

RESEARCH ARTICLE

Generating Labeled Multiple Attribute Trajectory Data With Selective Partial Anonymization Based on Exceptional Conditional Generative Adversarial Network

YEJI SONG¹, JIHWAN SHIN², JINHYUN AHN³, TAEWHI LEE⁴, AND DONG-HYUK IM⁵¹Department of Artificial Intelligence Convergence, Kwangwoon University, Seoul 01897, South Korea²Department of Artificial Intelligence Applications, Kwangwoon University, Seoul 01897, South Korea³Department of Management Information Systems, Jeju National University, Jeju 63243, South Korea⁴Smart Data Research Section, Electronics and Telecommunications Research Institute, Daejeon 34129, South Korea⁵School of Information Convergence, Kwangwoon University, Seoul 01897, South Korea

Corresponding author: Dong-Hyuk Im (dhim@kw.ac.kr)

This work was supported in part by the Institute of Information and Communications Technology Planning and Evaluation (IITP) Grant funded by the Korean Government [Ministry of Science and Information and Communications Technologies (MSIT)], Development of Approximate Database Management Systems (DBMS) Query Technology to Facilitate Fast Query Processing for Exploratory Data Analysis, 50%, under Grant 2021-0-00231; in part by the National Research Foundation of Korea (NRF) Grant funded by the Korean Government (MSIT), 40%, under Grant NRF-2021R1F1A1054739; and in part by the MSIT, South Korea, through the Information Technology Research Center (ITRC) Support Program, 10%, Supervised by the Institute for Information and Communications Technology Promotion (IITP), under Grant IITP-2023-2018-0-01417.


ABSTRACT Trajectory data generated in location-based service environments contain highly sensitive personal information, making them a prime target for privacy attacks. At the same time, however, valuable statistical information can be obtained from such private data. Optimizing this tradeoff between utility and privacy presents a challenge. This study introduces a novel method for partially anonymizing sensitive areas using a conditional generative adversarial network. The proposed method enables the learning of complex spatial, temporal, and categorical features of the selected sensitive area through the utilization of our condition label structure and loss function. In this study, we evaluate and analyze the contents by considering the spatial-temporal characteristics and dividing them into spatial usability and temporal usability. The experimental results demonstrate that the proposed method outperforms related models that employ generative adversarial networks. We achieved high scores in a majority of spatial evaluation items while also discussing the aspects that obtained relatively low scores.

INDEX TERMS Generative adversarial network, privacy protection, trajectory data, generating data, exceptional conditional.

I. INTRODUCTION

Users in location-based service (LBS) on social networking services (SNSs), such as Twitter and Foursquare, typically share their location using the global positioning system (GPS) [1], [2]. However, such sharing inevitably exposes sensitive personal information [3], [4]. Location data disclosed by users may cause personal privacy concerns regardless of the

frequency of exposure [5]. The infringement of personal privacy in sensitive locations such as hospitals and residential areas is a concern as the data could be targeted by malicious attackers. By contrast, location information for public facilities, such as large marts and schools, which have low privacy levels, is crucial. Location information can be useful for research and statistics. For example, such information can be used to monitor the spread of infectious diseases, such as COVID-19, by tracking the infection trajectory of infected people, classifying the visited dangerous areas [6],

The associate editor coordinating the review of this manuscript and approving it for publication was Jiachen Yang .

or investigating the impact of the movement trajectory between urban areas on the spread of infectious diseases [7], [8]. However, the exposure of sensitive information results in personal privacy concerns.

Information deletion can easily hide sensitive information. For instance, deleting an identifier that can identify an individual. However, even if the remaining attributes are combined, there is some scope for identification of the individual. Consequentially, data modification for anonymity is inevitable for attributes other than identifiers. The more properties you change, the higher the probability of anonymity. As a result, data usability is also reduced. Similarly, high-quality data can be obtained if data that are almost identical to the original are generated while maintaining usability. However, data that are not much different from the original increase the probability of identifying an individual, resulting in a decrease in anonymity.

The k -anonymity, l -diversity, and t -closeness form the basis of data anonymization methods [9], [10], [11]. The k -anonymity guarantees anonymity by modulating data so that all records have at least k identical quasi-identifier properties. For example, if k is 5 and the quasi-identifier attribute consists of gender and age in a table with 10 records, then 5-anonymity is guaranteed by changing the gender of at least five records to * and the age to 23–31. However, if all five records have the same sensitive attribute value, privacy is exposed even if k -anonymity is guaranteed if the records fall under the corresponding class. L -diversity is designed to protect privacy by securing at least L values of sensitive attributes in one class. For example, 3-diversity is guaranteed if three or more distinct sensitive properties exist in one class. In K -anonymity and l -diversity, quasi-identifiers are combined to protect individual records from identity exposure and from attribute exposure resulting from the same sensitive attribute value in an equivalence class. Moreover, a skewness attack exists wherein the distribution of sensitivity values in an equivalence class differs considerably, and a similarity attack occurs wherein the sensitivity values in an equivalence class are similar to each other. To defend against these attacks, in t -closeness, the relationship between sensitive attribute values within an equivalence class is considered.

Various other anonymization methods, including those that supplement the existing anonymization methods such as k -anonymity, l -diversity, and t -closeness, have been studied. Temuujin et al. [12] proposed a method for applying l -diversity based on the characteristic that records in a table are dynamically inserted and deleted. Jeon et al. [13] conducted a study to guarantee the privacy of individuals in public information by applying l -diversity to the dynamically changing resource description framework (RDF). Various anonymization studies are conducted in these scenarios, and differential privacy has been used as an anonymization method in these situations.

Differential privacy is a mathematically verifiable method that is used in various fields where data are used, as well

as in deep learning [14], [15], [16]. In differential privacy, anonymity is ensured regardless of the inclusion of personal information in the data. If a person's information is deleted or inserted into data that change dynamically, the amount of information in the entire data changes considerably. Therefore, the amount of deleted or inserted personal information can be tracked, which may result in an attack on personal privacy. Differential privacy is used to prevent privacy attacks by correcting the entire data mathematically. Kang et al. [17] mathematically defined the privacy level of approximate data generated by the conditional generative adversarial network (cGAN) by applying differential privacy to the cGAN.

The GAN is a powerful generative model, which has been widely adopted in many fields in recent years [18]. The input data used for learning to imitate generate fake data that are difficult to distinguish from real data. A hypothetical result that resembles real data but does not actually exist can be thought of as being anonymized. Therefore, it is useful not only for image generation but also for problems with applying anonymity.

The main disadvantage of the original GAN it cannot directly intervene in the output. Therefore, many studies have continued to develop derivative GAN models to intervene in outputs. Typically, there is a cGAN that gives additional conditions to the GAN so that an output that fits the label desired by the user is generated [19]. In the case of a dataset with labels in one set of data, the output can be learned to produce a specific label, resulting in the desired result by the user. This allows users to directly intervene in the GAN model, which mimics the entire learning data. If the original GAN is used to anonymize all information that exists in the database, all information can be anonymized with similar intensity. That is, information that need not be anonymized may be converted, or sensitive information may be less protected. Path data have different sensitivity levels for each visit point, and thus, all need not be anonymized with the same intensity. Therefore, less sensitive branches (e.g., restaurants, marts, etc.) should be close to the original, and sensitive branches (e.g., hospitals, police stations, etc.) should be strongly anonymized. To implement this, we devised a method to apply cGAN so that sensitive labels can be intensively hidden.

The major contributions of this study are as follows:

- The results of synthetic trajectory generation using the GAN confirmed that the method cannot be directly used in the output. Therefore, the method cannot distinguish between a sensitive area and a less sensitive area.
- In the cGAN, a specific condition label is assigned to specific data to obtain the output indicated by the label value assigned to the condition. Thus, only a specific point of the trajectory can be selected and hidden. Therefore, the trajectory selective partial anonymization (TrajSPA) method was proposed to select a partial area and reflect it in the output to create trajectory data similar to the original.

- SPALoss, which is suitable for cGAN, was designed by modifying the condition label structure and the existing loss formula of trajectory data, namely, spatial, temporal, and categorical complex data.
- Evaluating the anonymity and spatial and temporal utility of the approximate data generated by learning the shape of the original data revealed that the proposed model retained the distribution and characteristics of the original data. Consequently, the limitations caused by using the GAN can be overcome.

The remainder of the paper is structured as follows. In Section II, we discuss related research and the background for this study. Section III outlines the functioning of TrajSPA, a model that ingeniously combines cGAN and LSTM networks. In section IV, we present the experimental results of TrajSPA in comparison with the outcomes of prior studies and other models adopting similar approaches. Finally, Section V presents the conclusion of this study.

II. RELATED WORK

In terms of sensitive information, the medical field stands out as a significant domain where the privacy-utility balance has been the main subject of study. Numerous research efforts have sought solutions using data from this field. For instance, Yoon et al. [20] developed the anonymization through data synthesis-GAN framework, which facilitates the generation of synthetic data approximations for sharing patient data. Indhumathi and Devi [21] devised the healthcare Cramér GAN for generating synthetic medical data and compared its performance with the Wasserstein GAN. Piacentino et al. [22] proposed a GAN-based method for synthetic ECG generation to anonymize user information in distributed data.

In addition to the medical field, GAN frameworks have been employed in other domains, such as video creation and distribution, including social networking services (SNS) and autonomous vehicles. For example, Tieu et al. [23] introduced the spatial transformer-GAN to generate images for safeguarding the identities of individuals in videos shared on social media. Similarly, Xiong et al. [24] developed an anomaly detection GAN to prevent personal information leakage through videos recorded by autonomous vehicles.

In the context of trajectory data, the GAN framework has also been utilized in prior studies. Rao et al. [25] proposed the long short-term memory (LSTM)-TrajGAN, which employs an LSTM layer to learn the traffic data characteristics. Shin et al. [26], on the other hand, addressed the limitations of the GAN model by introducing a class-level trajectory data generation model using the auxiliary classifier GAN (AC GAN). Kim et al. [27] proposed an adversarial autoencoder (AEE) for applying differential privacy. Similarly, Zhang et al. [28] introduced LGAN-DP using a GAN framework with differential privacy. Additional information related these studies can be found in Table 1.

In our research, we aimed to build optimal solutions to overcome the limitations identified in Table 1. To achieve this, we utilized cGAN to generate trajectory data with

TABLE 1. Summary of existing studies using trajectory data and GAN.

	Advantages	Limitations
LSTM-TrajGAN, Rao et al. [25]	- Proposed a method of generating trajectory data using GAN. - Model design considering time series characteristics of trajectory Data.	- No direct involvement in data generation.
TCAC-GAN, Shin et al. [26]	- Higher performance of location data by using AC-GAN to better change sensitive categories.	- No consideration for temporal usability.
AEE, Kim et al. [27]	- Provides mathematically definable anonymity based on differential privacy.	- Data loss due to grid-based spatial reduction.
LGAN-DP, Zhang et al. [28]	- Applies differential privacy using clustering algorithms.	- Lacks comparison of anonymity and utility with the original data.

complex properties, which helped intervene in the generated data. Furthermore, we sought to incorporate temporal usability analysis in our trajectory data study. By adopting these approaches, we aimed to enhance the comprehensiveness and effectiveness of our model.

III. METHODOLOGY

In this section, explain the methodology of TrajSPA proposed in this study. We explain the overall functioning of the model, starting from how the data used by the model are processed to the model's structure and the Loss function.

A. DATA PROCESSING

Preprocessing is necessary for trajectory data to be used in model training. The raw data may contain unnecessarily large values or values that are difficult to be comprehended. Preprocessing is essential as it involves the transformation of the data to facilitate model training without losing information. In this section, we describe preprocessing, including the condition labels to be used for exception handling.

1) TRAJECTORY DATA PREPROCESSING

In this study, the NYC check-in data provided by Foursquare [29] was used. Regarding the trajectory data, as displayed in Fig. 1, the temporal, spatial, and category information of the points visited by a user was grouped into a trajectory on a weekly basis. The temporal information of the visit point consists of weekday and hour information. When this information is learned, it is one-hot encoded in a vector that matches the size of the property and has a size of seven because a week has seven values from Monday to Sunday. For example, a value for Wednesday is expressed

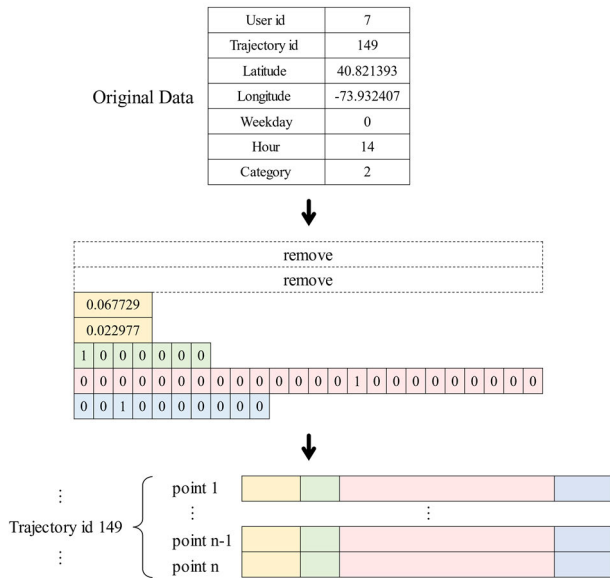


FIGURE 1. Original trajectory data and preprocessing.

as [0, 0, 1, 0, 0, 0, 0]. Because the hour, minute, and second values of the original data exist together, the attribute size becomes large. Thus, the size occupies an unnecessarily large space when expressed in a one-hot format. Therefore, the minute and second values were deleted from the hour attribute to ensure that only the values from 0 to 23 were expressed as a vector of size 24. Spatial information consists of latitude and longitude, and when the actual location was outside New York City, it was removed from the dataset. Before training, the data were standardized based on the central point among all latitude and longitude values to facilitate training. For the category attribute, 10 types of large categories provided by Foursquare were used. The spatial information was used for training by one-hot encoding as a vector of size 10, similar to the temporal information.

2) CONDITIONAL DATA

In the cGAN, the given condition with input data represents the data. Because data have only one condition, the data with the desired condition value can be generated as an output by applying the condition. Thus, if the condition value expressing the data is replaced by another value, then the data are learned by changing the value based on the replaced condition.

Because trajectory data have spatiotemporal characteristics, we used conditions by grouping them by latitude, longitude, hour, and category, which exhibit the same properties as the data. The location information to be hidden can be expressed with the same category value, such as “hospital” and “personal place.” Therefore, the category attribute was used to select sensitive information to be hidden. Fig. 2 illustrates an example of how to condition the data when the category of the sensitive data is set to 1. For data with a category other than the selected value (top of

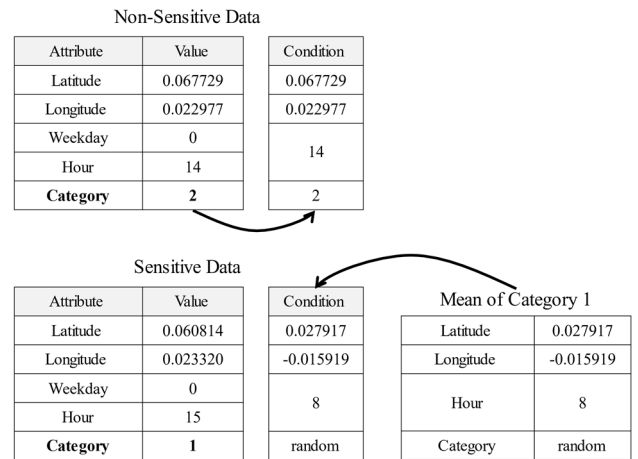


FIGURE 2. Conditions of non-sensitive and sensitive data.

the figure), we use the original data. However, if the category matches the selected value (bottom of the figure), we employ a pre-determined mean value. To introduce variability in conditions, we assigned a value between 0 and 9 to the category attribute within the sensitive areas. The category attribute had the same value in sensitive areas, and thus, a value between 0 and 9 was selected to obtain various conditions. The condition was applied to data corresponding to the selected category value by replacing it with a pre-defined value. In data other than the selected category value, the latitude, longitude, hour, and category values of the data were used as the condition without changing them. The replacement condition value was the mean value of the data with the corresponding category value. The data that contained sensitive areas were trained and changed based on the median value of values in the same category. Thus, the values can retain the distribution of the entire data.

B. FRAMEWORK

Fig. 3 displays the structure of the TrajSPA model. This model can be broadly categorized into generator and discriminator. The generator is trained by receiving preprocessed actual trajectory data as the input, and the discriminator receives synthetic trajectory data generated by the generator as the input. The generator and discriminator are trained by having them compete and increase their own performance to deceive their opponents. The GAN is a representative deep learning mini-max algorithm.

1) GENERATOR OF TrajSPA

The objective of the generator is to generate fake data that appear as identical as possible to the input data. The generated fake data are passed to the discriminator, and the weight is adjusted such that the discriminator cannot distinguish whether it is real or fake. The input of the generator is real data, and noise is added to ensure that the result becomes as similar as possible rather than copying the real data as they

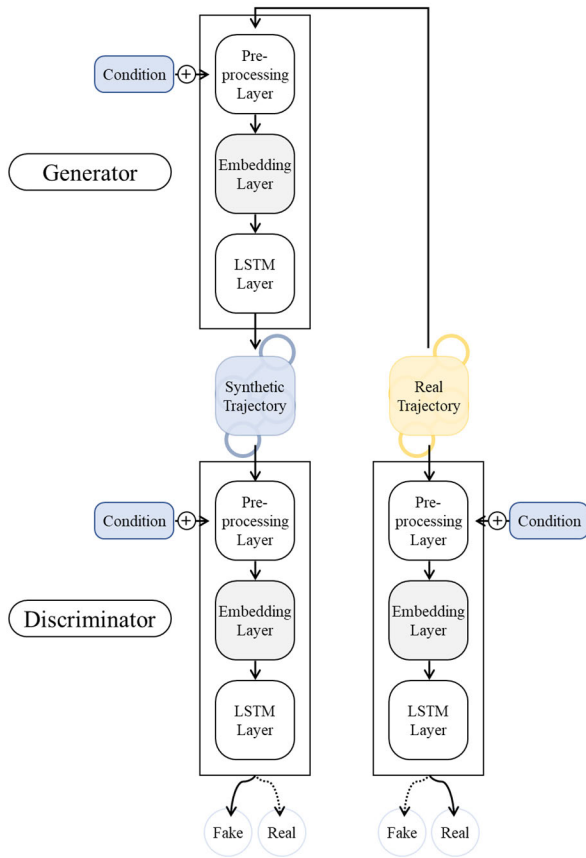


FIGURE 3. TrajSPA model.

are. The real data are transferred to the hidden layer of the generator by adding the conditions described in Section 2.2. The hidden layer proceeds to the next step by embedding the trajectory data in 100 dimensions. The trajectory data, which are spatiotemporal data with both spatial and temporal properties, pass through the LSTM [30] layer, where temporal characteristics can be learned well. This result was implemented as a many-to-many LSTM layer. The input data are learned, and output data of the same size are obtained accordingly. Therefore, the LSTM layer sequentially receives the points present in each trajectory and subsequently generates trajectory data while maintaining temporal characteristics. The data that pass through the LSTM layer are decoded and converted into trajectory data for publication.

2) DISCRIMINATOR OF TrajSPA

The objective of the discriminator is to accurately determine whether the input trajectory data are real or fake. Unlike the generator, in the discriminator, trajectory is not used as an output value. However, it indicates whether the input data are a real trajectory with a value between zero and one. Moreover, the discriminator goes through two training sessions inside the TrajSPA model. First, the discriminator learns the difference between real and fake data by receiving the original data as input and inputting the data generated by the generator.

The discriminator also has a similar structure to the generator, as displayed in Fig. 4. The input trajectory data are pre-processed to ensure that the model can be trained with them and they are embedded in 100 dimensions. The LSTM layer of the discriminator is implemented in a many-to-one format. This format is suitable for determining whether the input trajectory data are real or fake. The data that pass through the LSTM layer are the prediction result of the data received by the discriminator. Therefore, the result is displayed as a value between 0 and 1 using the sigmoid activation function. The corresponding value is used to train the generator again, and the generator is trained so that this value approaches zero.

3) TrajSPA G-LOSS AND D-LOSS

In deep learning models, the loss in every learning cycle is minimized during the learning process, and finally, optimization is performed to minimize the overall cost of the model. The objective function defines the optimization task of such a model as a function, which indicates the direction wherein the model is trained. The GAN and cGAN are identical except that cGAN adds a label to the input. Therefore, these models have similar objective functions, which are expressed as follows:

$$\min_G \max_D V(D, G) = Term_1 + Term_2 \tag{1}$$

$$Term_1 = \mathbb{E}_{x \sim p_{data}(x)} [\log D(x)] \tag{1a}$$

$$Term_2 = \mathbb{E}_{z \sim p_z(z)} [\log(1 - D(G(z)))] \tag{1b}$$

$$\min_G \max_D V(D, G) = cTerm_1 + cTerm_2 \tag{2}$$

$$cTerm_1 = \mathbb{E}_{x \sim p_{data}(x)} [\log D(x | y)] \tag{2a}$$

$$cTerm_2 = \mathbb{E}_{z \sim p_z(z)} [\log(1 - D(G(z | y)))] \tag{2b}$$

The objective functions of GAN and cGAN are Eqs. (1) and (2). $Term_1$, $Term_2$, $cTerm_1$, and $cTerm_2$ briefly expressed in equations (1) and (2) may be expressed as (1a), (1b), (2a), and (2b), respectively. Both equations have the same form except for y and perform the same operation. The $x \sim p_{data}(x)$, which is commonly used in both equations, represents data sampled from a probability distribution of the real data, and $D(x)$ distinguishes the real data by using real data x as input. $z \sim p_z(z)$ indicates that the data were sampled from random noise using a Gaussian distribution. $G(z)$ generates fake data using z as the input, and $D(G(z))$ classifies the data as fake data. Here, E represents the expected value. Thus, $\mathbb{E}_{x \sim p_{data}(x)}[f]$ and $\mathbb{E}_{z \sim p_z(z)}[f]$ represents expected values to be obtained by inputting samples of the real data and noise into the function defined in each $[f]$. Therefore, the generator should ensure that the objective function $V(D, G)$ is minimized such that $D(G(z))$ becomes 1. Conversely, the discriminator should maximize the objective function $V(D, G)$. Thus, the discriminator should make $D(x)$ equal to 1 and $D(G(z))$ equal to zero. Eventually, the two collide with each other, which causes the minimax problem. Eq. (2) has a difference that x and y are given the label y corresponding to the condition.

According to the objective function $V(D, G)$ defined in Eq. (2), the discriminator appears as a value between 0 and 1. Thus, binary cross entropy (BCE) loss is used. However, because the data used are trajectory data wherein spatial, temporal, and categorical values are combined, the loss function should be modified [25]. The TrajLoss proposed by Rao et al. is as follows [25]:

$$\text{TrajLoss}(y^r, y^p, t^r, t^s) = L_{OB} + L_{Os} + L_{Ot} + L_{Oc} \quad (3)$$

$$L_{OB} = \alpha L_{BCE}(y^r, y^p) \quad (3a)$$

$$L_{Os} = \beta L_s(t^r, t^s) \quad (3b)$$

$$L_{Ot} = \gamma L_t(t^r, t^s) \quad (3c)$$

$$L_{Oc} = c L_c(t^r, t^s) \quad (3d)$$

TrajLoss receives four values as input, y^r and y^p represent the ground truth label and the result predicted by the discriminator, respectively. Furthermore, t^r and t^s represent real and synthetic trajectories, respectively. Each of these four inputs is categorized into four again, calculated according to its characteristics, and then added. Here, α , β , γ , and c multiplied before each loss have distinct weight values. Here, L_{BCE} is a BCE loss function, and L_s , L_t , and L_c are functions for calculating the loss for spatial, temporal, and categorical properties, respectively. Spatial loss is calculated using the least square error (LSE), and temporal and categorical loss are calculated using Softmax Cross Entropy (SCE). Thus, such a configuration of TrajLoss is suitable for use in the LSTM-TrajGAN model implemented based on the original GAN. However, because the proposed TrajSPA is implemented based on the cGAN, the loss for the condition label is calculated as follows:

$$\text{SPALoss}(Y, T, l^r, l^s) = TLo + L_{Os} + L_{Ot} + L_{Oc} \quad (4)$$

$$TLo = \text{TrajLoss}(Y, T) \quad (4a)$$

$$L_{Os} = \tau L_s(l^r, l^s) \quad (4b)$$

$$L_{Ot} = \varphi L_t(l^r, l^s) \quad (4c)$$

$$L_{Oc} = \omega L_c(l^r, l^s) \quad (4d)$$

where Y and T denote y^r , y^p , and t^r , t^s , respectively, and TrajLoss performs the same role as in Eq. (3). Here, l^r and l^s represent the actual label and the label generated together with the synthetic trajectory, respectively. Furthermore, τ , φ , and ω denote weights for controlling various label loss results. TrajSPA may be confused with the trajectory input by the generator and discriminator because the data given as a condition are spatial, temporal, and categorical. Therefore, the weights τ , φ , and ω are assigned higher values than the weights β , γ , and c so that the model can focus on the given condition label. If the weight of the input data is 1 and the weight of the label is 8, the calculated value of the label loss becomes larger than the input data. Therefore, we focused on the label loss to reduce the total calculated value of SPALoss. Consequently, the trajectory of the input data may result in a change in the spatial, temporal, and categorical values included in the label given as a condition. Here, L_s , L_t , and L_c represent spatial, temporal, and categorical label losses,

respectively. Furthermore, L_{ls} is calculated using LSE, and L_{lt} and L_{lc} are calculated using SCE.

IV. EXPERIMENTAL RESULTS

The experimental results were compared based on the LSTM-TrajGAN model [25] using the original GAN. For comparative evaluation, the anonymity and usefulness of the generated trajectory data were evaluated.

A. DATASETS AND EXPERIMENT DESIGN

The dataset was created by collecting check-in records in New York City (NYC) and Tokyo for approximately 10 months, from April 12, 2012, to February 16, 2013, on Foursquare [29]. To consider only actual NYC data, the data were removed if the latitude and longitude were not those of NYC. Attributes included in the dataset consisted of userID, trajectoryID, latitude, longitude, temporal information (weekday, hour), and category (Food, Travel & Transport, Residence, Professional & Other Places, Shop & Service, Outdoors & Recreation, Collate & University, Arts & Entertainment, Nightlife Spot, Event). In actual training, the userID and trajectoryID attributes were not used. The dataset was divided into training and testing datasets in a 2:1 ratio. The number of users in the training dataset was 193, that of trajectories was 2,052, and the total number of points was 44,809. Similarly, the number of users, trajectory, and points in the testing dataset were 193, 1,027, and 22,153, respectively. In the dataset provided by Foursquare, data with values of latitude and longitude outside of the actual NYC location and trajectory lengths of less than 10 were removed.

TABLE 2. Attributes and details of the dataset used.

Attributes	Range
Latitude	40.5566 ~ 40.98139
Longitude	-74.2671 ~ -73.6889
Weekday	[0, 1, 2, 3, 4, 5, 6]
Hour	[0, 1, 2, 3, 4, 5, ..., 21, 22, 23]
Category	[0, 1, 2, 3, 4, 5, 6, 7, 8, 9]

Each attribute value was pre-processed to make learning easy and was then learned as a vector. The information to be transmitted as a condition was learned by assigning the remaining nine category labels to each point of data except for the category to be hidden. To learn the trajectory data, which are spatio-temporal data, the generator and discriminator can consider the continuity of point information according to the time of the trajectory data using the LSTM model. Label No. 1 (Travel & Transport) was selected for experimentation. Label No. 1 was selected arbitrarily for the experiment, although it was not a sensitive place. Table 2 explains each attribute in detail.

The training conditions of the LSTM-TrajGAN model were set the same to train it under the same conditions as the existing models. The full training of each model was repeated 2,000 times. The batch size was 256, the learning rate was 0.001, and the Adam optimizer was used as the optimizer.

B. RESULTS

The anonymity test was performed using the trajectory user linking (TUL) method for linking trajectory and user information, and the data usefulness test was conducted separately for spatial and temporal data. In the case of spatial information, the score of TrajSPA was similar to or higher than that of other models. In terms of temporal information, the score of other models was calculated to be higher than that of TrajSPA. The TrajSPA model performed better than the existing models in terms of data usability and anonymity.

TABLE 3. Comparison of anonymization score test results by model.

Method	LSTM-TrajGAN	TrajSPA
ACC@1	0.406037	0.363194 (-10.6%)
ACC@5	0.651412	0.654333 (0.4%)
Macro-F1	0.345144	0.308291 (-10.7%)
Macro-P	0.390910	0.363444 (-7.0%)
Macro-R	0.374779	0.352025 (-6.1%)

1) ANONYMIZATION RESULTS

The TUL score is a test of whether the user who generated the trajectory can be connected. Because the accuracy score is typically an indicator of the performance of the model, it is expressed as a value between zero and one. The closer the value to one, the better the performance of the model; that is, the higher the accuracy. The TUL evaluation is a test of whether the user is well connected based on the trajectory data. The closer the value to one, the lower the anonymity, which indicates that if data are disclosed, the privacy of the user who generated the trajectory data is not protected. Therefore, the closer the score is to zero, the more difficult it is to find users using only the generated trajectory data. This phenomenon indicates that anonymization was performed satisfactorily. Table 3 summarizes the test results. The relative difference from the LSTM-TrajGAN score is presented at the right to TrajSPA scores. When five accuracy scores (Top-1 Accuracy, Top-5 Accuracy, Macro-F1, Macro-P, and Macro-R) were calculated, the anonymization score of the TrajSPA model was calculated to be higher than that of other models. This phenomenon suggests that the data in the selected category were distorted.

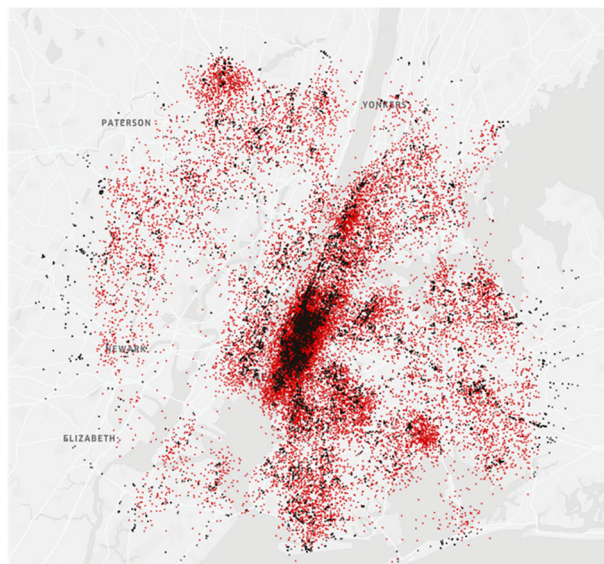
TABLE 4. Comparison of spatial data usability test results.

Model Method	MIN	MAX	AVG	MEAN
LSTM-TrajGAN				
Hausdorff	0.006838	0.062886	0.020647	0.019752
Manhattan	0.086091	2.866825	0.428797	0.328549
Euclidean	0.073883	2.254540	0.332206	0.253892
TrajSPA				
Hausdorff	0.006454	0.085016	0.019176	0.017582
Manhattan	0.080386	2.614288	0.415828	0.316735
Euclidean	0.062978	2.007773	0.324835	0.248615

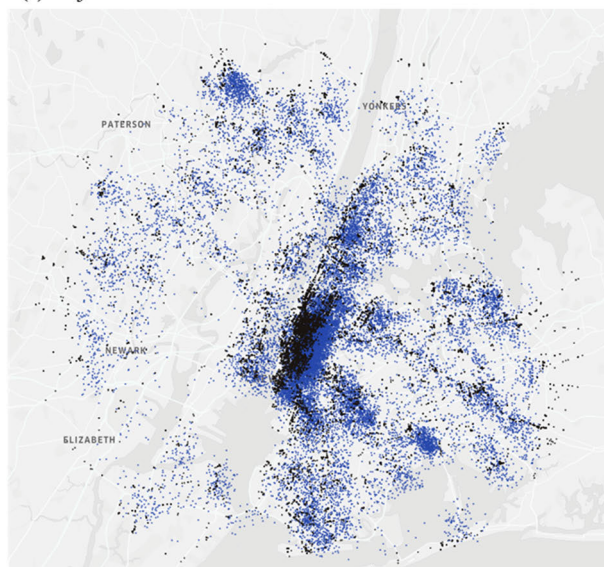
2) SPATIAL INFORMATION RESULTS

Spatial information was evaluated based on a comparison of the distance values of latitude and longitude of the original trajectory data with those of the synthetic trajectory data. The trajectory data on the coordinate plane are expressed as a single graph that connects points representing locations. Two graph distances can be obtained through the average Hausdorff distance [31], [32] test. Another method is to determine the linear distance between each point in one trajectory and evaluate it with the calculated Manhattan and Euclidean distances. Table 4 provides a summary of the results for evaluating spatial information, showcasing scores for both the base model and TrajSPA. Notably, the TrajSPA model achieved the highest score in all indicators for the Manhattan and Euclidean methods, except for MAX in the Hausdorff method, where it still performed significantly well.

Fig. 4 displays the data generated by the TrajSPA and LSTM-TrajGAN models. The black dots represent the original data, and the red dots represent the data generated by TrajSPA. The black and red dots are displayed on (a) of Fig. 4 for comparison between the distribution of the original data and generated data. Similarly, the blue dots represent data generated by the LSTM-TrajGAN model, and they are displayed together with black dots representing the original data on (b). The data (blue dots) generated by LSTM-TrajGAN ((b) in Fig. 4) were skewed to the right compared with the original data (black dots). The data (red) generated by the TrajSPA ((a) in Fig. 4) model were more consistent with the location of the original data than the LSTM-TrajGAN. Both models simulate the distribution of the original data well, but the LSTM-TrajGAN replicates the original data more clearly. Because TrajSPA was trained by transforming point data with sensitive areas into other categories, it appears to spread slightly from the center under the impact of data with sensitive areas. This feature simulates places that many people visit publicly, such as dense areas or public places.



(a) TrajSPA



(b) LSTM-TrajGAN

FIGURE 4. Plot comparing generated data and latitude and longitude of original data with dots on the NYC map.

However, places that are far from the center or should be protected are displayed as distinct points by widening the dispersion.

3) TEMPORAL INFORMATION RESULTS

Temporal information was used to evaluate the similarity between weekday and hour properties. One trajectory runs consecutively in time from Monday to Sunday. For example, 23 o'clock on Tuesday is a consecutive number that differs from 0 o'clock on Wednesday by 1 h. However, because weekday and hour are stored separately as integer values of 0 to 6 and 0 to 23, respectively, combining the two properties into one value becomes critical. By using a simple method,

the weekday value of each point is multiplied by 24, and subsequently, it is added by the hour. Therefore, 23 o'clock on Monday is expressed as 23, and 23 o'clock on Tuesday is expressed as 47. Using this method, 23 o'clock on Tuesday and 0 o'clock on Wednesday have the values of 47 and 48, respectively. Thus, successive temporal information can be expressed to evaluate similarity.

TABLE 5. Comparison of temporal data usability test results by model.

Model	MIN	MAX	AVG	MEAN
LSTM-TrajGAN	97.6%	100%	99.9%	99.9%
TrajSPA	94.2%	100%	99.9%	99.9%

Table 5 presents a summary of the results of evaluating temporal information using cosine similarity. As displayed in this table, the LSTM-TrajGAN model has the best similarity, whereas the TrajSPA was calculated with a low score. However, with the exception of the 3% difference in the MIN index, all models achieve more than 99% performance, and TrajSPA exhibits comparable performance.

TABLE 6. Comparison of temporal data usability test results by model.

Weekday	Original	LSTM-TrajGAN	TrajSPA	TrajSPA (sort)
0	0	0	21	10
0	0	0	21	10
0	0	0	21	11
0	10	10	10	11
0	10	10	10	11
0	11	11	11	11
0	11	11	11	11
0	11	11	11	11
0	11	11	11	11
0	11	11	11	12
0	11	11	11	12
0	11	11	11	21
0	12	12	12	21
0	12	12	12	21
2	23	22	11	11

The trajectory showing a similarity of 94.2% in Table 5 can be confirmed in Table 6. The data in Table 6 express Weekday and Hour of one trajectory with ID 409 among the entire dataset. The original in the first column indicates that the original data set was imported, and LSTM-TrajGAN and TrajSPA denote that the trajectory information corresponding to the ID of 409 is obtained from the route data generated

by each model. The TrajSPA generated the Hour value at points 1–3 as 21 and the Hour value at point 15 as 11. These values differ considerably compared to those of the LSTM-TrajGAN model, wherein the Hour at point 15 was changed to 22. However, the original value of points 1–3 is 0, which denotes midnight. The difference between 21:00 and 0:00 is only 3 h, not 21 h. Moreover, because the original data were sorted in ascending order based on the temporal data, the synthetic data should be rearranged based on the Weekday and Hour data. Trajectory 409 after sorting can be found in the TrajSPA (sort) row in Table 6. Again, the Hour values of more points than before sorting did not match the original data, but the error between the points decreased.

C. DISCUSSION

From TABLE 4, it was observed that evaluating spatial data usability, TrajSPA scored lower on one item, specifically the Hausdorff evaluation method. This method compares distances between all points of two graphs to obtain the average distance from the nearest point at each location. If one trajectory includes a large number of sensitive points, it can significantly alter the generated trajectory. Thus, with Hausdorff, the longer the trajectory, the greater the potential difference. We acknowledge that longer trajectories provide more information and increase the risk of individual identification, making it more challenging to protect sensitive points. In future studies on spatial data usability of long trajectories, we will attempt to address this issue.

Regarding temporal usefulness, TrajSPA has a lower score compared to the existing model, LSTM-TrajGAN. However, as mentioned previously, higher similarity with the original data leads to reduced anonymity. In the anonymity test over time, the TrajSPA model ensures superior anonymity despite the lower temporal usefulness score.

V. CONCLUSION

The cGAN model was adopted to generate similar trajectory data using the trajectory dataset of Foursquare users. Furthermore, the condition was used as input data during training to change the generated result data. The trajectory data exhibit distinct sensitivity depending on whether the classification of the visited place is a public facility used by many people or a place that an individual visits for a special purpose. Therefore, a performance comparable to the existing model can be obtained, and anonymity can be increased by intentionally hiding specific point information that can identify individuals while simultaneously preserving insensitive data from which significant statistical information can be obtained. In particular, the proposed model differs from existing models in that it can affect the output by selecting only the desired part to hide the sensitive locations. In Section II, we summarized relevant studies to confirm the necessity and distinctiveness of this research. In Section III, we discussed the methodology of the TrajSPA model. We explained the preprocessing of trajectory data and the structure of the model for its utilization. Additionally, we proposed a new Loss function and explained

the principle of applying exception handling conditions to the model. The performance of the proposed model in this study is summarized in Section IV. We study the anonymization, spatial usability, and temporal usability of the data to verify the model's performance. Barring some indicators of anonymization and spatial usability, TrajSPA demonstrated performance similar to or better than the base model. There was a slight difference in temporal usability, emphasizing the need to understand the continuity of time.

In future works, we will address certain limitations by considering the following points. First, we will expand the scope of data used beyond labeled data to include trajectory data without category labels. This will enable conditional generation of trajectory data without labels. Second, recognizing that longer trajectories may contain more useful and sensitive information, we will implement conditions during trajectory creation to enhance data usability. Last, to improve temporal data usability, we will focus on data preprocessing and learning approaches that take temporal characteristics into account. These measures will aid in overcoming the identified limitations and enhance the overall effectiveness of our study.

AUTHOR'S CONTRIBUTIONS

Yeji Song and Dong-Hyuk Im conceived the problem and supervised the overall research; Yeji Song and Jihwan Shin implemented the algorithm and performed the experiments; Taewhi Lee and Jinhyun Ahn clarified some points that helped Dong-Hyuk Im to supervise the research; Yeji Song, Jihwan Shin, Jinhyun Ahn, Taewhi Lee, and Dong-Hyuk Im wrote the article.

COMPETING INTERESTS

The authors declare that they have no competing interests.

REFERENCES

- [1] L. Ni, Y. Liu, and Y. Liu, "Privacy protection model for location-based services," *J. Inf. Process. Syst.*, vol. 16, no. 1, pp. 96–112, 2020, doi: 10.3745/JIPS.04.0163.
- [2] H. Li, X. Xue, Z. Li, L. Li, and J. Xiong, "Location privacy protection scheme for LBS in IoT," *Wireless Commun. Mobile Comput.*, vol. 2021, pp. 1–18, Aug. 2021, doi: 10.1155/2021/9948543.
- [3] J. Son, D. Kim, R. Tashakkori, A. O. Tokuta, and H. Oh, "A new mobile online social network based location sharing with enhanced privacy protection," in *Proc. 25th Int. Conf. Comput. Commun. Netw. (ICCCN)*, Waikoloa, HI, USA, 2016, pp. 1–9.
- [4] H. Li, H. Zhu, S. Du, X. Liang, and X. Shen, "Privacy leakage of location sharing in mobile social networks: Attacks and defense," *IEEE Trans. Dependable Secure Comput.*, vol. 15, no. 4, pp. 646–660, Jul. 2018.
- [5] W. He, "Research on LBS privacy protection technology in mobile social networks," in *Proc. IEEE 2nd Adv. Inf. Technol., Electron. Autom. Control Conf. (IAEAC)*, Chongqing, China, Mar. 2017, pp. 73–76.
- [6] Y. Luo, W. Li, T. Zhao, X. Yu, L. Zhang, G. Li, and N. Tang, "Deep-Track: Monitoring and exploring spatio-temporal data: A case of tracking COVID-19," *Proc. VLDB Endowment*, vol. 13, no. 12, pp. 284–2844, 2020.
- [7] X. Niu, Y. Yue, X. Zhou, and X. Zhang, "How urban factors affect the spatiotemporal distribution of infectious diseases in addition to intercity population movement in China," *ISPRS Int. J. Geo-Inf.*, vol. 9, no. 11, p. 615, Oct. 2020.

- [8] Q. Shi, D. Dorling, G. Cao, and T. Liu, "Changes in population movement make COVID-19 spread differently from SARS," *Social Sci. Med.*, vol. 255, Jun. 2020, Art. no. 113036.
- [9] L. Sweeney, "K-anonymity: A model for protecting privacy," *Int. J. Uncertainty, Fuzziness Knowl.-Based Syst.*, vol. 10, no. 5, pp. 557–570, Oct. 2002, doi: [10.1142/S0218488502001648](https://doi.org/10.1142/S0218488502001648).
- [10] A. Machanavajjhala, D. Kifer, J. Gehrke, and M. Venkatasubramanian, "L-diversity: Privacy beyond K-anonymity," *ACM Trans. Knowl. Discovery Data*, vol. 1, no. 1, p. 3, Mar. 2007, doi: [10.1145/1217299.1217302](https://doi.org/10.1145/1217299.1217302).
- [11] N. Li, T. Li, and S. Venkatasubramanian, "T-closeness: Privacy beyond K-anonymity and L-diversity," in *Proc. IEEE 23rd Int. Conf. Data Eng.*, Istanbul, Turkey, Apr. 2007, pp. 106–115.
- [12] O. Temuujin, J. Ahn, and D.-H. Im, "Efficient L-diversity algorithm for preserving privacy of dynamically published datasets," *IEEE Access*, vol. 7, pp. 122878–122888, 2019.
- [13] M. Jeon, O. Temuujin, J. Ahn, and D.-H. Im, "Distributed L-diversity using spark-based algorithm for large resource description frameworks data," *J. Supercomput.*, vol. 77, no. 7, pp. 7270–7286, Jul. 2021.
- [14] M. Ul Hassan, M. H. Rehmani, and J. Chen, "Differential privacy in blockchain technology: A futuristic approach," *J. Parallel Distrib. Comput.*, vol. 145, pp. 50–74, Nov. 2020, doi: [10.1016/j.jpdc.2020.06.003](https://doi.org/10.1016/j.jpdc.2020.06.003).
- [15] M. A. Husnoo, A. Anwar, R. K. Chakraborty, R. Doss, and M. J. Ryan, "Differential privacy for IoT-enabled critical infrastructure: A comprehensive survey," *IEEE Access*, vol. 9, pp. 153276–153304, 2021, doi: [10.1109/ACCESS.2021.3124309](https://doi.org/10.1109/ACCESS.2021.3124309).
- [16] P. Zhao, G. Zhang, S. Wan, G. Liu, and T. Umer, "A survey of local differential privacy for securing Internet of Vehicles," *J. Supercomput.*, vol. 76, no. 11, pp. 8391–8412, Nov. 2020, doi: [10.1007/s11227-019-03104-0](https://doi.org/10.1007/s11227-019-03104-0).
- [17] J. Kang, S. Jeong, D. Hong, and C. Seo, "A study on synthetic data generation based safe differentially private GAN," *J. Korea Inst. Inf. Secur. Cryptol.*, vol. 30, no. 5, pp. 945–956, 2020.
- [18] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, and Y. Bengio, "Generative adversarial nets," in *Proc. 27th Int. Conf. Neural Inf. Process. Syst.*, Montreal, QC, Canada, vol. 2, 2014, pp. 2672–2680.
- [19] M. Mirza and S. Osindero, "Conditional generative adversarial nets," 2014, *arXiv:1411.1784*.
- [20] J. Yoon, L. N. Drumright, and M. van der Schaar, "Anonymization through data synthesis using generative adversarial networks (ADS-GAN)," *IEEE J. Biomed. Health Informat.*, vol. 24, no. 8, pp. 2378–2388, Aug. 2020.
- [21] R. Indhumathi and S. S. Devi, "Healthcare Cramér generative adversarial network (HCGAN)," *Distrib. Parallel Databases*, vol. 40, no. 4, pp. 657–673, Dec. 2022, doi: [10.1007/s10619-021-07346-x](https://doi.org/10.1007/s10619-021-07346-x).
- [22] E. Piacentino, A. Guarner, and C. Angulo, "Generating synthetic ECGs using GANs for anonymizing healthcare data," *Electronics*, vol. 10, no. 4, p. 389, Feb. 2021.
- [23] N.-D.-T. Tieu, H. H. Nguyen, H.-Q. Nguyen-Son, J. Yamagishi, and I. Echizen, "Spatio-temporal generative adversarial network for gait anonymization," *J. Inf. Secur. Appl.*, vol. 46, pp. 307–319, Jun. 2019.
- [24] Z. Xiong, Z. Cai, Q. Han, A. Alrawais, and W. Li, "ADGAN: Protect your location privacy in camera data of auto-driving vehicles," *IEEE Trans. Ind. Informat.*, vol. 17, no. 9, pp. 6200–6210, Sep. 2021.
- [25] J. Rao, S. Gao, Y. Kang, and Q. Huang, "LSTM-TrajGAN: A deep learning approach to trajectory privacy protection," in *Proc. 11th Int. Conf. Geographic Inf. Sci. (GIScience)*, vol. 2021, p. 12.
- [26] J. Shin, Y. Song, J. Ahn, T. Lee, and D.-H. Im, "TCAC-GAN: Synthetic trajectory generation model using auxiliary classifier generative adversarial networks for improved protection of trajectory data," in *Proc. IEEE Int. Conf. Big Data Smart Comput. (BigComp)*, Jeju, South Korea, Feb. 2023, pp. 314–315.
- [27] J. W. Kim and B. Jang, "Deep learning-based privacy-preserving framework for synthetic trajectory generation," *J. Netw. Comput. Appl.*, vol. 206, Oct. 2022, Art. no. 103459.
- [28] Z. Zhang, X. Xu, and F. Xiao, "LGDAN-DP: A novel differential private publication mechanism of trajectory data," *Future Gener. Comput. Syst.*, vol. 141, pp. 692–703, Apr. 2023.
- [29] D. Yang, D. Zhang, V. W. Zheng, and Z. Yu, "Modeling user activity preference by leveraging user spatial temporal characteristics in LBSNs," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 45, no. 1, pp. 129–142, Jan. 2015.
- [30] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997.
- [31] J. Ribera, D. Guera, Y. Chen, and E. J. Delp, "Locating objects without bounding boxes," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Long Beach, CA, USA, Jun. 2019, pp. 6479–6489.
- [32] W. Li, Z. Liang, P. Ma, R. Wang, X. Cui, and P. Chen, "Hausdorff GAN: Improving GAN generation quality with Hausdorff metric," *IEEE Trans. Cybern.*, vol. 52, no. 10, pp. 10407–10419, Oct. 2022, doi: [10.1109/TCYB.2021.3062396](https://doi.org/10.1109/TCYB.2021.3062396).



YEJI SONG received the B.S. degree in computer engineering from Hoseo University, Asan, in 2021. She is currently pursuing the combined M.S./Ph.D. degree with the Department of Artificial Intelligence Convergence, Kwangwoon University, Seoul, South Korea. Her research interests include data privacy and machine learning.



JIHWAN SHIN received the B.S. degree in computer engineering from Hoseo University, Asan, South Korea, in 2022. He is currently pursuing the M.S. degree with the Department of Artificial Intelligence Applications, Kwangwoon University, Seoul, South Korea. His research interests include data privacy (differential privacy) and machine learning.



JINHYUN AHN received the B.S. and M.S. degrees in computer science education from Korea University, Seoul, South Korea, in 2005 and 2008, respectively, and the Ph.D. degree in computer science from Seoul National University, Seoul, in 2017. He is currently an Associate Professor with the Department of Management Information Systems, Jeju National University, South Korea. His research interests include distributed/parallel computing, knowledge engineering, data privacy, and graph processing.



TAEWHI LEE received the B.S. and Ph.D. degrees in computer science and engineering from Seoul National University, in 2004 and 2014, respectively. He is currently a Principal Researcher with the Electronics and Telecommunications Research Institute, Daejeon, South Korea. His research interests include large-scale data processing and approximate query processing using machine learning models.



DONG-HYUK IM received the B.S. degree in computer science from Korea University, in 2003, and the M.S. and Ph.D. degrees from the School of Computer Science and Engineering, Seoul National University, in 2005 and 2011, respectively. He is currently a Professor with the School of Information Convergence, Kwangwoon University, South Korea. His research interests include big data processing and analysis, data privacy, and machine learning.

• • •