

Differentially Private GANs for Generating Synthetic Indoor Location Data

Vahideh Moghtadaiee^{a,1}, Mina Alishahi², Milad Rabiei³

¹Cyberspace Research Institute, Shahid Beheshti University, Tehran, Iran

²Department of Computer Science, Open Universiteit, The Netherlands

³School of Electrical and Computer Engineering, Shahid Beheshti University, Tehran, Iran

Received: date / Accepted: date

Abstract The advent of location-based services has led to the widespread adoption of indoor localization systems, which enable location tracking of individuals within enclosed spaces such as buildings. While these systems provide numerous benefits such as improved security and personalized services, they also raise concerns regarding privacy violations. As such, there is a growing need for privacy-preserving solutions that can protect users' sensitive location information while still enabling the functionality of indoor localization systems. In recent years, Differentially Private Generative Adversarial Networks (DPGANs) have emerged as a powerful methodology that aims to protect the privacy of individual data points while generating realistic synthetic data similar to original data. DPGANs combine the power of generative adversarial networks (GANs) with the privacy-preserving technique of differential privacy (DP). In this paper, we introduce an indoor localization framework employing DPGANs in order to generate privacy-preserving indoor location data. We evaluate the performance of our framework on a real-world indoor localization dataset and demonstrate its effectiveness in preserving privacy while maintaining the accuracy of the localization system.

Keywords Generative adversarial network · Indoor location fingerprinting · Differential Privacy · Private GAN

1 Introduction

In recent years, IoT devices have become increasingly popular and have been developing quickly for smart cities, homes, health, agriculture, and factories [1], [2].

^ae-mail: v_moghtadaiee@sbu.ac.ir

It has become extremely common for people to use indoor location-based services (LBSs) employing their smart devices [3]. Today, most people spend 80% of their time in indoor areas while they actively use their smart devices. Three in four users of smart devices (74%) utilize location-based applications [4]. To provide location-based services to users, such as guiding them through unfamiliar environments like hospitals, shopping malls, airports, and railway stations, indoor LBSs must rely on precise indoor localization. As a result, suitable indoor localization is essential for indoor LBSs to function effectively [3].

The most popular technique of indoor localization is location fingerprinting, which consists of two phases, training and localization [5]. In the training phase, the indoor localization system is trained with the fingerprints of various locations in the desired indoor area. The fingerprints here are referred to features of locations, such as received signal strength (RSS) from existing signal emitters. In the localization phase, the fingerprints of users are given to the formerly trained model as input so that it could monitor and localize them in that area.

Indoor localization systems can be location-based or zone-based. Location-based localization provides (x, y) coordinates, while zone-based localization gives a zone. A zone in an indoor area is referred to a room, corridor, or any smaller part of the whole indoor environment. Indoor localization systems often require accurate location data to provide useful services in various applications, such as navigation, asset tracking, security monitoring, and other location-based recommendations. However, the collection and use of this data raise various privacy and security concerns as it can reveal information about the user's activities, habits, and movements, which can be used to infer personal information about the user [6]. This information can be then used for targeted adver-

tising or other purposes without the user’s knowledge or consent. In indoor localization systems, user privacy can be violated in several ways: (1) *Location Tracking*: Indoor localization systems track the location of a user inside a building. This location data can be sensitive and can reveal a user’s activities, habits, and interests. If this data falls into the wrong hands, it can be used to track and monitor users without their knowledge or consent [3]. (2) *Identification*: Indoor localization systems can also be used to identify individual users based on their unique location patterns [7]. For example, if a user’s location data is collected over a period of time, their daily routine and habits can be identified. This can lead to targeted marketing, personal profiling, and even discrimination. (3) *Re-identification*: Even if the user’s identity is not initially known, it may be possible to re-identify the user by combining the location data with other information [7]. For example, if an attacker has access to social media data or other publicly available information, they may be able to link the location data with other data sources to re-identify the user. (4) *Inference*: In some cases, the location data can reveal sensitive information about the user even without revealing their exact location [8]. For example, if a user frequently visits a specific medical facility, this may reveal sensitive information about their health condition.

There are ongoing efforts to develop techniques for protecting the privacy of individuals while still enabling the use of location-based services in indoor environments, such as anonymization, encryption, differential privacy (DP), and federated learning [7], [9], [10], [6], [11]. However, one of the most attractive solutions for protecting data privacy and mitigating possible attacks that have not been considered for indoor location privacy is to employ generated synthetic data instead of real data for training Artificial Intelligence (AI) models and analyzing data [12]. This can be done by Generative adversarial networks (GANs) [13]. These networks can learn the training data distribution and generate synthetic data with a distribution very similar to the training data [14]. They are a way of creating a generative model through competition between two neural networks. The generator network takes in random noise as input and transforms it into synthetic data that resembles real data. The discriminator network then tries to determine whether the data it receives is real or generated by the generator. The generator network is trained through its competition with the discriminator to produce more and more convincing synthetic data [13]. The GAN framework is a highly effective learning model that has been utilized for various applications, including but not limited to imitating expert policies and domain transfer [15].

Similar to other types of machine learning, GAN frameworks are also vulnerable to information leakage. In particular, the generator model tries to predict the underlying distribution of a dataset and produces realistic examples at random. This implies that the generator has the ability to recall training samples by means of deep neural networks. As a result, if the GAN model is employed on a private or confidential dataset, it may potentially expose the privacy of the dataset. To confront this issue, a potent solution, the GAN-obfuscator, is introduced in [16]. This framework employs a differentially private GAN (DPGAN) approach that introduces well-crafted Gaussian noise to the learning model gradients during the training phase. With the GAN-obfuscator in use, limitless synthetic data can be produced for any task, all while safeguarding the privacy of the training data [15].

Despite the fact that private GANs are getting increasingly popular, their effectiveness on indoor location data remains unclear, which is the main focus of this work. The main contributions of this paper are listed as follows:

- To the best of our knowledge, this is the first work that introduces DPGANs for generating private indoor location data for both location-based and zone-based indoor localization.
- Our proposed DPGAN framework for indoor localization not only preserves the privacy of indoor location data but also enhances the accuracy of localization at the same time.
- We investigate the influence of two popular DPGANs for indoor location synthetic data, Differentially Private Wasserstein GAN (DPWGAN) and Differentially Private Conditional GAN (DPCGAN), and analyze the similarity of the generated datasets to the original dataset, the localization accuracy, and the privacy issues.
- The efficiency and the performance of the suggested DPGAN framework for indoor localization are verified via a real-world experimental testbed.

The rest of the paper is organized as follows. Section 2 discusses related studies on users’ privacy preservation methods in indoor areas. In Section 3, the preliminary concepts about indoor fingerprinting systems, DP, GANs and two DP-based GANs are explained. The suggested framework for implementing DPareGANs for indoor location data is presented in Section 4. The experimental setup, evaluation metrics, and model parameters is discussed in 5. Section 6 is devoted to the performance evaluation of the proposed framework employing DP-GANs for generating privacy-preserving indoor location data and Section 7 concludes the paper.

2 Related Work

It has been less than a decade that indoor location data privacy has been getting more attention. Authors in [17], summarize some of the studies on the privacy of users in indoor areas. A number of techniques have been developed to counter threats to indoor location privacy such as anonymization techniques, cryptography, differential privacy (DP) and local differential privacy (LDP), and federated learning (FL). Here, we briefly summarize existing studies using each of these techniques.

Among anonymization techniques, k -anonymity is mostly employed for indoor location privacy protection. It hides the user's identity from the attackers in $k - 1$ other users. Employing other anonymization approaches, *e.g.*, ℓ -diversity, t -closeness, (α, k) -anonymity and δ -presence, has been recently suggested in [7] to mitigate the attacks for k -anonymity.

Cryptography is also utilized in indoor localization [9], [18], in which where users apply homomorphic encryption on their RSS vectors and then send them to the server. Authors in [19], on the other hand, deploy secure multi-party mechanisms and combine two secure two-party computing protocols to develop PPIL circuits construction components. The problems with these two techniques are heavy computation and communication overhead as users are mostly moving and their positions are continuously updated [20].

In addition to the aforementioned techniques, DP is being increasingly used to guarantee privacy regardless of adversary background knowledge [21]. Since DP is basically added to the RSS vectors of users, it leads to a considerable localization error. Recently, Adp-FSELM scheme is proposed in [10] employing differentially private Wi-Fi and Bluetooth fingerprints fusion with semi-supervised extreme learning machine for indoor localization via a trusted aggregation in edge nodes, which also needs the participation of the high number of people. In order to mitigate the problems of DP, such as requiring a trusted server and a large number of data points, LDP is introduced for indoor positioning systems in [22], [23], where no trusted server is necessary. Specifically, the combination of k -anonymity and LDP has suggested a paradigm-driven LDP framework for indoor localization in [24]. LDP in indoor positioning, nevertheless, requires a large number of participants, while there might not be that number of users in indoor areas.

Furthermore, authors in [6] propose indoor Geo-indistinguishability, which brings Geo-indistinguishability (that was originally introduced for outdoor positioning [25]) into indoor localization. It is a DP-based privacy-preserving framework in a Geo-indistinguishability setting that protects the indoor location privacy of users by

obfuscating the location data before leaving the users' devices.

Finally, FL is also applied as a recently proposed privacy-preserving mechanism in indoor localization [11], [26]. Using FL provides privacy advantages because it is distributed and requires clients to train local models instead of transferring data to the central server [27]. The authors in [28] present FLOC utilizing a homomorphic encryption-based federated learning framework. Another work proposes personalized federated learning due to the fact that the fingerprint data is actually non-IID data as users usually move around in the indoor areas [29]. However, there is a high risk of getting users' information if attackers can access the weights of the FL-based system. Authors in [30] try to decrease this risk by applying DP on the model weights. In addition, employing FL is usually feasible when the indoor positioning system is trained based on the crowdsourcing technique, in which users are constantly involved in training to improve the localization model, which is not always the case.

None of the aforementioned papers considers the applicability of DPGANs in indoor environments, in which there is no need for another trusted party and the presence of any other participants. Moreover, unlike other privacy-preserving mechanisms, DPGANs enable indoor positioning systems to be simultaneously private and accurate.

3 Preliminaries

In this section, preliminary concepts used in this study including (*i*) location fingerprinting in indoor environments, (*ii*) DP, (*iii*) GAN descriptions including WGAN and CGAN, and (*iv*) two of commonly-used DPGANs (DPWGAN and DPCGAN) are explained.

3.1 Fingerprinting Localization

The location fingerprinting technique is inspired by the unique fingerprinting of people. Just as every individual has their own specific fingerprint, every location has also its own particular signal features. Depending on what kind of signal(s) we are receiving at each location, the signal features can be the RSS values. The most popular type of signal used in indoor localization is Wi-Fi signals.

Indoor fingerprinting is the process of creating a map of signal strengths or other features detected by sensors located at different points within an indoor environment. The purpose of this map is to estimate the location of a user's device based on the signals it receives from these sensors. Fingerprinting localization, in general,

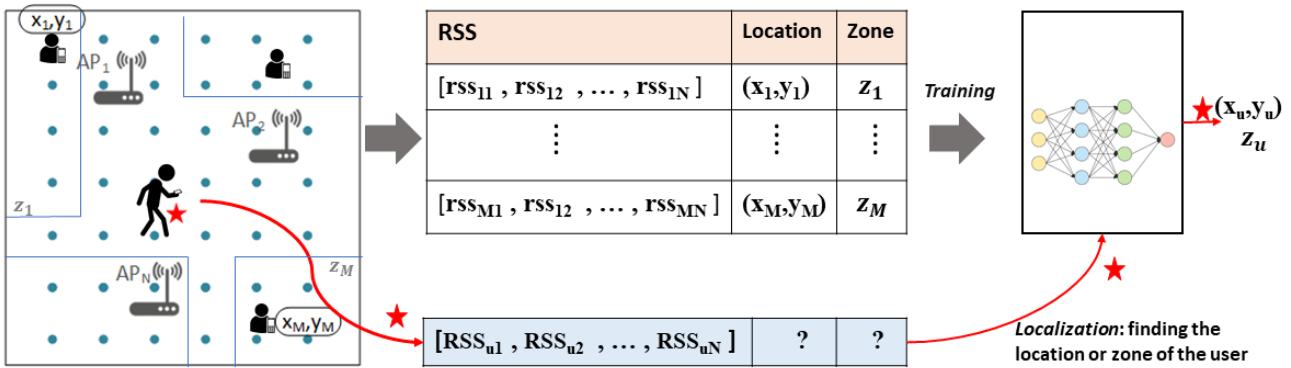


Fig. 1: The general overview of indoor location fingerprinting learning and localization processes.

involves two stages: training and localization. Both of these phases make use of existing Wi-Fi Access Points (APs) as signal sources. Any device that is capable of detecting Wi-Fi signals can be utilized as a receiver [31]. The indoor positioning system is mainly trained by conducting surveys of the targeted area or getting huge amounts of data from people around that area to identify significant characteristics. The RSS values collected from existing APs at the known Reference Points (RPs) along with the (x, y) coordinates of the RPs are stored in the radiomap on the server.

The recorded radiomap used for the AI training has a $M \times N$ dimension, where M and N are the numbers of RPs and APs, respectively. The fingerprint at a location (x_i, y_i) is denoted by $rssi^{RP} = [rss_{i1}, rss_{i2}, \dots, rss_{iN}]$, where $rssi_{ij}$ is the RSS of AP_j at the i th RP. The AI model is exploited after the training. Then in the online phase, the location fingerprint of the user is injected as an input to the AI model so that the localization of the user can be carried out and LSP can estimate the position of the user, (x, y) . The user's fingerprint is obtained as the vector of RSS values, $S_u = [RSS_{u1}, RSS_{u2}, \dots, RSS_{uN}]$, where RSS_{uj} is the RSS of AP_j received by the user. Machine learning algorithms for fingerprinting localization include traditional machine learning (ML), deep learning (DL), and deep reinforcement learning algorithms [32]. Figure 1 illustrates the indoor fingerprinting localization system with training/localization phases.

3.2 Differential Privacy

Differential privacy (DP) [33] is commonly recognized as the go-to privacy analysis method. Depending on the degree of trust in the server, DP can be enforced either in a *local* or *global* context. By adding a suitable

amount of noise to real-time or statistical data, DP ensures that it stays confidential while preserving an acceptable equilibrium between precision and privacy. In DP, if two datasets differ in at most one data point (one row), they are called *adjacent*. Let privacy budget, $\varepsilon \in \mathbb{R}_{\geq 0}$, and f be an algorithm operating on datasets. We say f satisfies ε -Differential Privacy if for adjacent datasets D, D' and all sets of possible outputs M , we have:

$$\mathbb{P}(f(D) \in M) \leq e^\varepsilon \mathbb{P}(f(D') \in M). \quad (1)$$

By ensuring that the probability distributions on the output space originating from two input datasets cannot differ too much, ε -DP provides plausible deniability about any row's true value, even if all other rows are compromised. The lower ε , the stronger privacy ε -DP guarantees. To ensure privacy in the regressor/classifier learning setting, we demand that a regressor/classifier training algorithm satisfies ε -DP.

(ε, δ) -differential privacy [34] A randomized mechanism M is (ε, δ) -differential private if for any set S and any two neighbor databases D and D'

$$P(M(D) \in S) \leq e^\varepsilon P(M(D') \in S) + \delta \quad (2)$$

The Gaussian mechanism has been widely used to achieve (ε, δ) -DP. In this model, for a function $f : D \rightarrow R$, its L_2 sensitivity is defined as $\Delta_2 f = \max_{D, D'} \|f(D) - f(D')\|_2$. This function satisfies (ε, δ) -DP, if Gaussian noise is added to f 's output, i.e., $M(D) = f(D) + \mathcal{N}(0, (\sigma \Delta_2 f)^2)$ for $\varepsilon < 1$ and $\sigma > \sqrt{2 \ln 1.25 / \delta} / \varepsilon$.

Rényi differential privacy [35] is a privacy framework that quantifies the privacy guarantee provided by a randomized algorithm in terms of the divergence between the distribution of the output of the algorithm on neighboring inputs. Formally, a randomized mechanism

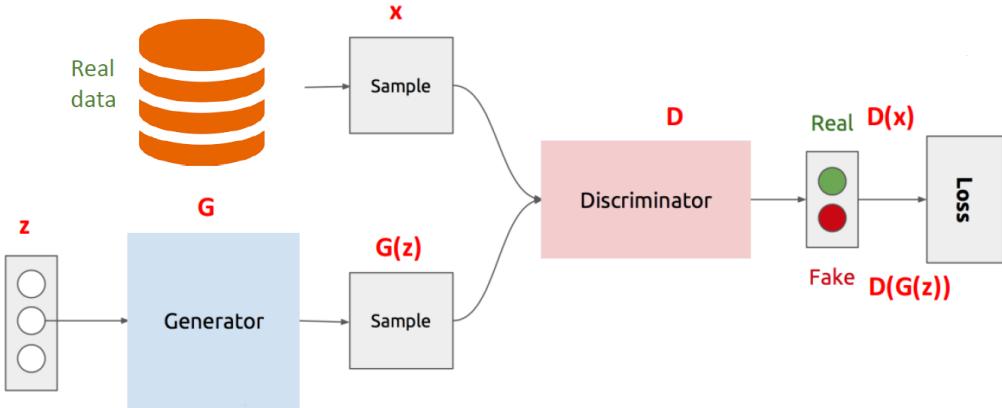


Fig. 2: The schematics of a GAN

M satisfies ε -Rényi differential privacy (ε -RDP) if for all pairs of neighboring datasets D, D' and all $\alpha > 0$, the following inequality holds:

$$\ln \left(\mathbb{E}[e^{\alpha M(D)}] \right) - \ln \left(\mathbb{E}[e^{\alpha M(D')}] \right) \leq \varepsilon \frac{\alpha^2}{2} \quad (3)$$

where $\mathbb{E}[e^{\alpha M(D)}]$ and $\mathbb{E}[e^{\alpha M(D')}]$ denote the expected value of the exponential mechanism applied to datasets D and D' , respectively. This definition means that for any pair of neighboring datasets, the probability of any output from the mechanism M on the first dataset being generated is at most $e^{\varepsilon\alpha^2/2}$ times greater than the probability of the same output being generated on the second dataset.

3.3 GANs

Generative Adversarial Networks (GANs), introduced by Goodfellow et al. in 2014 [13], are employed to generate synthetic data with maximum similarity to a given dataset by learning its features' distribution over time using a Minmax game. As it is shown in Figure 2, there are two neural network models in GANs: a *Generator*, G , which contains a generative model for generating synthetic data and a *Discriminator*, D , which is an adversarial model for detecting fake data. First, a Generator creates synthetic data using noise inputs drawn from a "latent space". A Discriminator, on the other hand, gets both original data and this Generator data as inputs and classifies the Generator data as fake or real by comparing them with the original data. This process results in a loss, which is used to optimize both Generator and Discriminator. Hence, there is quite a battle between Generator and Discriminator. The Generator tries to fool the Discriminator by learning to produce synthetic data that cannot be distinguishable from real

ones, while the Discriminator tries to discriminate the authenticity of data by calculating the probability that input comes from real data instead of the Generator. The mathematical model on the Minmax game in GAN can be formulated as below to minimize the performance of G while maximizing the performance of D :

$$\min_G \max_D \mathbb{E}_{\mathbf{x} \sim p_{\text{data}}(\mathbf{x})} [\log D(\mathbf{x})] + \mathbb{E}_{\mathbf{z} \sim p_z(\mathbf{z})} [1 - \log D(G(\mathbf{z}))] \quad (4)$$

where \mathbf{x} and \mathbf{z} are sets of random samples from the data probability distribution $p_{\text{data}}(\mathbf{x})$ and the noise probability distribution $p_z(\mathbf{z})$ (the *prior*), respectively. The first term indicates how well D works on the real data and the second term determines how bad D works on the generated data.

The Generator and Discriminator are trained using a loss function that measures the difference between the Generator's output and the true labels (real or fake) of the input data. Typically, the Generator and Discriminator use different loss functions. During the training process, the Generator and Discriminator are trained iteratively in a loop. In each iteration, the Generator generates synthetic data and the Discriminator classifies the input data as real or fake. The loss function is used to update the weights of the Generator and Discriminator, and the process is repeated for a fixed number of epochs. There are also different variations of GANs, such as Wasserstein GAN (WGAN) [36] and Conditional GAN (CGAN) [37].

Wasserstein GAN (WGAN) has proposed to improve the training stability by minimizing the distance between p_z and p_{data} through the following objective function:

$$\min_G \max_{w \in W} \mathbb{E}_{x \sim p_{\text{data}}} [f_w(x)] - \mathbb{E}_{z \sim p_z} [f_w(G(z))] \quad (5)$$

where $\{f_w\}_{w \in W}$ represent a family of K-Lipschitz functions for some constant K , that is $\|f\|_{Lip} \leq K$.

Conditional GAN (CGAN) allows both Generator and Discriminator to be conditional on the side of class labels, denoted by y , such that CGAN generates synthetic data for a given class. The objective of CGAN is computed as:

$$\min_G \max_D \mathbb{E}_{(\mathbf{x}, \mathbf{y}) \sim p_{xy}} [\log D(x, y)] + \mathbb{E}_{\mathbf{z} \sim p_z, \mathbf{y} \sim p_y} [\log(1 - D(G(z, y), y))] \quad (6)$$

where p_{xy} is the joint distribution from real samples x and class labels y , and p_y is the label distribution.

3.4 DPGANs System Model

DPGANs can provide a way to balance privacy and accuracy by generating synthetic data that is statistically similar to the real data but does not reveal specific information about any specific data points. This can enable LBS to provide accurate services without compromising user privacy. Therefore, here we provide details on two re-known DPGANs, DPWGAN and DPCGAN, which are evaluated in this paper as the benchmark synthesizers for generating private indoor location data.

Following work in [38], a number of studies utilized DP-SGD and GANs to generate differential private synthetic data [39], [14]. In DPGANs, the DP mechanism is used to add noise to the gradients of the Generator during the training process to protect the privacy of the training data. The privacy budget in DP refers to the amount of noise that is added to the data, which is controlled by ε . In general, as the value of ε decreases, the amount of noise added to the data increases, which can affect the quality of the generated data.

3.4.1 DPWGAN

The DPWGAN [39] model enforces DP by incorporating noise into the Discriminator during training. Since differential privacy guarantees post-processing privacy, privatizing the Discriminator results in DP being enforced on the parameters of the Generator. This is because the mapping function between the Generator and Discriminator does not involve any private data. DPWGAN is based on WGAN and adds noise to the gradients while only clipping the model weights to maintain the network's Lipschitz property. These models have been tested on image data and electronic health records in previous studies.

Formally, a batch of prior samples $\{z^{(i)}\}_{i=1}^m \sim p(z)$ and a batch of real data points $\{z^{(i)}\}_{i=1}^m \sim p_{data}(x)$ are

sampled. For each i , the gradients of the Discriminator are updated as follows:

$$g_w(x^{(i)}, z^{(i)}) \leftarrow \nabla_w [f_w(x^{(i)}) - f_w(g_\theta(z^{(i)}))] \quad (7)$$

the Gaussian noise is then added to the average of gradients as follows:

$$\bar{g}_w \leftarrow \frac{1}{m} \left(\sum_{i=1}^m g_w(x^{(i)}, z^{(i)}) + \mathcal{N}(0, \sigma_n^2 c_g^2 I) \right) \quad (8)$$

and the weights on the next iteration are updated as below:

$$w^{(t+1)} \leftarrow w^{(t)} + \alpha_d \cdot RMSProp(w^{(t)}, \bar{g}_w), \quad (9)$$

$$w^{(t+1)} \leftarrow clip(w^{(t+1)}, -c_p, c_p) \quad (10)$$

The Generator then updates its gradients using the data received from a differentially private Discriminator. The network is as a whole differentially private due to the post-processing property of DP.

3.4.2 DPCGAN

DPCGAN [14] has been proposed to generate synthetic data as well as corresponding labels. In this setting, similar to DPWGAN, the differential private noise is added to the Discriminator's gradients based on the objective function of CGAN (Equation 6). DPCGAN has two innovations: 1) the training process splits the Discriminator loss between real data ($\log D(x, y)$) and generated data ($\log(1 - D(G(z, y), y))$); 2) RDP accountant is proposed to be applied instead of Moment accountant. Formally, the gradients of the Discriminator are updated when the Gaussian noise is added to them:

$$\frac{1}{m} \left(\sum_{i=1}^m g_w(x^{(i)}, z^{(i)}) + g_{w'}(x^{(i)}, z^{(i)}) + \mathcal{N}(0, \sigma_n^2 c_g^2 I) \right) \quad (11)$$

where g_w and $g_{w'}$ are respectively the gradients of real and fake data, respectively.

4 Indoor Location DPGAN framework

This section introduces the proposed indoor location framework based on DPGANs. The motivation, the methodology, and the utility/privacy analysis of the proposed framework are discussed here.

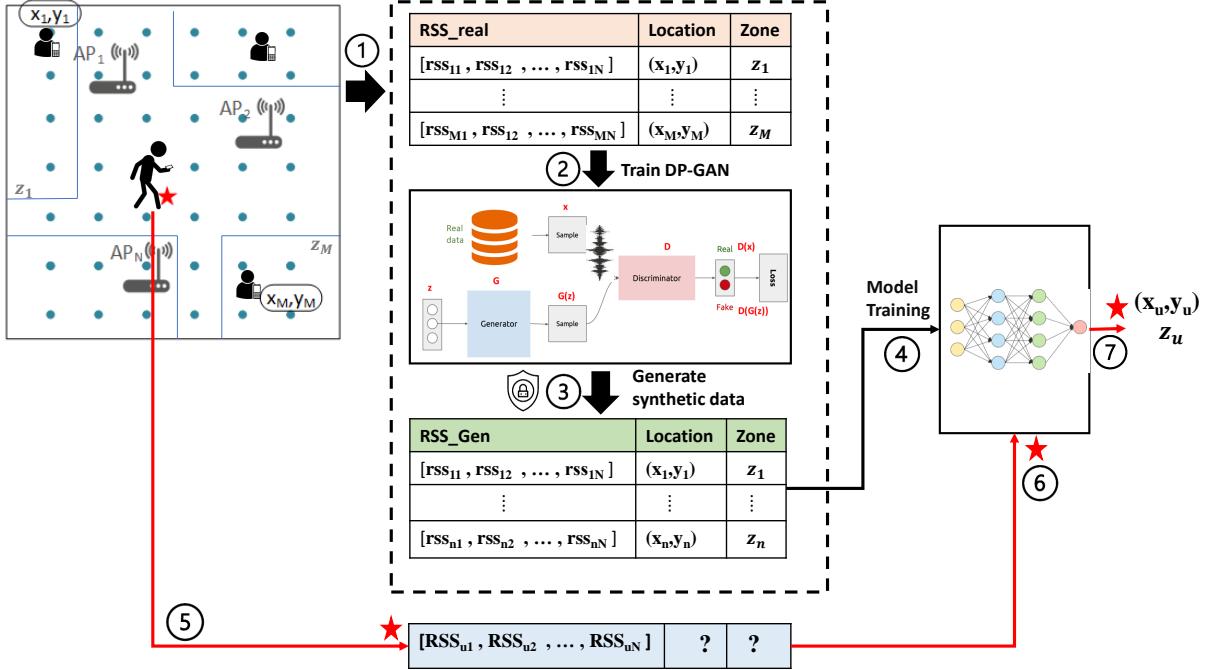


Fig. 3: The proposed indoor location DPGAN framework.

4.1 Motivation

The generated synthetic indoor location data can be used for a variety of tasks, such as testing and evaluating indoor positioning algorithms or simulating indoor environments for virtual reality applications. The generated data is differentially private, ensuring the privacy of individuals whose data was used to train the model, while still capturing the statistical patterns and relationships of the real-world data.

One possible motivation scenario for using DPGANs for indoor localization could be in a commercial shopping mall setting. Consider a shopping mall that intends to provide personalized recommendations to shoppers based on their current location and browsing history, but also needs to protect the shoppers' privacy. The mall has a network of beacons installed throughout the mall to track the location of shoppers, and collects this location data to build models for personalized recommendations. However, collecting the location data can be sensitive and can reveal information about the shoppers' activities and interests. To protect the shoppers' privacy while still providing accurate recommendations, the mall could use a variation of DPGANs to generate synthetic location data. The DPGAN could be trained on the real location data collected by the beacons and could generate synthetic data that is statistically similar

to the real data but does not reveal information about specific individuals. This synthetic data could be then used to build the recommendation models without compromising the shoppers' privacy. DPGANs enable the shopping mall to provide personalized recommendations to shoppers while protecting their privacy, which could lead to increased customer satisfaction and loyalty.

4.2 Methodology

The architecture of our indoor location DPGAN framework has been depicted in Figure 3 and it can be described in the following steps:

Step ①: The real location fingerprinting data is measured and gathered from people within a target indoor area with a crowdsourcing technique in order to create a dataset for that environment. For the location-based approach, the dataset contains RSS values and (x, y) coordinates, while for the zone-based approach, the dataset consists of RSS values and zones.

Step ②: The dataset collected in the previous step is used to train the DPGAN. This dataset is considered input data to the Discriminator network to build a DL model for DPGAN. The training process of DPGAN is dependent on whether the indoor localization system is location-based or zone-based. For location-based localization, DPWGAN is used and trained on

RSS values and (x, y) location coordinates, while for zone-based localization both DPWGAN and DPCGAN can be employed and trained on RSS values and zones.

Step ③: The trained location-based or zone-based DPGAN is employed for generating synthetic data similar to the original data. It can also generate more synthetic fingerprinting data in order to improve localization accuracy and preserve privacy at the same time.

Step ④: The synthetic dataset is used to train an ML/DL model for localization so that we could further localize new users in that environment. Based on the location-based or zone-based approaches of the indoor localization system, this model can be either a regressor or a classifier and the position of a new user, hence, is provided based on its (x, y) coordinates or its zone.

Step ⑤: A new user arrives, who requires to receive services based on her/his position. The fingerprints of her/his place are then collected by her/his own devices.

Step ⑥: The RSS fingerprints of a new user are given to the trained localization model.

Step ⑦: Resting on whether the indoor localization system is location-based or zone-based, the location coordinates (x_u, y_u) or zone (z_u) of a new user is computed.

4.3 Utility/Privacy Analysis

Basically, DPGANs achieve their privacy by adding noise to the gradient updates of the Generator and Discriminator during the training process. This noise then helps to ensure that the output of the Generator is not dependent on any single input data point, which makes it more difficult for an attacker to infer any information about an individual's data from the generated dataset. Moreover, DPGANs provide formal privacy guarantees in the form of DP, *i.e.*, they are designed to ensure that the probability distribution of the generated data does not change significantly when a single individual's data is removed from the original dataset. This means that an attacker would not be able to determine whether or not a particular individual's data was included in the original dataset based on the generated data alone. On the other hand, the Generator network of DPGAN is designed to learn the underlying distribution of the original data and to produce synthetic data that is similar to it. This means that the network generates data with a distribution very similar to that of original data, which results in obtaining accurate outcomes for location-based services purposes.

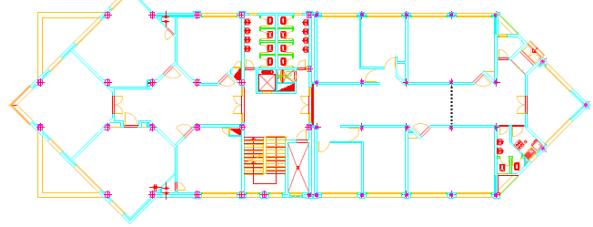


Fig. 4: The floor plan of the CRI testbed.

5 Experimental setup

This section explains the real-world dataset employed to test our framework, the evaluation metrics, and the hyper-parameters of the ML models utilized in the proposed framework.

5.1 Dataset

Here, we employed CRI dataset which is measured in a $51m \times 18m$ testbed situated on the second floor of the Cyberspace Research Institute (CRI) at Shahid Beheshti University. The layout can be seen in Figure 4. There are nine Wi-Fi APs installed on the ceiling. The radio signal strength values of 384 reference points were recorded from all nine access points in four directions, with 100 samples taken in each direction, and saved in the radiomap database. Additionally, within a short period of time, 100 inquiry samples were gathered at different times for the purpose of evaluating the experiment. All 400 samples at each point are averaged to get a more accurate fingerprint of them. The whole environment also consists of 15 zones including rooms and corridors. Therefore, the dataset has 12 columns including, 9 RSS values from all APs, the (x, y) coordinates of the points, and the zone number where the point is located. The details on this dataset can be found in [5].

5.2 Evaluation Metrics

Here the performance of location-based and zone-based DPWGAN and zone-based DPCGAN in indoor locations are evaluated and compared with the performance of non-private WGAN and CGAN from two perspectives: data utility and privacy evaluation.

In the utility assessment, generally, if synthetic data exhibit similar statistical characteristics and model accuracy as the original data, it is considered to have high utility. Therefore, the correlation between features in the original dataset is compared with feature correlations in

the synthetics datasets. Furthermore, we employed various machine learning models to explore the usefulness of generated datasets in AI training activities in both location-based and zone-based modes measuring Root Mean Square Error (RMSE) in meters and accuracy (%), respectively.

In the privacy assessment, we assessed the effectiveness of our framework in preventing the disclosure attack. DPWGan/DPCGAN is used to generate synthetic data based on the original data used to train them. Although it is expected that the records of real data will not appear in synthetic data, still we should be cautious about the possibility of accidental privacy leaks. This leakage might arise due to the unintentional memorization of some original records during training. To examine this matter, we can check for identical records in the generated datasets using a straightforward but practical approach based on Euclidean distance. To assess the potential memorization issue (disclosure risk), we measure the average Euclidean distance between the generated records and the closest records in the original dataset. This metric helps us determine how far a generated point is away from the original data points and whether this could result in privacy leakage.

5.3 Model Structure and Parameters

We have implemented all the experiments to evaluate our framework in Python. The experiments were performed on a machine with an Intel Corei7 1.8GHz CPU and 12GB of RAM. In order to assess the datasets' quality, we use scikit-learn tools and employed a group of four models (MLP, Random forest, SVM, and Decision tree). These models are trained on the datasets and their predictive performance and reliability are evaluated by testing their ability to accurately predict the labels. The learning rate here is 10^{-4} . Other parameters are the default parameters of the models in scikit-learn.

The training process also is based on the five-fold cross-validation. In other words, training the models involves dividing the generated dataset into five subsets, and then using each subset in turn for validation while training the model on the remaining four subsets. After training, the performance of the model is evaluated by testing it on the portion of the generated dataset, when having been also trained on a portion of the generated dataset.

6 Empirical Evaluation

In this section, we assess our indoor localization DPGAN framework from data utility and privacy points of view.

6.1 Utility Evaluation

The objective of the utility evaluation is to examine if the synthetic data produced by our framework maintains the correlations of the original data. To measure how the correlation between features in the original dataset is preserved compared to generated datasets, we compute the correlation matrix in terms of *Pearson correlation coefficient*. This metric quantifies the strength and direction of the linear association between two variables. Its value ranges from -1 to $+1$, with 0 indicating no correlation. A correlation coefficient of 1 indicates a perfect positive linear relationship between the two variables, while a correlation coefficient of -1 indicates a perfect negative linear relationship. The closer the correlation coefficient is to 0 , the weaker the linear relationship between the variables. Formally, the Pearson correlation coefficient between two random variables X and Y with means μ_X and μ_Y and standard deviations σ_X and σ_Y , respectively, is defined as:

$$\rho_{X,Y} = \frac{\text{cov}(X, Y)}{\sigma_X \sigma_Y} = \frac{E[(X - \mu_X)(Y - \mu_Y)]}{\sigma_X \sigma_Y} \quad (12)$$

where $\text{cov}(X, Y)$ is the covariance of X and Y , E is the expected value operator, and μ_X , μ_Y , σ_X , and σ_Y are the means and standard deviations of X and Y , respectively. In this study, we just consider the absolute values of correlation coefficients as the direction of correlation has no impact on our analysis. Thus, the coefficient values are reported in the range $[0, 1]$.

Here, the Pearson correlation coefficient is used to measure the mutual correlation between each pair of features in (i) the original dataset, (ii) datasets generated using WGAN and CGAN, and (iii) datasets generated using DPWGan and DPCGAN ($\epsilon = 1$). For DPWGan, we have tried both location-based and zone-based localization, but DPCGAN is used only for zone-based positioning. Figures 5, 6, and 7 depict the feature correlation results for location-based DPWGan, zone-based DPWGan, and zone-based DPCGAN, respectively. The darker color shows that there is a stronger correlation between the associated features (higher coefficient value).

It can be observed that, in general, the trend of correlation remains consistent across each set of experiments. This means that if a pair of features have a strong correlation in the original dataset (indicated by darker colors), that pair of features have still strong correlation in datasets generated by non-private WGAN/CGAN and DPWGan/DPCGAN ($\epsilon = 1$). This trend, however, has been better preserved when location-based WGAN/DPWGan has been used for generating synthetic data. This result suggests applying DPWGan for generating location-based indoor location data when

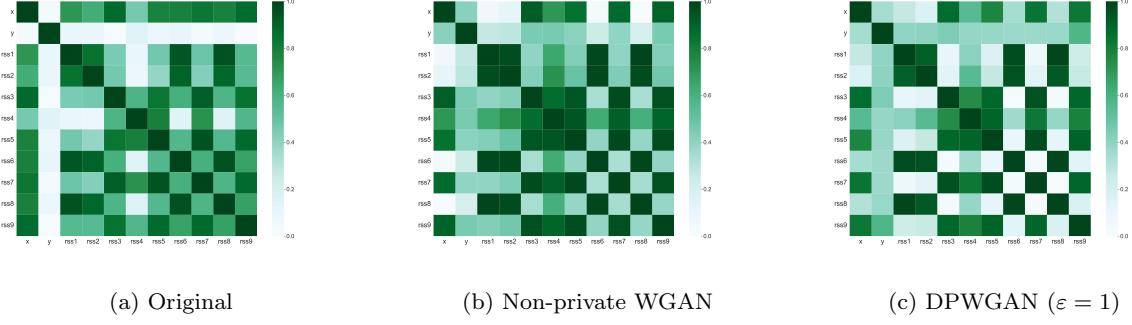


Fig. 5: Correlation results for location-based DPWGAN

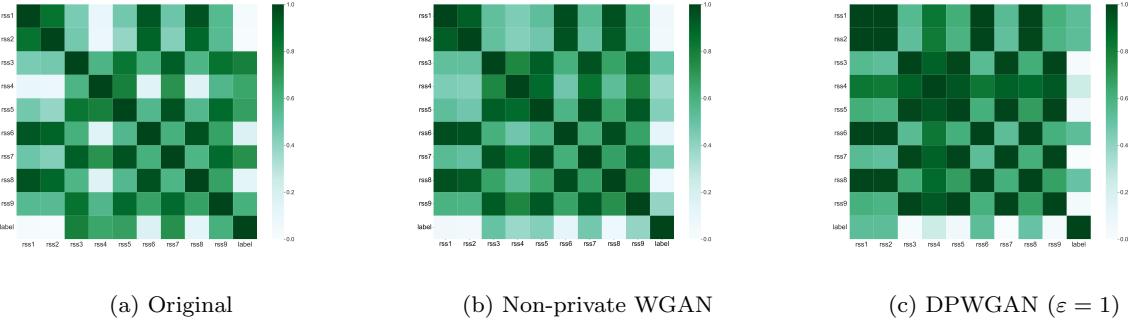


Fig. 6: Correlation results for zone-based DPWGAN

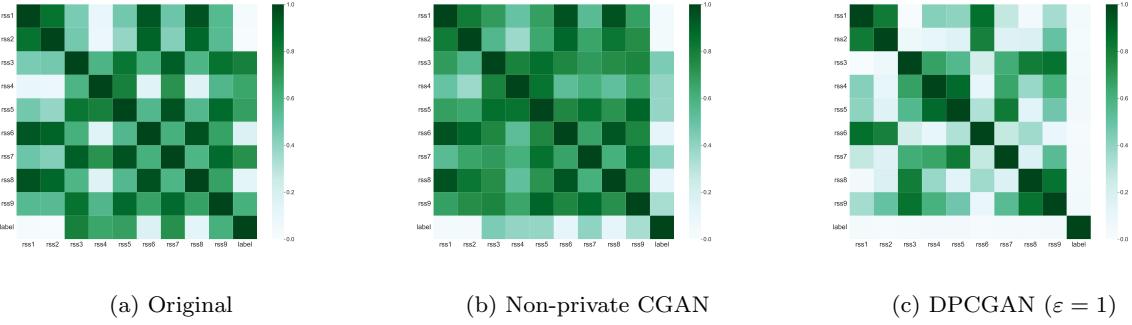


Fig. 7: Correlation results for zone-based DPCGAN

protecting the correlation between features is of crucial importance.

6.2 AI Training Performance

Similar to what is done in [8], in order to assess the usefulness of synthetic data in various AI training endeavors, a variety of ML algorithms are employed. Therefore, we use the original dataset in comparison with the generated dataset by non-private GANs and DPGANs to

train the ML model for step 4 of our framework. For a high utility private data generation, it is expected to achieve similar accuracy to the accuracy obtained with the real dataset while it is private at the same time. Using indoor location datasets, we can train both regressors and classifiers to evaluate the performance of the private synthetic dataset on them.

6.2.1 Regression results

In the case of regression training, we remove the zone number column from the datasets and report the locations in (x, y) coordinates. To evaluate the utility of our framework, we trained four regressors, namely MLP, Random Forest, SVM, and Decision Tree, on the original CRI dataset, and on datasets generated using non-private WGAN, and DPWGAN ($\varepsilon = 1, 5, 10, 15$) in a location-based mode. The utility has been measured based on the average distance in terms of RMSE (meters) between the real locations and those estimated by regressors trained on different synthetic data. Table 1 reports the RMSE results for the location-based localization. As it is expected, it can be seen that the utility of generated data reduces (RMSE increases), when more privacy constraint is required (lower ε value). This trend is independent of the selection of regressor. It also indicates that the RMSE obtained by applying MLP and SVM regressors on the generated dataset by DPWGAN with ε greater than 10 are very similar to the RMSE by the original dataset and the non-private WGAN.

Figure 8 depicts the RMSE of MLP regressor when it is trained on the original dataset (dashed black line) vs generated datasets by non-private WGAN and DPWGAN for $\varepsilon = 1, 5, 10, 15$ and $\delta = 10^{-5}$ based on the number of fingerprints in the dataset. The number of records in the original dataset is 384, while for the synthetic datasets, we tried to generate more fingerprint records so that we could improve the localization accuracy as well. The results show that the accuracy of the regressor has been improved when it has learned the pattern from a larger dataset. This is specifically considerable up to the point that the number of records reaches 1000. For a higher number of records, the accuracy of the indoor location fingerprinting system goes to the saturated phase which is compatible with [5]. Therefore, using DPWGAN not only provides a more private dataset but also decreases the localization error as more data could be generated for training. Even with more private synthetic datasets ($\varepsilon = 1, 5$), we can achieve the same or less localization error as the original dataset in a higher number of fingerprint samples.

6.2.2 Classification results

Here, classification models are applied for zone-based indoor localization. In this case, zone numbers are considered as labels for the ML models and the locations are reported in zone numbers, not in (x, y) coordinates. For classification purposes, thus, we remove (x, y) coordinates from the dataset and generate the RSS columns and labels. In DPWGAN, this is done by considering

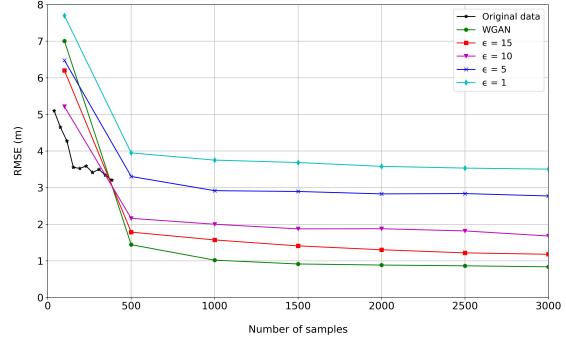


Fig. 8: The location-based RMSE of MLP regressor using original dataset vs generated datasets by non-private WGAN and DPWGAN with $\varepsilon = 1, 5, 10, 15$ and $\delta = 10^{-5}$, based on the number of fingerprint samples.

the label as another feature and generating all features (RSS and label) simultaneously. In DPCGAN, which is conditioned by labels, the model is trained on the data associated with each label separately.

We again consider four classifiers including MLP, Random Forest, SVM, and Decision Tree and apply the original CRI dataset, and datasets generated using non-private WGAN, DPWGAN ($\varepsilon = 1, 5, 10, 15$), non-private CGAN, and DPCGAN ($\varepsilon = 1, 5, 10, 15$) on them, all in zone-based localization mode. Table 2 and 3 list the zone-based accuracy of these four classifiers for DPWGAN and DPCGAN, respectively. As expected, in lower ε values when privacy constraints are higher, the utility of the generated data reduces and we obtain lower accuracy. This trend is consistent with all classifiers. Table 2 also specifies that the accuracies of MLP and SVM classifiers for the generated dataset by DPWGAN with ε greater than 10 are very similar to the accuracy by the original dataset and the non-private WGAN. Table 3, on the other hand, indicates the lower accuracies of generated datasets when the conditional form of GAN is employed even in non-private mode. The zone-based accuracy degrades more when the DPCGAN is taken into the account. It can be also inferred from these two Tables that in the zone-based localization mode, both non-private WGAN and DPWGAN show higher accuracy compared with the non-private CGAN and DPCGAN.

In order to evaluate the accuracy of a zone-based classifier for both DPWGAN and DPCGAN when the number of data points is increased, we consider the MLP classifier as it shows better accuracy in the last two Tables. Figure 9 illustrates the accuracy of zone-based MLP classifiers when it is trained on the original dataset (dashed black line) vs generated datasets by

Table 1: The location-based RMSE of four regressors when using original dataset vs generated dataset by non-private WGAN and DPWGIAN on $\epsilon = 1, 5, 10, 15$ and $\delta = 10^{-5}$ (for 384 records).

	RMSE (m)			
	MLP	Random Forest	SVM	Decision Tree
Original	3.28 ± 0.2	2.36 ± 0.4	3.17 ± 0.2	3.29 ± 0.8
WGAN (non-private)	3.51 ± 0.2	2.46 ± 0.2	2.29 ± 0.3	4.30 ± 0.2
DPWGIAN ($\epsilon = 15$)	3.38 ± 0.2	3.75 ± 0.3	2.66 ± 0.4	5.36 ± 0.2
DPWGIAN ($\epsilon = 10$)	3.46 ± 0.2	3.98 ± 0.1	3.79 ± 0.2	5.59 ± 0.1
DPWGIAN ($\epsilon = 5$)	4.22 ± 0.1	4.33 ± 0.1	3.81 ± 0.2	4.63 ± 0.1
DPWGIAN ($\epsilon = 1$)	4.97 ± 0.2	4.63 ± 0.1	5.02 ± 0.4	5.58 ± 0.3

Table 2: The zone-based accuracy of four classifiers when using original dataset vs generated dataset by non-private WGAN and DPWGIAN on $\epsilon = 1, 5, 10, 15$ and $\delta = 10^{-5}$ (for 384 records).

	Accuracy (%)			
	MLP	Random Forest	SVM	Decision Tree
Original	74.0 ± 3.1	90.2 ± 3.7	74.7 ± 1.1	86.7 ± 2.0
WGAN (non-private)	78.1 ± 3.4	81.5 ± 4.4	83.4 ± 3.6	76.5 ± 3.2
DPWGIAN ($\epsilon = 15$)	77.3 ± 1.1	85.7 ± 1.5	88.9 ± 0.9	75.7 ± 1.5
DPWGIAN ($\epsilon = 10$)	71.3 ± 2.4	71.8 ± 2.1	74.0 ± 2.1	56.6 ± 3.4
DPWGIAN ($\epsilon = 5$)	63.2 ± 3.1	64.7 ± 1.9	65.2 ± 2.7	56.4 ± 2.7
DPWGIAN ($\epsilon = 1$)	61.7 ± 3.8	60.9 ± 3.2	62.5 ± 1.9	62.9 ± 4.6

Table 3: The zone-based accuracy of four classifiers when using original dataset vs generated dataset by non-private CGAN and DPCGAN on $\epsilon = 1, 5, 10, 15$ and $\delta = 10^{-5}$ (for 384 records).

	Accuracy (%)			
	MLP	Random Forest	SVM	Decision Tree
Original	74.0 ± 3.1	90.2 ± 3.7	74.7 ± 1.1	86.7 ± 2.0
CGAN (non-private)	66.2 ± 3.2	65.3 ± 2.8	64.3 ± 0.7	71.1 ± 0.4
DPCGAN ($\epsilon = 15$)	57.6 ± 2.4	63.9 ± 1.9	61.2 ± 4.0	61.9 ± 2.2
DPCGAN ($\epsilon = 10$)	53.4 ± 3.1	57.9 ± 4.5	58.8 ± 1.5	57.8 ± 3.2
DPCGAN ($\epsilon = 5$)	49.2 ± 1.5	57.6 ± 3.4	51.5 ± 1.4	55.9 ± 3.5
DPCGAN ($\epsilon = 1$)	46.9 ± 2.7	53.8 ± 2.1	49.9 ± 2.1	55.3 ± 1.2

non-private WGAN and DPWGIAN for $\epsilon = 1, 5, 10, 15$ and $\delta = 10^{-5}$ based on the number of fingerprints in the dataset. As it is mentioned before, the number of records in the original dataset is 384, while for the synthetic datasets, we tried to generate more fingerprint records so that we could improve the localization accuracy as well. The results show that higher accuracy occurs in the original, non-private WGAN and DPWGIAN with $\epsilon = 15$ datasets when the data points incremented up to 500; however, for more private datasets, higher number of records has no specific influence on the accuracy.

Besides, the accuracy of zone-based MLP classifiers for CGAN and DPCGAN is plotted in Figure 10. Here

also the classifier is trained on the original dataset (dashed black line) vs generated datasets by non-private WGAN and DPWGIAN for $\epsilon = 1, 5, 10, 15$ and $\delta = 10^{-5}$ based on the number of data points in the dataset. It can be seen that the zone-based accuracies of the trained classifier by non-private CGAN and DPCGAN are much lower than the accuracy of the trained classifier by the original dataset. However, it has been improved when it has learned from more synthetic records. The reason for this is that there are 15 labels for all 384 records in a zone-based mode, so we have a low number of records for each label which is not that helpful for training the classifiers. With a higher number of records, on the

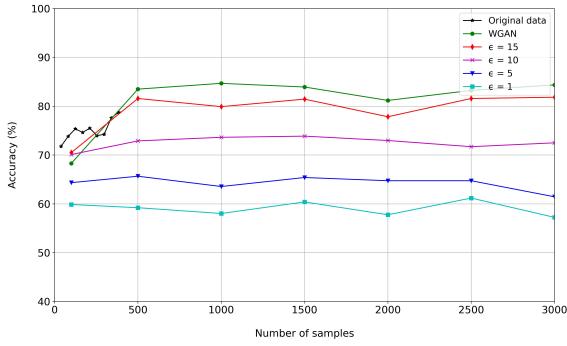


Fig. 9: The zone-based accuracy of MLP classifiers when using original dataset vs generated dataset by non-private WGAN and DPWGAN on $\epsilon = 1, 5, 10, 15$ and $\delta = 10^{-5}$ based on the number of fingerprint samples in the dataset.

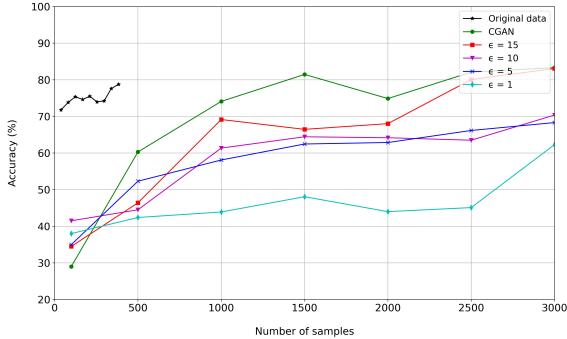


Fig. 10: The zone-based accuracy of MLP classifiers when using original dataset vs generated dataset by non-private CGAN and DPCGAN on $\epsilon = 1, 5, 10, 15$ and $\delta = 10^{-5}$ based on the number of fingerprint samples in the dataset.

other hand, more data is involved to train the synthetic dataset for every label.

In addition, note that since DPCGAN is conditionally trained based on the labels, the higher number of records affects this conditionally private GAN more in comparison with DPWGAN, which labels are also generated similarly to RSS features. It is also observed that for a higher number of records, even more private datasets achieve the same or higher level of accuracy than the original dataset.

6.3 Privacy Evaluation

To measure the disclosure risk of our model, we compute the average Euclidean distances between the generated

data points and the closest point in the original dataset. Formally, let D and D' denote the original and generated dataset, respectively. The disclosure risk is then defined as follows:

$$\text{Disclosure}_{\min}(D, D') = \frac{1}{|D'|} \sum_{x' \in D'} \arg \min_{x \in D} \|x - x'\| \quad (13)$$

where $|D'|$ denotes the number of records in D' , and $\|\cdot\|$ is the Euclidean distance between the points $x \in D$ and $x' \in D'$; $\arg \min$ selects the minimum of these distances, and $\frac{1}{|D'|}$ returns the average of the associated distances. It should be noted that the higher value of Disclosure_{\min} means that the disclosure risk (privacy violation) is lowered.

Table 4 reports the disclosure risk (in meters) for the generated dataset linked to the original dataset. It can be observed that, as expected, when more privacy is obtained (ϵ reduces), the disclosure risk is reduced (Disclosure_{\min} increases). In general, among the three techniques, the non-private CGAN and DPCGAN show higher privacy gain (higher Disclosure_{\min}) in the zone-based mode, and non-private WGAN and DPWGAN show lower privacy gain (lower Disclosure_{\min}) in the location-based mode.

6.4 Discussion

The amount of noise that is added to the gradients can be controlled by the privacy budget (ϵ) and the choice of the DP mechanism used, which can be optimized to balance utility and privacy. Therefore, while the privacy budget (controlled by ϵ) does affect the privacy guarantees of DPGANs, it has a relatively small impact on the quality of the generated data compared to other factors. This is because the DP mechanism is only applied to the gradients of the Generator, which are used to update the parameters of the Generator during the training process. The generated data is not directly affected by the DP mechanism, as it is only indirectly affected by the gradients. The choice of ϵ should be then made carefully to balance utility and privacy, but other factors such as the architecture of the GAN, the quality of the training data, and the choice of hyper-parameters are also important for the quality of the generated data.

Furthermore, our experimental results for the DP-GAN framework on indoor location data suggest using DPWGAN in location-based localization, whenever accuracy is a matter of importance. However, this accuracy comes with the price of privacy loss. On the other hand, DPCGAN in zone-based mode shows higher privacy gain with lower accuracy, and DPWGAN in zone-based

Table 4: Average minimum Euclidean distances.

	Location-based (DP)WGAN				
	WGAN (non-private)	DPWGAN ($\varepsilon = 15$)	DPWGAN ($\varepsilon = 10$)	DPWGAN ($\varepsilon = 5$)	DPWGAN ($\varepsilon = 1$)
$Disclosure_{Min}(m)$	5.88	6.20	6.56	6.98	7.01

(a) Disclosure risk of the dataset generated by non-private WGAN and DPWGAN in the location-based localization.

	Zone-based (DP)WGAN				
	WGAN (non-private)	DPWGAN ($\varepsilon = 15$)	DPWGAN ($\varepsilon = 10$)	DPWGAN ($\varepsilon = 5$)	DPWGAN ($\varepsilon = 1$)
$Disclosure_{Min}(m)$	6.78	7.54	8.03	8.81	9.89

(b) Disclosure risk of the dataset generated by non-private WGAN and DPWGAN in the zone-based localization.

	Zone-based (DP)CGAN				
	CGAN (non-private)	DPCGAN ($\varepsilon = 15$)	DPCGAN ($\varepsilon = 10$)	DPCGAN ($\varepsilon = 5$)	DPCGAN ($\varepsilon = 1$)
$Disclosure_{Min}(m)$	7.50	7.92	11.38	14.68	18.38

(c) Disclosure risk of the dataset generated by non-private CGAN and DPCGAN in the zone-based localization.

mode shows the best trade-off between privacy gain and utility loss. These results provide insight into the selection of the appropriate DPGAN model for the indoor location framework based on the need of users, which is higher utility, higher privacy, or a better trade-off between utility and privacy.

7 Conclusion and Future Work

In this paper, we suggest an indoor location framework based on differentially private GANs (DPGANs) aiming to generate a privacy-preserving synthetic dataset for both location-based and zone-based indoor localization systems. We specifically investigate the use of two popular DPGANs, namely DPWGAN and DPCGAN. DPGANs can provide a promising approach for balancing the competing demands of privacy and accuracy by generating synthetic location data that is statistically similar to the real data but does not reveal specific information about any individual user. This can enable accurate location-based services to be provided without compromising user privacy. We have also exhibited that synthetic data can enhance the performance of fingerprinting localization, especially in situations where the procedure of collecting data is expensive and time-consuming. While the DPGANs framework can ensure the confidentiality of training data, the synthetic data is vulnerable in very private modes especially when the privacy budget is less than 1. Therefore, improving the indoor location DPGAN framework for higher privacy is one of the future works. In addition, the current framework is limited by having only a single Generator and Discriminator. Hence, a promising avenue for further investigation could be to expand this framework to encompass multiple Generators and Discriminators, enabling them to tackle more intricate problems.

References

- J. Tamas and Z. Toth, “Topology-based evaluation for symbolic indoor positioning algorithms,” *IEEE Transactions on Industry Applications*, vol. 55, no. 6, pp. 6324–6331, 2019.
- M. Sheikhalishahi, A. Saracino, F. Martinelli, and A. L. Marra, “Privacy preserving data sharing and analysis for edge-based architectures,” *Int. J. Inf. Sec.*, vol. 21, no. 1, pp. 79–101, 2022. [Online]. Available: <https://doi.org/10.1007/s10207-021-00542-x>
- P. Roy and C. Chowdhury, “A survey of machine learning techniques for indoor localization and navigation systems,” *J. Intell. Robotics Syst.*, vol. 101, no. 3, mar 2021. [Online]. Available: <https://doi.org/10.1007/s10846-021-01327-z>
- N. Singh, S. Choe, and R. Punmiya, “Machine learning based indoor localization using wi-fi rssi fingerprints: An overview,” *IEEE Access*, vol. 9, pp. 127 150–127 174, 2021.
- V. Moghadasaiee, S. A. Ghorashi, and M. Ghavami, “New reconstructed database for cost reduction in indoor fingerprinting localization,” *IEEE Access*, vol. 7, pp. 104 462–104 477, 2019.
- A. Fathalizadeh, V. Moghadasaiee, and M. Alishahi, “Indoor geo-indistinguishability: Adopting differential privacy for indoor location data protection,” *IEEE Transactions on Emerging Topics in Computing*, 2022.
- , “On the privacy protection of indoor location dataset using anonymization,” *Computers & Security*, vol. 117, p. 102665, 2022.
- X. Jiang, X. Zhou, and J. Grossklags, “Privacy-preserving high-dimensional data collection with federated generative autoencoder,” *Proc. Priv. Enhancing Technol.*, vol. 2022, no. 1, pp. 481–500, 2022.
- Z. Hu, Y. Li, G. Jiang, R. Zhang, and M. Xie, “Prihorus: Privacy-preserving rss-based indoor positioning,” in *ICC 2022 - IEEE International Conference on Communications*, 2022, pp. 5627–5632.
- X. Zhang, F. He, Q. Chen, X. Jiang, J. Bao, T. Ren, and X. Du, “A differentially private indoor localization scheme with fusion of wifi and bluetooth fingerprints in edge computing,” *Neural Comput. Appl.*, vol. 34, no. 6, p. 4111–4132, 2022.
- O. Tasbaz, V. Moghadasaiee, and B. Farahani, “Zone-based federated learning in indoor positioning,” in *2022 12th International Conference on Computer and Knowledge Engineering (ICCKE)*, 2022, pp. 163–168.

12. N. Park, M. Mohammadi, K. Gorde, S. Jajodia, H. Park, and Y. Kim, "Data synthesis based on generative adversarial networks," *Proc. VLDB Endow.*, vol. 11, no. 10, p. 1071–1083, jun 2018.
13. I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in neural information processing systems*, 2014, pp. 2672–2680. [Online]. Available: <http://papers.nips.cc/paper/5423-generative-adversarial-nets.pdf>
14. R. Torkzadehmahani, P. Kairouz, and B. Paten, "Dp-gan: Differentially private synthetic data and label generation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2019.
15. T. Zhu, D. Ye, W. Wang, W. Zhou, and P. S. Yu, "More than privacy: Applying differential privacy in key areas of artificial intelligence," *IEEE Transactions on Knowledge & Data Engineering*, vol. 34, no. 06, pp. 2824–2843, jun 2022.
16. C. Xu, J. Ren, D. Zhang, Y. Zhang, Z. Qin, and K. Ren, "Ganoffuscator: Mitigating information leakage under gan via differential privacy," *IEEE Transactions on Information Forensics and Security*, vol. 14, pp. 2358–2371, 2019.
17. S. Holcer, J. Torres-Sospedra, M. Gould, and I. Remolar, "Privacy in indoor positioning systems: A systematic review," in *International Conference on Localization and GNSS (ICL-GNSS)*, 2020, pp. 1–6.
18. H. Li, L. Sun, H. Zhu, X. Lu, and X. Cheng, "Achieving privacy preservation in wifi fingerprint-based localization," in *IEEE Conference on Computer Communications (INFOCOM)*, 2014, pp. 2337–2345.
19. K. Järvinen, H. Leppäkoski, E.-S. Lohan, P. Richter, T. Schneider, O. Tkachenko, and Z. Yang, "Pilot: Practical privacy-preserving indoor localization using outsourcing," in *IEEE European Symposium on Security and Privacy (EuroS P)*, 2019, pp. 448–463.
20. P. Zhao, W. Liu, G. Zhang, Z. Li, and L. Wang, "Preserving privacy in wifi localization with plausible dummy locations," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 10, pp. 11 909–11 925, 2020.
21. Y. Wang, M. Huang, Q. Jin, and J. Ma, "Dp3: A differential privacy-based privacy-preserving indoor localization mechanism," *IEEE Communications Letters*, vol. 22, no. 12, pp. 2547–2550, 2018.
22. J. W. Kim and B. Jang, "Workload-aware indoor positioning data collection via local differential privacy," *IEEE Communications Letters*, vol. 23, no. 8, pp. 1352–1356, 2019.
23. H. Navidan, V. Moghtadaiee, N. Nazaran, and M. Alishahi, "Hide me behind the noise: Local differential privacy for indoor location privacy," in *IEEE European Symposium on Security and Privacy*, 2022, pp. 1–10.
24. P. Zhao, H. Jiang, J. C. S. Lui, C. Wang, F. Zeng, F. Xiao, and Z. Li, "P3-loc: A privacy-preserving paradigm-driven framework for indoor localization," *IEEE/ACM Transactions on Networking*, vol. 26, no. 6, pp. 2856–2869, 2018.
25. M. E. Andrés, N. E. Bordenabe, K. Chatzikokolakis, and C. Palamidessi, "Geo-indistinguishability: Differential privacy for location-based systems," in *ACM SIGSAC Conference on Computer Communications Security*, ser. CCS '13. Association for Computing Machinery, 2013, p. 901–914.
26. W. Li, C. Zhang, and Y. Tanaka, "Pseudo label-driven federated learning-based decentralized indoor localization via mobile crowdsourcing," *IEEE Sensors Journal*, vol. 20, no. 19, pp. 11 556–11 565, 2020.
27. B. S. Ciftler, A. Albaseer, N. Lasla, and M. Abdallah, "Federated learning for rss fingerprint-based localization: A privacy-preserving crowdsourcing method," in *2020 International Wireless Communications and Mobile Computing (IWCMC)*. IEEE, 2020, pp. 2112–2117.
28. Y. Liu, H. Li, J. Xiao, and H. Jin, "Floc: Fingerprint-based indoor localization system under a federated learning updating framework," in *2019 15th International Conference on Mobile Ad-Hoc and Sensor Networks (MSN)*. IEEE, 2019, pp. 113–118.
29. P. Wu, T. Imbiriba, J. Park, S. Kim, and P. Closas, "Personalized federated learning over non-iid data for indoor localization," in *2021 IEEE 22nd International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*. IEEE, 2021, pp. 421–425.
30. Z. Wu, X. Wu, X. Long, and Y. Long, "A privacy-preserved online personalized federated learning framework for indoor localization," in *2021 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*. IEEE, 2021, pp. 2834–2839.
31. Q. D. Vo and P. De, "A survey of fingerprint-based outdoor localization," *IEEE Communications Surveys Tutorials*, vol. 18, no. 1, pp. 491–506, 2016.
32. P. Roy and C. Chowdhury, "A survey of machine learning techniques for indoor localization and navigation systems," *Journal of Intelligent & Robotic Systems*, vol. 101, no. 3, pp. 1–34, 2021.
33. C. Dwork and A. Roth, "The algorithmic foundations of differential privacy," *Found. Trends Theor. Comput. Sci.*, vol. 9, no. 3–4, p. 211–407, 2014. [Online]. Available: <https://doi.org/10.1561/0400000042>
34. C. Dwork, K. Kenthapadi, F. McSherry, I. Mironov, and M. Naor, "Our data, ourselves: Privacy via distributed noise generation," in *Advances in Cryptology - EUROCRYPT 2006*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, pp. 486–503.
35. I. Mironov, "Rényi differential privacy," in *2017 IEEE 30th Computer Security Foundations Symposium (CSF)*, 2017, pp. 263–275.
36. M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein generative adversarial networks," in *Proceedings of the 34th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, D. Precup and Y. W. Teh, Eds., vol. 70. PMLR, 06–11 Aug 2017, pp. 214–223.
37. I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. C. Courville, and Y. Bengio, "Generative adversarial networks," *Commun. ACM*, vol. 63, no. 11, pp. 139–144, 2020. [Online]. Available: <https://doi.org/10.1145/3422622>
38. M. Abadi, A. Chu, I. Goodfellow, H. B. McMahan, I. Mironov, K. Talwar, and L. Zhang, "Deep learning with differential privacy," in *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, ser. CCS '16. New York, NY, USA: Association for Computing Machinery, 2016, p. 308–318. [Online]. Available: <https://doi.org/10.1145/2976749.2978318>
39. L. Xie, K. Lin, S. Wang, F. Wang, and J. Zhou, "Differentially private generative adversarial network," 2018.