcomes: A machine learning approach. *Computers, Environment and Urban Systems,* 85, Article 101562. https://doi.org/10.1016/j.compenvurbsys.2020.101562.

[32] Chen, T., & Guestrin, C. (2016, August). *XGBoost: A Scalable Tree Boosting System.* Oral session presentation at the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. San Francisco, CA. https://doi.org/10.1145/2939672.2939785.

[33] Kuhn, M., Weston, S., Keefer, C., & Coulter, N. (2012). *Cubist Models for Regression.* Retrieved from

https://cran.r-project.org/web/packages/Cubist/. Accessed April 1, 2022.

[34] Liaw, A., & Wiener, M. (2002). Classification and regression by randomForest. *R News*, 2(3), 18–22.

[35] Kuhn, M. (2008). Building Predictive Models in R Using the caret Package. *Journal of Statistical Software*, 28, 1–26. https://doi.org/10.18637/jss.v028.i05.

[36] Chen, T., & He, T. (2017). *xgboost: eXtreme Gradient Boosting*. Retrieved from https://cran.r-project.org/web/packages/xgboost/. Accessed April 1, 2022.

[37] Malshe, A. (2019). *Data Analytics Applications*. Retrieved from https://ashgreat.github.io/analyticsAppBook/xgboost. Accessed April 1, 2022.

[38] Divina, F., Gilson, A., Goméz-Vela, F., García Torres, M., & Torres, J. F. (2018). Stacking Ensemble Learning for Short-Term Electricity Consumption Forecasting. *Energies*, 11(4), Article 949. https://doi.org/10.3390/en11040949.

[39] Hyndman, R. J., & Athanasopoulos, G. (2018). *Forecasting: principles and practice*. OTexts.

[40] Rahman, R., Otridge, J., & Pal, R. (2017). IntegratedMRF: random forest-based framework for integrating prediction from different data types. *Bioinformatics*, 33(9), 1407–1410. https://doi.org/10.1093/bioinformatics/btw765.

[41] Molnar, C. (2020). *Interpretable Machine Learning*. Lulu.com.