International Journal of Internet, Broadcasting and Communication Vol.15 No.3 176-184 (2023)
http://dx.doi.org/10.7236/IJIBC.2023.15.3.176

IJIBC 23-3-22

A Stay Detection Algorithm Using GPS Trajectory and Points of Interest Data

Eunchong Koh[1], Changhoon Lyu[2], Goya Choi[2], Kye-Dong Jung[3], Soonchul Kwon[4], Chigon Hwang[5]

[1]Graduate School of Smart Convergence Kwangwoon University, Korea
[2]LOYQU Inc., 14, Magokjungang 8-ro, Gangseo-gu, Seoul, Republic of Korea
[3]Professor, Ingenium College of liberal arts, Kwangwoon University, Republic of Korea
[4]Associate professor, Graduate School of Smart Convergence, Kwangwoon University, Republic of Korea
[5]Visiting Professor, Department of Computer Engineering, Institute of Information Technology, Kwangwoon University, Seoul, 01897, Republic of Korea
silvergun@kw.ac.kr,
{lyuchanghoon, goya.choi}@loyqu.com, {gdchung, ksc0226, duck1052}@kw.ac.kr

Abstract

Points of interest (POIs) are widely used in tourism recommendations and to provide information about areas of interest. Currently, situation judgement using POI and GPS data is mainly rule-based. However, this approach has the limitation that inferences can only be made using predefined POI information. In this study, we propose an algorithm that uses POI data, GPS data, and schedule information to calculate the current speed, location, schedule matching, movement trajectory, and POI coverage, and uses machine learning to determine whether to stay or go. Based on the input data, the clustered information is labelled by k-means algorithm as unsupervised learning. This result is trained as the input vector of the SVM model to calculate the probability of moving and staying. Therefore, in this study, we implemented an algorithm that can adjust the schedule using the travel schedule, POI data, and GPS information. The results show that the algorithm does not rely on predefined information, but can make judgements using GPS data and POI data in real time, which is more flexible and reliable than traditional rule-based approaches. Therefore, this study can optimize tourism scheduling. Therefore, the stay detection algorithm using GPS movement trajectories and POIs developed in this study provides important information for tourism schedule planning and is expected to provide much value for tourism services.

Keywords: *POI, k-means clustering, SVM, location estimation, GPS*

1. Introduction

Manuscript Received: July. 7, 2023 / Revised: July. 12, 2023 / Accepted: July. 16, 2023
Corresponding Author: duck1052@kw.ac.kr
Tel: +82-2-940-8379
Visiting Professor, Department of Computer Engineering, Institute of Information Technology, Kwangwoon University, Korea

In the past, cognitive technologies were based on rule-based knowledge, but recent advances in machine learning have led to research on predicting situations through learning. Machine learning in artificial intelligence is broadly classified into supervised and unsupervised learning, and technologies that combine or apply them are also emerging [1][2].

A point of interest (POI) is information about a location that may be of interest to a user, such as a place where people often go or a place that is meaningful to the user [3]. For example, POIs include information about major locations such as tourist attractions, cultural facilities, stations, markets, etc. [4].

In POI, movement trajectory is information based on GPS coordinates, which refers to movement route data calculated from the user's GPS coordinates collected at some point in time, with the coordinates changing continuously over time. This can be done by collecting the user's movement path using a mobile device to generate movement trajectories, and based on these data, k-means clustering techniques can be used to identify points of stay [4].

Using GPS coordinates, a learning model to predict the traveling trajectory and future location is based on vehicle location information passively collected from open-source data. The approach of first using trip itinerary and time information through machine learning improves the prediction accuracy, which can be achieved through machine learning methods that incorporate previous locations with GPS coordinates and associate locations with specific trip types [5].

In this paper, we study a system that uses GPS coordinates and POI (Point Of Interest) information from a smartphone to recommend a travel schedule by recognizing the situation in a trip. The measured GPS coordinates and POI data are used to calculate the vehicle's movement speed, activity trajectory, Time Series trajectory differ, and POI coverage, which are used as feature vectors for machine learning. This information does not have the labels required for supervised learning. Therefore, for supervised learning, labels are calculated using the k-means++ algorithm of the k-means clustering technique based on the input feature vectors calculated above. The presented feature vectors and labels are used to predict the probability of arrival by calculating whether the traveler has entered the POI area of a particular tourist destination during the journey through a Support Vector Machine (SVM). Therefore, this study can optimize the tourist itinerary. Therefore, the stay detection algorithm using GPS coordinates and POIs can provide important information in areas that require schedule planning, such as the travel industry, and is expected to provide much value for tourism services.

2. Related research

2-1. Location techniques

There are several approaches to real-time destination prediction. First, the method using trajectory data and historical GPS coordinates. Using POI information provided by publicly available geographic information systems and the user's GPS trajectory, a technique was proposed to generate a semantic trajectory consisting of industry information of POIs visited by the user through location information and POI coordinate values [3].

The second type of model requires a map database for the location of roads and previous travel routes. This model has a personalized prediction algorithm based on the prediction of the next trip based on the previous trip performed by the user, and the approach that deviates from this personalized approach and uses only the trajectory data presented in the current trip has the problem that it requires less data and post-processing, has

low accuracy, and cannot improve accuracy through historical data. Other approaches use map information, trajectory synthesis, and data mining. These methods also improve prediction accuracy, but require additional information and do not show higher prediction accuracy at the beginning of the trip. One such methodology is the use of Markov chains [5][6].

Third, the dwell point identification technique analyses the user's GPS coordinates and uses a dwell point identification technique. This is done by clustering the GPS coordinates to identify the user's point of stay. This identification method uses K-means clustering technique, G-means clustering technique, and DBSCAN technique based on GPS coordinates. In this paper, we use the k-means clustering technique [7].

2-2. Nonlinear SVM and K-means clustering

K-means clustering is a technique for finding a set of clusters from input data, where every cluster has at least one sample and every sample belongs to only one cluster. Equation 1 is the objective function () of the k-means clustering technique, where is the cluster centre and is a matrix with the information of the samples.

$$J(Z, A) = \sum_{i=1}^{n} \sum_{j=1}^{k} a_{ij} \| x_i - z_j \|^2 \tag{1}$$

In a nonlinear SVM, if , it is classified as either 1 or -1. This is , and the weight is equal to Equation 2.

$$\boldsymbol{\omega} = \sum_{i=1}^{n} \alpha_i y_i \mathbf{x}_i = \sum_{\alpha_k != 0} \alpha_k y_k \mathbf{x}_k \tag{2}$$

Using to convert to H- space and applying from the above expression, is converted to Equation 3.

$$d(\boldsymbol{x}) = \omega^T \Phi(\boldsymbol{x}) + b = \sum_{\alpha_k \neq 0} \alpha_k y_k K(\boldsymbol{x}_k, \boldsymbol{x}) + b \tag{3}$$

If , it is classified as 1, otherwise -1. This study uses k-means clustering for the labelling part to perform SVM, and SVM for classification [8][9].

3. System for applying the proposed algorithm

The system uses POI data, schedule data that specifies the POIs to be visited according to time, and GPS data that is inputted at regular intervals. In the preprocessing process, the distance between the location information is first calculated using the Haversine formula. Using the calculated distance and the time difference between the location information, the speed while traveling from each point to another can be obtained. The preprocessed information, including distance and speed, is used to calculate POI coverage, activity trajectory, and time series trajectory differ as input vectors for machine learning. POI coverage takes POI data and GPS data as input, checks whether the GPS data is within the range of the POI, and outputs a result based on that. Activity trajectory starts tracking when GPS data is within a predefined POI range and determines how the incoming GPS data moves within the POI range. Time series trajectory differ uses both POI data, schedule data, and GPS data to determine if there is a delay in the schedule and to determine the need for schedule adjustments. Each process outputs an input vector for training based on the input data, and the output input vector can be used to perform K-means clustering to generate labels, which can then be used for SVM training [10].

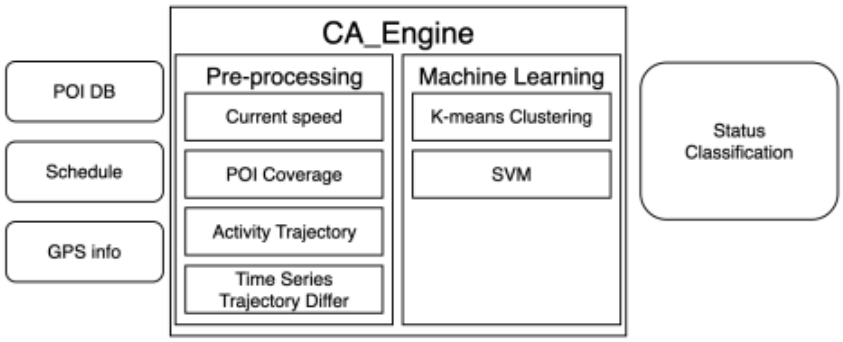

Figure 1. CA Engine

3-1. Pre-Processing

The Haversine formula is used to find the distance between two points on a sphere. The latitude and longitude values used in the Haversine formula are in radians, but since GPS data is based on decimal coordinates, they need to be converted to radians. In this case, a decimal value can be converted to radians, such as .

After converting the coordinate information of two points to radians, we can use the Haversine formula to find the direct line between the two points. Here, and are the latitude and longitude differences between the two points respectively, and and are the latitudes of the two points. Since this is a formula for distance on a sphere, we need to define the radius of the sphere, which in this case is , based on an approximation of the Earth's radius. The result of the distance calculation can be used as the basis for the algorithm's stay determination. For example, this is used to determine if GPS data is within range of a specific POI. The result of the distance calculation can then be compared to the radius of the POI to determine whether or not to stay.

(a) in Figure 2 shows the process of calculating POI coverage, which takes in GPS data and POI data to determine which POI is currently located. POI coverage is calculated through the interaction of GPS data and POI data. This algorithm provides important information for determining whether to stay.

First, it determines whether the GPS data is within the range of the POI. The GPS data and POI data include coordinate information and range data. The Haversine formula is used to calculate the distance between two GPS locations, which is used to determine whether the location is within the range of the POI.

Then, if your speed is below the threshold speed, you are determined to be stationary. By comparing the previous location in the GPS data to the current location, a path vector is calculated, which is used to calculate the speed of the movement. If the traveling speed is below the threshold speed and the GPS data is within the POI range, the GPS data is considered to be stationary [11].

Instead of outputting 1s and 0s, we convert the result to an average over all POI data and use it as an input vector to the SVM model. This is necessary for normal learning, so the current average value is compared to the average value that only includes the previous data, and if the average value has increased, the probability that the person is currently staying has increased.

(b) in Figure 2 shows the computation process of an activity trajectory. GPS data is input, converted to a vector, and a scalar value is calculated to represent the straightness of the entire movement path. The activity trajectory is an important factor to track and judge the user's activity based on the movement path. To do this, we use the last GPS data and the current GPS data to calculate the path vector. Represent the latitude and longitude of the last GPS data as , and the latitude and longitude of the current GPS data as .

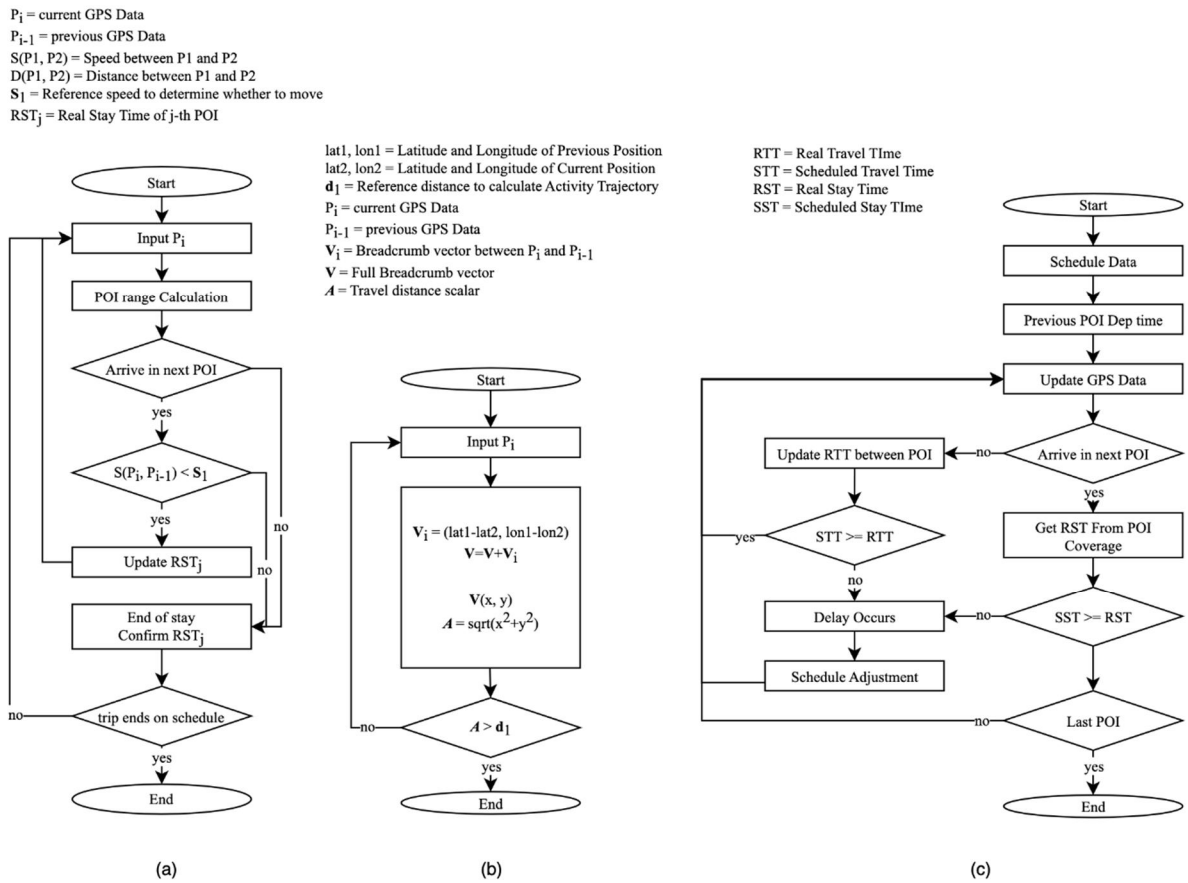

Figure 2. Processing Algorithm (a) shows the process of calculating POI coverage sequentially. (b) is the computation process of the activity trajectory. (c) is a flowchart showing the calculation of the time series trajectory differ

The movement path vector for calculating the activity trajectory is defined as: . This allows us to represent the magnitude and direction of the movement in each GPS data. After calculating the total movement vector, it can be converted to a scalar value to indicate the degree of straightness of the movement. We use Euclidean distance to calculate the magnitude of the scalar value A for the entire path is defined as .

Next, set the reference distance . If is at least twice as large as the reference distance , the final activity trajectory input vector value will be 2. Otherwise, , which is divided by the reference distance , is used as the input vector.

So, finally, the activity trajectory input vector can be defined as follows:

- If , then the activity trajectory input vector = 2
- If , then the activity trajectory input vector =

(c) in Figure 2 is a flowchart of the time series trajectory differ, which uses the scheduled travel/dwell time and actual travel/dwell time contained in the POI data to determine whether the schedule is delayed and improve the accuracy of SVM predictions. POI coverage determines whether GPS data exists within the range of the POI, and based on this, the time series trajectory differ calculates the actual travel time and dwell time. If the current state is moving, the actual travel time is calculated until it is within the range of the destination POI to calculate the error with the scheduled travel time, and when it is within the range of the destination

POI, the current state is changed to staying and the tracking of the dwell time is started.

Therefore, finally, the "time series trajectory differ" input vector can be defined as follows:

- If the scheduled time/actual time is 1, the "time series trajectory differ" input vector = 1.
- If less than 1, time series trajectory differ input vector = scheduled time/actual time

As the time series trajectory differ value approaches zero, it indicates that more time has been spent than scheduled, which is an abnormal situation. In this case, it is necessary to adjust the schedule according to the time spent.

3-2. Machine Learning

This system is designed to determine whether a person is staying or moving in a travel destination. At this time, it is impossible to make a judgment based on GPS data alone, so features are extracted through the 3-1 preprocessing process.

Since there are no labels for staying and traveling in the data after feature extraction, it is labeled using k-means clustering. This determines the feature vectors and labels, and SVM is used for classification. Since there are difficulties in determining the exact arrival or departure due to GPS errors and other geographical conditions, SoftMax is used to calculate the probability of staying and moving.

4. Implementation and evaluation

Figure 3 shows the raw data of the POI data included in the schedule data and the GPS data collected. The preprocessing described in Chapter 3 yields the data in Table 1(a). Since the data all have different ranges, we can use MinMaxScaler to normalize all the data to a value between 0 and 1 to get the data shown on the (b) of Table 1. Perform k-means clustering on the normalized data to generate labels for each piece of data. We set the number of clusters to 2 and used k-means++ to select the initial clustering points. Looking at the data visualized after k-means clustering, we can see that the normalized POI coverage values are concentrated around 0.2. This is shown in the Figure 4. This is because the POI coverage value is averaged over the whole, so as the GPS data accumulates, the number becomes smaller and smaller, and when normalized, it appears to be close to 0.2. The cluster centroids were each located close to both ends of the activity trajectory values, so the stronger the tendency to move out of POI coverage in the GPS data, the higher the probability of being classified as mobile.

	poi	coordinate		Time		
		Lat	Lng	Start	End	duration (min)
1	Busan Station	35.11489	129.04206	10:00	10:05	5
	Travel			10:05	10:30	25
2	Pork Soup Main Store	35.131782	129.092243	10:30	11:10	40
	Travel			11:10	11:30	20
3	Gwangalli Beach	35.15317	129.118652	11:30	12:50	80
	Travel			12:50	13:05	15
4	Marine City	35.155329	129.145935	13:05	14:25	80
	Travel			14:25	14:40	15
5	The Westin Chosun Busar	35.15596	129.15398			


```
"0": {
 "Time": "10:01:00",
 "latitude": 35.11521636348864,
 "longitude": 129.04152961131922
},
"1": {
 "Time": "10:02:00",
 "latitude": 35.11518196747183,
 "longitude": 129.04164555486938
},
"2": {
 "Time": "10:03:00",
 "latitude": 35.11516476946341,
 "longitude": 129.04170352664443
},
"3": {
 "Time": "10:04:00",
 "latitude": 35.11511317543818,
 "longitude": 129.04187744196963
},
```


Figure 3. Raw Data

Table 1. Preprocessed and normalized data

POI Coverage	Activity Trajectory Series	Trajectory Differ	Time
0.11111	0.37007	1	
-0.11111	0.30869	1	
-0.00909	0.24779	1	
-0.00757	0.15492	1	
0.07051	0.15556	1	
0.06043	0.15556	1	
0.05238	0.13433	1	

(a)

POI Coverage	Activity Trajectory Series	Trajectory Differ	Time
0.23636	0.11650	1	
0.23863	0.06968	1	
0.35576	0.06999	1	
0.34065	0.06999	1	
0.32857	0.05929	1	
0.31875	0.05929	1	
0.22242	0.03102	1	

(b)

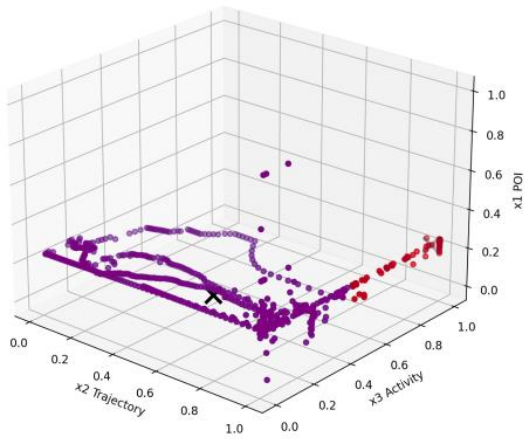

Figure 4. Clustered Data

```
Sample 123: T Probability: 0.00, F Probability: 1.00, Predicted: 0
Sample 124: T Probability: 0.03, F Probability: 0.97, Predicted: 0
Sample 125: T Probability: 0.03, F Probability: 0.97, Predicted: 0
Sample 126: T Probability: 0.03, F Probability: 0.97, Predicted: 0
Sample 127: T Probability: 0.76, F Probability: 0.24, Predicted: 1
Sample 128: T Probability: 0.95, F Probability: 0.05, Predicted: 1
Sample 129: T Probability: 0.97, F Probability: 0.03, Predicted: 1
Sample 130: T Probability: 0.99, F Probability: 0.01, Predicted: 1
```

Figure 5. SVM Result

After training the SVM model on the clustered data, we preprocess the separate data to classify it using the SVM model. At this point, we applied SoftMax to the SVM model so that the results can be displayed as probabilities, and the probabilities and classification results for staying and moving are shown as follows.

Based on the schedule data, individual GPS data were labelled as either stay or go to evaluate the performance of the model. The GPS data was collected every minute, resulting in a total of 280 data points, of

which 205 data points corresponded to stays and 75 data points corresponded to trips. As shown in the confusion matrix, 33 of the movements were classified as stays. This is an error that occurs when GPS data falls within the range of more than one POI, and it is determined that a person is already staying within another POI when they are actually moving. This can be fixed by adjusting the ranges so that they don't overlap, or by changing the way we determine if you are within a POI range.

Table 2. Confusion Matrix.

	Real: Travel	Real: Stay
Predicted: Travel	42	0
Predicted: Stay	33	205

Accuracy: 88.21%

5. Conclusion

This study uses the schedule information and GPS coordinates per unit time to calculate the travel route, and then calculates the labels through clustering techniques. The calculated labels are used as labels for SVM technique, and the feature vectors for applying SVM are speed, location, schedule matching, movement trajectory, and POI coverage calculated from the initial information, GPS coordinate data and schedule information. The extracted feature vectors are classified using SVM, and the classified information is predicted with probability using SoftMax.

This method is a combination of machine learning techniques that apply various variables compared to the existing rule-based prediction to calculate the arrival of a specific area, and can be applied not only to travel but also to tasks that require schedule correction, and can be applied to scheduling and adjustment in various fields, not only in the field of travel.

Acknowledgement

This research was supported by Seoul R&BD Program (IC220005) through the Seoul Business Agency (SBA) funded by The Seoul Metropolitan Government

References

1. S.Y. Lim, and J.D. Huh, “Technology Trends of Context Aware Computing Applications,” *Electronics and Telecommunications Trends,* Vol. 19, No. 5, pp. 31-40, 2004.
   DOI: https://doi.org/10.22648/ETRI.2004.J.190504
2. Raul Navarro-Almanza, Mauricio A. Sanchez, Juan R. Castro, Olivia Mendoza, Guillermo Licea, “Interpretable Mamdani neuro-fuzzy model through context awareness and linguistic adaptation,” *Expert Systems with Applications*, Volume 189, March 2022.
   DOI: https://doi.org/10.1016/j.eswa.2021.116098
3. Yuhee Jang, Juwon Lee, and Hyo-sang Lim, “A Technique for Generating Semantic Trajectories by Using GPS Positions and POI Information,” *KIPS Transactions on Software and Data Engineering*, vol. 4, no. 10, pp. 439–

446, Oct. 2015.
DOI: https://doi.org/10.3745/KTSDE.2015.4.10.439

4. S. Jeong, D. Jeong, and S. Lee, "Method of User Route Analysis based on POI through Stay Point Identification," *The Journal of Korean Institute of Information Technology*, vol. 19, no. 11, pp. 1–12, Nov 2021.
DOI: https://doi.org/10.14801/jkiit.2021.19.11.1
5. C. M. Krause and L. Zhang, "Short-term travel behavior prediction with GPS, land use, and point of interest data," *Transportation Research Part B: Methodological*, vol. 123, pp. 349–361, May 2019.
DOI: https://doi.org/10.1016/j.trb.2018.06.012
6. J. Yeon, L. Elefteriadou, and S. Lawphongpanich, "Travel time estimation on a freeway using Discrete Time Markov Chains," *Transportation Research Part B: Methodological*, vol. 42, no. 4, pp. 325–338, May 2008
DOI: https://doi.org/10.1016/j.trb.2007.08.005
7. K. Korovkinas, P. Danėnas, and G. Garšva, "SVM and k-Means Hybrid Method for Textual Data Sentiment Analysis," *Baltic Journal of Modern Computing*, vol. 7, no. 1, 2019,
DOI: https://doi.org/10.22364/bjmc.2019.7.1.04.
8. R. Kumar, G. Ganapathy, and J.-J. Kang, "A Hybrid Mod K-Means Clustering with Mod SVM Algorithm to Enhance the Cancer Prediction," *International Journal of Internet, Broadcasting and Communication*, vol. 13, no. 2, pp. 231–243, May 2021.
DOI: https://doi.org/10.7236/IJIBC.2021.13.2.231
9. S. H. Hong, B. Kim, and Y. G. Jung, "Correlation Analysis of Airline Customer Satisfaction using Random Forest with Deep Neural Network and Support Vector Machine Model," *International Journal of Internet, Broadcasting and Communication*, vol. 12, no. 4, pp. 26–32, Nov. 2020.
DOI: https://doi.org/10.7236/IJIBC.2020.12.4.26
10. K. P. Sinaga and M.-S. Yang, "Unsupervised K-Means Clustering Algorithm," *IEEE Access*, vol. 8, pp. 80716–80727, 2020.
DOI: https://doi.org/10.1109/access.2020.2988796
11. Y.-H. Kong, H.-J. Kim, Y.-J. Yi, and S.-J. Kang, "Development of Incident Detection Algorithm using GPS Data," *The Journal of the Korea institute of electronic communication sciences*, vol. 16, no. 4, pp. 771–782, Aug. 2021.
DOI: https://doi.org/10.13067/JKIECS.2021.16.4.771