

Privacy-Preserving Wi-Fi Data Generation via Differential Privacy in Diffusion Models

Ningning Wang*, Tianya Zhao*, Shiwen Mao†, Xuyu Wang*[§]

*Knight Foundation School of Computing and Information Sciences, Florida International University, Miami, FL 33199, US

†Department of Electrical and Computer Engineering, Auburn University, Auburn, AL 36849, USA

Emails: nwang012@fiu.edu, tzhao010@fiu.edu, smao@ieee.org, xuyuwang@fiu.edu

Abstract—Due to the considerable effort and time needed to collect and label wireless data, there is a compelling need for data generation to facilitate data augmentation. To ensure the reliability of the data, the generated data needs to perform well in common evaluation metrics. However, this process can lead to the model memorizing some training data, resulting in potential privacy leaks. One major threat is the membership inference attack (MIA), which determines whether a specific sample was used in training the target model. While MIA has been extensively studied for discriminative models, its impact and defenses for generative models remain less explored. In this paper, we propose a hybrid training method for the diffusion model applied to Wi-Fi data as a defense against MIAs. The approach involves initially training the model without privacy constraints. After a specified number of training rounds, differential privacy (DP) is incorporated for fine-tuning. During this second phase, a co-optimization process is conducted in parallel to counteract the effects of the added noise. Experimental results demonstrate that the hybrid training method effectively defends against state-of-the-art MIAs for generative models without compromising model performance or requiring additional training efforts, showing significant promise for practical applications.

Index Terms—Wi-Fi Data, Diffusion Models, Membership Inference Attack, Differential Privacy.

I. INTRODUCTION

Recently, wireless sensing has been extensively leveraged in various Internet of Things (IoT) applications (e.g., activity recognition, vital sign monitoring, fall detection) because of its high accessibility [1]. Consequently, the demand for wireless data is steadily rising to support the diverse range of IoT sensing tasks. To meet the different task requirements, researchers are exploring the use of generative models for wireless data augmentation. Existing wireless data generation models primarily rely on variational autoencoders (VAE), generative adversarial networks (GAN), and diffusion models [2]. These models require substantial training data to achieve better results. Essentially, these models learn the distribution within the training data and then generate new wireless data that follow this distribution. However, these models primarily focus on expanding feature-level distributions and struggle to precisely generate raw wireless signals due to their limited representation capabilities [3]. Among these models, the diffusion model demonstrates superior performance compared to other generative models in various tasks [4]. Its training

process is straightforward and avoids typical problems like mode collapse or convergence issues, as it does not involve balancing competing parts or require delicate fine-tuning [5]. Although the privacy of generated data has received increasing attention in the field of computer vision [6]–[9], it has been largely neglected in the wireless domain.

Membership inference attacks (MIAs) pose a significant privacy threat by determining whether a specific data sample was used in training a target model [10]. For instance, if an individual’s medical record is included in the training of a disease prediction model, it could violate privacy protocols and reveal personal information. Although privacy attacks on training data with generative models are less frequently discussed compared to those involving discriminative models, they are gaining increased attention due to their potential threats. For example, Hayes et al. highlight MIA attacks on generative models [11]. To address this problem, a privacy-preserving GAN model is proposed to solve the privacy leakage problem caused by generated images [12]. In the wireless domain, similar privacy concerns arise. While wireless data is widely used for its strong privacy protections, current methods do not fully address privacy leakage issues, leaving sensitive data (e.g., health or user information) vulnerable to exposure by models. Shi et al. discuss the application of MIA to a wireless signal classifier [13]. This leakage of private information can be exploited by adversaries, for example, by spoofing signals that mimic those from authorized users using similar radio devices and waveforms in comparable spectrum environments.

The defense for MIAs is a crucial aspect of maintaining the privacy and security of machine learning models. Various methods have been proposed to mitigate the risk of MIAs [14], [15]. While various defense methods exist for generative models, including differential privacy (DP), adversarial regularization, and overfitting minimization [16], [17], these approaches have primarily focused on GANs. Given that diffusion models now outperform GANs, ensuring their privacy has become increasingly important. Moreover, in the wireless domain, the data structure and properties differ from those of images, making commonly used methods not directly applicable to wireless data. Therefore, to study and enhance the privacy-preserving capabilities of wireless data generated by diffusion models while maintaining data usability, we face the following challenges.

[§]The corresponding author is Xuyu Wang (xuyuwang@fiu.edu).

Challenges. First, wireless data often uses complex-valued numbers to represent phase and amplitude, which complicates data processing with conventional methods. Additionally, the temporal information in these signals is highly sensitive to perturbations, making it difficult to maintain the original physical characteristics after processing. Second, while DP effectively protects data by adding noise, it can significantly degrade model performance, especially in the sensitive wireless domain. This creates a critical trade-off between privacy and utility. Third, the challenge is exacerbated in diffusion models, which have a large number of parameters. Applying fine-tuning or noise addition across all parameters not only severely impacts model performance but also greatly increases training complexity. A more practical approach is needed to balance privacy protection with model effectiveness and computational efficiency. These challenges underscore the importance of careful consideration in the effective and secure use of wireless data across various applications. Thus, addressing and mitigating the risks of privacy leakage in generative tasks involving wireless data is crucial for advancing the field and ensuring both data security and model performance.

Solution. First, to preserve the original physical characteristics after processing, our approach begins by training a complex-valued diffusion model in the conventional manner and then using the weights of this trained model as a pre-trained model. Next, we continue training this base model with differentially private stochastic gradient descent (DP-SGD), selectively applying DP-SGD to the attention and embedding modules of the diffusion model. This approach minimizes retraining overhead and addresses the initial complexity of processing wireless data. Finally, we introduce a small neural network for co-optimization, designed to mitigate the noise introduced by DP-SGD and perform noise reduction on the raw data. This step helps to further alleviate the side effects of adding DP and avoids the need for fine-tuning the entire model. Our in-depth analysis demonstrates that this hybrid training approach effectively lowers the upper bound of the area under the curve (AUC) for the MIA, thereby enhancing privacy protection while maintaining high-quality data generation.

The main contributions in this paper include:

- 1) To the best of our knowledge, this is the first work that harnesses the DP in the diffusion model to defend against MIA in wireless signals.
- 2) We also propose a hybrid training method to alleviate the side effects of DP. This approach is simple yet effective in reducing the interference caused by noise.
- 3) We implement the proposed system to generate Wi-Fi data, improving its resistance to MIA while maintaining the reliability of the generated data. This reduces the attack success rate from 97% to 72% for the raw data.

The remainder of this paper is organized as follows. Section II reviews related work. Section III presents preliminaries and motivation. Section IV details our methodology and Section V demonstrates our defense method. The experimental study is in Section VI. Section VII concludes this paper.

II. RELATED WORK

In this section, we review the existing literature and research related to the MIA defense method in the wireless domain. We first introduce the state-of-the-art (SOTA) wireless data generation. Subsequently, the MIA methods are included. Finally, the defense mechanisms against MIAs are discussed.

A. Wireless Sensing Data Generation

Due to the widespread presence of wireless signals in various environments, utilizing wireless data for sensing functions has become increasingly prevalent [1], [18]. More specifically, deep learning methods have been used for improving wireless sensing performance [19]. For example, Zhang et al. propose deep learning-aided wireless sensing systems for human detection [20]. However, collecting labeled wireless data is challenging. It has led to increased exploration of generative models for wireless data augmentation. For example, Patel et al. leverage a conditional GAN model to enhance the performance of wireless modulation classification [21]. In addition, Rizk et al. utilize generated data to extract features from the original data, thereby improving the performance of wireless localization systems [22].

Building on these advancements, diffusion models, particularly the denoising diffusion probabilistic model (DDPM), have demonstrated superiority over GANs and VAEs in generative data techniques [23]. These models utilize a diffusion process for data generation, which involves two main stages: forward diffusion and reverse denoising. Recently, these diffusion models have been used in the wireless sensing domain. For example, RF Genesis employs diffusion models to generate and fuse wireless data, enhancing the generalization of wireless sensing and enabling adaptation to new environments [24]. Moreover, Chi et al. apply diffusion models in various wireless domains, including Wi-Fi sensing, radar monitoring, and wireless communications, achieving results that closely resemble the original data and significantly enhance data quality [25].

Although the aforementioned works focus on generating more realistic data, they do not address the potential privacy risks associated with the generated wireless data. Our proposed hybrid training approach not only generates high-quality data but also effectively prevents information leakage from the original data.

B. Membership Inference Attacks

MIA is a privacy attack where an adversary determines if a specific data point was in the training set, threatening models trained on private data by revealing included records. It was first proposed by Shokri et al. [10], targeting discriminative models. This attack leverages the fact that machine learning models often behave differently on training data compared to data they have never seen. The adversary can create shadow models by training several models on data similar to what they suspect the target model was trained on. These shadow models help the adversary understand how models typically

behave on training versus non-training data, and this concept has been widely adopted in subsequent MIA methods.

With the advent of MIA, attacks against generative models have also been proposed [11], [26], [27]. In [11] and [27], new GAN structures were introduced to score target samples using the discriminator. In [26], membership is determined by examining how closely synthesized samples cluster around the target sample. The core idea of most MIAs is that training samples tend to exhibit a smaller reconstruction loss compared to unseen samples.

C. Defense against MIAs

Recently, numerous studies have concentrated on developing defense techniques to protect generative models from MIAs. DP has proven to be an effective means of preventing MIA attacks [28]. Current MIA defense methods for GANs focus on configuring the discriminator [29] and directly tuning the data [30]. For defending against attacks on diffusion models, two main methods have emerged: adding DP and fine-tuning the model. For example, Dockhorn et al. propose differential private diffusion models (DPDM) [31], while Lyu et al. introduce a framework for differentially private generative modeling by fine-tuning the attention modules and conditioning embedders using DP-SGD [32].

Unlike previous attacks on generative models that primarily targeted GANs by exploiting the overfitting of the GAN's discriminator, *our proposed method focuses on the diffusion model in the Wi-Fi domain to prevent overfitting to real training data*. Diffusion models have shown superiority in generating realistic wireless data but pose privacy risks through MIA. By employing a hybrid training approach and selectively applying DP-SGD, we ensure high-quality data generation while effectively mitigating MIA risks, thus maintaining robust privacy protection and model performance.

III. PRELIMINARIES AND MOTIVATION

In this section, we will introduce the theoretical background of our approach, incorporating topics such as membership inference attacks in the wireless domain, differential privacy, and hybrid training methods.

A. Membership Inference Attack in the Wireless Domain

Membership inference attack leverages deep learning to infer the presence of training data by comparing the predicted probability distributions of the outputs to determine if a given sample is a member of the training set. Let us denote a DNN classifier as $f(x|\theta)$, trained using algorithm A on a dataset $D_{\text{train}} = \{(x^{(n)}, y^{(n)})\}_{n=1}^N$, where x is the data sample and y is the class label. Once the training is completed, the parameters of the classifier $f(x|\theta^*)$ will be fixed, which can then be used to make prediction vector $\hat{p}(y|x)$ on unseen data. MIA is defined as the ability of an attacker, given an input x and access to the classifier model, to determine whether x belongs to D_{train} .

While MIA has been extensively studied in different domains like image data, its application in the wireless domain remains limited. This is primarily due to the less intuitive

nature of wireless data compared to image data and the added complexity and temporal information inherent in wireless data. Despite these challenges, recent studies such as [13] have highlighted the high success rate of MIA on wireless data, undermining its perceived security.

This paper focuses on a *black-box MIA*, where the adversary does not know the target classifier. The adversary cannot directly access the target classifier but can query it and collect data from its outputs such as corresponding posterior probabilities. The attack exploits the differences in the behavior of the classifier on training data versus non-training data to infer the membership. The shadow model is trained to approximate the target model $f(x|\theta)$. The goal is to determine if a sample x belongs to the training set D_{train} . When we have the private dataset and the other dataset, we can obtain two probability distributions $P_D(y|x)$ and $P_{\bar{D}}(y|x)$, respectively. The probability distributions are provided by the inference model m . Then, the gain function for MIA developed in [33] is given by

$$G(m) = \mathbb{E}_{(x^{(n)}, y^{(n)}) \sim P_D} [\log m] + \mathbb{E}_{(x^{(n)}, y^{(n)}) \sim P_{\bar{D}}} [\log(1 - m)], \quad (1)$$

where $E[\cdot]$ is the expectation function. Since the probability distributions are unknown, we consider an empirical gain on a data set D^A , which is a representative subset of D , and a data set \bar{D}^A , which is a representative subset of \bar{D} , respectively. The empirical gain [33] is defined by

$$G_{D^A, \bar{D}^A}(m) = \frac{1}{|D^A|} \sum_{(x^{(n)}, y^{(n)}) \in D^A} \log m + \frac{1}{|\bar{D}^A|} \sum_{(x^{(n)}, y^{(n)}) \in \bar{D}^A} \log(1 - m). \quad (2)$$

The objective is to find the optimal inference model m by maximizing the empirical gain:

$$\max_m G_{D^A, \bar{D}^A}(m). \quad (3)$$

If the subset is the same, it means the optimal solution to the above problem $m = 0.5$ for all samples. The MIA is not successful if there is no difference in distributions. Therefore, we aim to make $\hat{m} \approx 0.5$.

B. Differential Privacy

DP is a rigorous mathematical framework designed to quantify the privacy guarantees provided when performing statistical analyses on sensitive data. It has proven to be an effective means of preventing MIA attacks [28]. Generally, a training algorithm is said to satisfy DP if, after observing the output of the algorithm, an adversary cannot confidently determine whether any data was included in the input to the algorithm. This privacy guarantee is controlled by two parameters ϵ and δ , which can increase the privacy by decreasing the two parameters [34]. There is an inherent trade-off between utility and privacy: models with high privacy guarantees may have limited practical utility. The definition of DP is as follows:

a mechanism $M : D \rightarrow R$ with domain D and range R satisfies (ϵ, δ) -DP if for all possible sets of the mechanism's outputs S and all neighboring datasets d, d' that differ by a single entry. For any subset of outputs $S \subseteq R$, it holds that,

$$\Pr[M(d) \in S] \leq e^\epsilon \Pr[M(d') \in S] + \delta. \quad (4)$$

The Gaussian mechanism adds noise drawn from a Gaussian distribution to the output of a function to ensure DP. Given a function $\mu : D \rightarrow \mathbb{R}^p$, the Gaussian mechanism adds noise $n \sim \mathcal{N}(0, \sigma^2 \Delta_\mu^2 I)$, where σ is a function of ϵ and δ , and Δ_μ is the global sensitivity of the function [35]. Adding noise results in the released function $\bar{\mu}(D)$ being less accurate than its non-DP counterpart, $\mu(D)$. This introduces a trade-off between privacy and accuracy.

A DP algorithm is leveraged to train a neural network using sensitive data. The primary method in our work is DP-SGD [36], which is a modified stochastic gradient descent algorithm. It clips the gradients to a predefined norm and adds Gaussian noise to the clipped gradients to ensure privacy. The parameter update in DP-SGD is defined by

$$\theta \leftarrow \theta - \eta \left(\frac{1}{B} \sum_{i \in B} \text{clip}_C(\nabla_\theta l_i(\theta)) + \frac{C}{B} z \right), \quad (5)$$

where $z \sim \mathcal{N}(0, \sigma_{\text{DP}}^2 I)$, B is a mini-batch of training examples, η is the learning rate, l_i is the loss of training sample i , and $\text{clip}_C(g) = \min\left(1, \frac{C}{\|g\|_2}\right) g$.

IV. METHODOLOGY

A. Problem Statement

To ensure the reliability of the generated data, it needs to perform well in common evaluation metrics and maintain the same distribution as the original training data. However, this similarity increases the risk of private training data leakage, thereby raising the likelihood of MIA. Consequently, our goal is to mitigate privacy leakage while maintaining the quality of the generated data.

In this paper, we protect wireless data by adding noise to the model, thus introducing the challenge of balancing privacy protection with generation quality. To preserve the original physical information of the data, we use DP-SGD to selectively add noise to specific parameters of the model, which minimizes the impact on overall model performance and reduces training complexity. Furthermore, we utilize a joint optimization model to mitigate the effects of noise. Our proposed hybrid training method ensures that the generated data remains useful while significantly reducing the risk of privacy leakage.

B. Hybrid as a MIA Defense

In our hybrid training approach, we can divide the training process into two distinct parts: non-private and private steps. First, we train a diffusion model using the original data without considering privacy leakage issues. The goal of this initial training step is to enable the model to converge quickly and capture the essential features of the original data,

thereby generating high-quality synthetic data. Specifically, let $\mathbf{X} = \{\mathbf{x}_i\}_{i=1}^n$ represent our original data and θ represent the model parameters. The training process can be expressed as

$$\theta^* = \arg \min_{\theta} \mathcal{L}(\mathbf{X}; \theta), \quad (6)$$

where \mathcal{L} is the loss function used to measure the difference between the generated data and the real data. This step focuses on optimizing the model parameters θ to minimize the loss function, thereby ensuring that the model learns to generate data that closely resembles the original dataset.

After the initial training phase, we fine-tune the trained model by introducing DP. Specifically, we add noise to the model parameter updates to make the model insensitive to individual data points. When we update the parameter θ at the t -th iteration by introducing noise $n \sim \mathcal{N}(0, \sigma^2 \Delta_\mu^2 I)$, the parameter update rule is expressed by

$$\theta_{t+1} = \theta_t - \eta(\nabla \mathcal{L}(\mathbf{X}; \theta_t) + n), \quad (7)$$

where η is the learning rate. This update rule ensures that the added noise makes it difficult to determine the presence of any individual data point, thereby guaranteeing DP.

To further mitigate the impact of noise on model performance, we introduce a joint optimization model for training. Let ϕ represent the parameters of the joint optimization model. We optimize both θ and ϕ using a joint objective function \mathcal{J} , which balances the trade-off between data quality and privacy protection. The optimization process can be expressed as:

$$(\theta^*, \phi^*) = \arg \min_{\theta, \phi} \mathcal{J}(\mathbf{X}; \theta, \phi), \quad (8)$$

where \mathcal{J} is the joint optimization loss function that considers both the generated data quality and the privacy protection capability of the model. This joint optimization approach ensures that the model parameters θ and ϕ are fine-tuned to achieve a balance between generating high-quality data and maintaining privacy protection.

The above approach can enable the diffusion model to generate high-quality data while ensuring privacy protection. By first training the model on the original data to capture its features and then fine-tuning it with DP mechanisms, we achieve a robust hybrid training method. This method leverages the strengths of both non-private and private training phases, ensuring that the synthetic data generated is of high quality and the privacy of the original data is preserved, which is formulated by

$$\mathcal{L}(\mathbf{X}; \theta) = \mathbb{E}_{\mathbf{x} \sim \mathbf{X}} [\|\mathbf{x} - f_\theta(\mathbf{z})\|^2], \quad (9)$$

$$\mathcal{J}(\mathbf{X}; \theta, \phi) = \mathcal{L}(\mathbf{X}; \theta) + \lambda \mathcal{R}(\mathbf{X}; \phi), \quad (10)$$

where f_θ denotes the model function, \mathbf{z} represents the latent variables, \mathcal{R} is the regularization term for privacy, and λ is the hyperparameter controlling the trade-off between data quality and privacy protection.

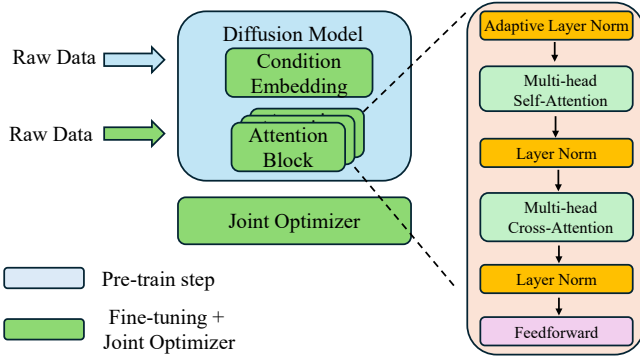


Fig. 1: A schematic of hybrid training diffusion model.

V. OUR DEFENSE METHOD

A. Pre-training

Our generative model is a complex-valued diffusion model specifically designed for wireless data [25], with its structure shown in Fig. 1. During the pre-training process, the input data is first down-sampled to a uniform size. Then, the original signal is progressively eliminated as the diffusion step t increases. The signal x_t denotes the input signal x after t steps of noise addition, eventually converging to a noise distribution:

$$q(x_t|x_0) = \mathcal{CN}(x_t; \bar{\gamma}_t x_0, \bar{\sigma}_t^2 I), \quad (11)$$

where $q(x_t|x_0)$ is the probability density function of the transformed signal x_t at step t , \mathcal{CN} indicates that x_t follows a complex normal distribution. $\bar{\gamma}_t$ and $\bar{\gamma}_s$ denote as the scaling factor at step t and step s , respectively. The variance $\bar{\sigma}_t^2$ can be expressed in terms of the parameters of the diffusion process:

$$\bar{\sigma}_t = \sum_{s=1}^t \left(\sqrt{1 - \alpha_s} \frac{\bar{\gamma}_t}{\bar{\gamma}_s} \right), \quad (12)$$

where α_s is a predefined parameter controlling the noise schedule, $\alpha_s \in (0, 1)$ and $\epsilon \sim \mathcal{CN}(0, I)$.

As the diffusion step t increases, the original signal is gradually eliminated, and x_t converges to the noise:

$$\lim_{T \rightarrow \infty} x_T = \lim_{T \rightarrow \infty} \sum_{t=1}^T (\sqrt{1 - \alpha_t} / \bar{\gamma}_t) \epsilon. \quad (13)$$

The reverse process aims to restore the original data distribution by removing noise. In the reverse restoration process, to learn a parameterized distribution $p_\theta(x_0)$ that approximates the original distribution $q(x_0)$, we utilize the Kullback-Leibler (KL) divergence and minimize the loss function [23]:

$$\theta = \arg \min_{\theta} D_{KL}(q(x_{t-1}|x_t, x_0) \| p_\theta(x_{t-1}|x_t)), \quad (14)$$

where $q(x_{t-1}|x_t, x_0)$ is the reverse process conditioned on x_0 , $p_\theta(x_{t-1}|x_t)$ denotes the reverse process fitted by diffusion model. Since both $q(x_{t-1}|x_t, x_0)$ and $p_\theta(x_{t-1}|x_t)$ are Gaussian distributions, the KL divergence between two Gaussian

distributions $q = \mathcal{N}(y; \mu_q, \sigma_q^2 I)$ and $p = \mathcal{N}(y; \mu_p, \sigma_p^2 I)$ is given by:

$$D_{KL}(q \| p) = \frac{1}{2} \left(\frac{\sigma_q^2}{\sigma_p^2} + \frac{(\mu_q - \mu_p)^2}{\sigma_p^2} - 1 + \log \frac{\sigma_p^2}{\sigma_q^2} \right). \quad (15)$$

Assume $q(x_{t-1}|x_t, x_0) \sim \mathcal{N}(x_{t-1}; \tilde{\mu}_{t-1}, \tilde{\sigma}_{t-1}^2 I)$ and $p_\theta(x_{t-1}|x_t) \sim \mathcal{N}(x_{t-1}; \mu_\theta(x_t), \sigma_\theta^2 I)$, where $\tilde{\mu}_{t-1}$ and $\tilde{\sigma}_{t-1}^2$ are the mean and variance of the true reverse process, respectively, and $\mu_\theta(x_t)$ and σ_θ^2 are the mean and variance of the parameterized reverse process, respectively.

We aim to minimize $D_{KL}(q(x_{t-1}|x_t, x_0) \| p_\theta(x_{t-1}|x_t))$. Assuming $\tilde{\sigma}_{t-1} = \sigma_\theta$ (i.e., the variances are the same and only the means differ), the KL divergence simplifies to the mean squared error (MSE) between the means:

$$D_{KL}(q \| p) \propto \frac{1}{2\sigma_\theta^2} E_{q(x_0)} [\|\tilde{\mu}_{t-1} - \mu_\theta(x_t)\|^2], \quad (16)$$

where the term $\frac{1}{2\sigma_\theta^2}$ is a constant for θ , so that minimizing the KL divergence is equivalent to minimizing the MSE between the means. Therefore, the parameter optimization is simplified to minimizing Eq. 16. By following the aforementioned sequence, we can successfully add noise to the data and subsequently recover it.

B. Fine-tuning and DP

In the second training phase, DP-SGD is incorporated into the diffusion model. Given the complexity and size of the entire diffusion model, it is impractical to apply DP-SGD to all parameters. Thus, we selectively apply DP-SGD to the attention module and the embedding module within the model. By fine-tuning these modules, we can retain the characteristics of the previously trained data while ensuring DP.

The attention modules are illustrated in Fig. 1. The cross-attention module is responsible for computing the key (K) and value (V) vectors as projections of the conditioning embedded inputs. This allows the model to attend to different parts of the input data based on the conditioning information. The multi-head self-attention component is designed to extract autocorrelation features from the noisy input, enhancing the model's ability to learn complex dependencies within the data.

By focusing on the attention modules and the conditioning embedder, we ensure that the essential features of the data are preserved while introducing DP. The DP-SGD mechanism is applied to the parameters of the attention modules and embedding modules, denoted as θ_{att} and θ_{embed} , respectively. According to Eq. 7, the parameter of DP-SGD will be updated. This update rule is specifically applied to the parameters θ_{att} and θ_{embed} , ensuring that the privacy-preserving noise is introduced effectively.

The schematic diagram in Fig. 1 further clarifies the structure and functionality of the attention modules. The cross-attention mechanism allows the model to align and integrate information from the conditioning inputs, while the multi-head self-attention mechanism enables the model to capture intricate patterns within the data. By fine-tuning these modules

with DP-SGD, we achieve a balance between maintaining the quality of the generated data and protecting the privacy of the original data.

The introduction of DP-SGD into the attention and embedding modules of the diffusion model ensures that the essential characteristics of the original data are retained while providing robust privacy protection. By selectively applying DP-SGD to these critical components, we effectively manage the trade-off between data quality and privacy, allowing the model to generate high-fidelity data with enhanced privacy guarantees.

C. Joint Optimizer

To mitigate the effect of noise introduced by DP, we introduce a denoising model $g_\phi(z)$, where z is the output of the diffusion model $f_\theta(x)$. Our goal is to train the denoising model to minimize the denoising loss. The denoising loss function measures the error between the output of the denoising model and the original input. The training objective for the denoising model is formulated as:

$$\min_{\phi} \mathbb{E}_{(x,y) \sim \mathcal{D}} [\mathcal{L}(g_\phi(f_\theta(x)), x)], \quad (17)$$

where \mathcal{L} denotes the loss function that quantifies the difference between the denoised output and the original input data x .

To further improve the performance, we introduce a joint optimization approach that simultaneously optimizes the parameters of both the diffusion model and the denoising model. The joint loss function, which is *total_loss*, is designed to balance the trade-off between the quality of the generated data and the effectiveness of the denoising process. The joint optimization objective is given by

$$\min_{\theta, \phi} \mathbb{E}_{(x,y) \sim \mathcal{D}} [\mathcal{L}(f_\theta(x), y) + \lambda \mathcal{L}(g_\phi(f_\theta(x)), x)] + \mathcal{N}(0, \sigma^2), \quad (18)$$

where λ is a parameter that balances the loss of the diffusion model, $\mathcal{L}(f_\theta(x), y)$, with the loss of the denoising model, $\mathcal{L}(g_\phi(f_\theta(x)), x)$. The term $\mathcal{N}(0, \sigma^2)$ represents the noise introduced by the DP mechanism, with σ being the standard deviation of the noise.

In the joint optimization process, the parameters of the diffusion model θ and the denoising model ϕ are alternately updated to minimize the joint loss. This ensures that the generated data maintains high fidelity while effectively mitigating the noise introduced by DP. The alternate updating mechanism allows for the fine-tuning of both models, achieving an optimal balance between data quality and privacy protection.

In summary, the introduction of a denoising model, combined with a joint optimization strategy, effectively addresses the noise introduced by DP. By carefully balancing the loss functions of both the diffusion and denoising models, we ensure that the generated data is of high quality while providing robust privacy protection. This method leverages the strengths of both models, resulting in enhanced performance and reliability.

The hybrid training approach is provided in Algorithm 1. Initially, the diffusion model parameters, denoted as θ , are

Algorithm 1 Hybrid Training Algorithm

Input: Dataset \mathcal{D} , learning rate η , noise multiplier σ , gradient clipping threshold C , fine-tuning hyperparameter λ

Output: $\hat{x}^{(i)}$

- 1: Initialize diffusion model parameters θ
 - 2: Pre-train the diffusion model on the original data \mathcal{D} until convergence
 - 3: Obtain the pre-trained diffusion model $\theta_{\text{pretrained}}$
 - 4: Initialize denoising model parameters ϕ
 - 5: **for** each fine-tuning iteration **do**
 - 6: Sample a batch of data $\{(x^{(i)}, y^{(i)})\}_{i=1}^N$ from \mathcal{D}
 - 7: $g_i = \nabla_{\theta_{\text{att}}, \theta_{\text{embed}}} \mathcal{L}(A(x^{(i)}; \theta_{\text{att}}, \theta_{\text{embed}}), y^{(i)})$
 - 8: DP-SGD: $\tilde{g} = \frac{1}{B} \sum_{i \in B} \text{clip}_C(\nabla_{\theta} l_i(\theta)) + \frac{C}{B} z$
 - 9: $\theta_{\text{att}}, \theta_{\text{embed}} \leftarrow \theta_{\text{att}}, \theta_{\text{embed}} - \eta \tilde{g}$
 - 10: $\mathcal{L}_{\text{denoising}} = \mathcal{L}(g_\phi(f_{\theta_{\text{att}}, \theta_{\text{embed}}}(x^{(i)})), x^{(i)})$
 - 11: $\phi \leftarrow \phi - \eta \nabla_{\phi} \mathcal{L}_{\text{denoising}}$
 - 12: $\mathcal{L}_{\text{diffusion}} = \mathcal{L}(f_{\theta_{\text{att}}, \theta_{\text{embed}}}(x^{(i)}), y^{(i)})$
 - 13: Compute the total loss: \rightarrow Eq.18
 - 14: $\theta \leftarrow \theta - \eta \nabla_{\theta} \text{total_loss}$
 - 15: $\phi \leftarrow \phi - \eta \nabla_{\phi} \text{total_loss}$
 - 16: Compute $\hat{x}^{(i)}$: $\hat{x}^{(i)} = g_\phi(f_{\theta_{\text{att}}, \theta_{\text{embed}}}(x^{(i)}))$
 - 17: **end for**
 - 17: **return** $\hat{x}^{(i)}$
-

initialized, and the model is pre-trained on the original dataset, \mathcal{D} , until convergence is achieved, resulting in pre-trained parameters, $\theta_{\text{pretrained}}$. The denoising model parameters, ϕ , are then initialized. During each fine-tuning iteration, a mini-batch of data, $(x^{(i)}, y^{(i)})_{i=1}^N$, is sampled from the dataset. The algorithm then computes the gradients for the attention and embedding modules for the loss function, $\mathcal{L}(A(x^{(i)}; \theta_{\text{att}}, \theta_{\text{embed}}), y^{(i)})$, where θ_{att} and θ_{embed} represent the parameters for the attention and embedding modules, respectively. A is the diffusion model.

DP-SGD is applied to the computed gradients. The adjusted gradient, \tilde{g} , is obtained by clipping the individual gradients to a maximum norm C , summing them, and adding Gaussian noise scaled by $\frac{C}{B}$, where B is the batch size. The diffusion model parameters θ_{att} and θ_{embed} are then updated using these adjusted gradients with a learning rate η . Subsequently, the denoising loss, $\mathcal{L}_{\text{denoising}}$, is computed based on the output of the denoising function g_ϕ applied to the output of the diffusion model, $f_{\theta_{\text{att}}, \theta_{\text{embed}}}(x^{(i)})$, with respect to the original input $x^{(i)}$. The denoising model parameters ϕ are updated using the gradient of this loss.

The diffusion loss, $\mathcal{L}_{\text{diffusion}}$, is calculated using the diffusion model output and the corresponding labels $y^{(i)}$. The total loss is then computed as the sum of the diffusion loss and a weighted denoising loss, controlled by the hyperparameter λ . Finally, both the diffusion model parameters θ and the denoising model parameters ϕ are jointly optimized by updating them with the gradients of the total loss. $\hat{x}^{(i)}$ is the output of the optimized diffusion model of input data $x^{(i)}$. This algorithm ensures that the diffusion model is effectively

fine-tuned with the incorporation of DP, while the denoising model is simultaneously optimized to enhance the overall model performance.

VI. EXPERIMENT AND EVALUATION

We implement our method and evaluate its performance through experiments. We use PyTorch 2.3.1 and run them on NVIDIA A100. The detailed settings are illustrated as follows.

Dataset: We leverage a public dataset from Widar3.0 [37], which contains 9 gestures of 16 users collected across 75 domains (5 positions \times 5 orientations \times 3 environments). We focus on 9 specific gestures and 9 users to analyze overall performance because the distribution of other users across the 75 domains is not uniform.

Metrics: To evaluate the performance of our method, we use accuracy (ACC), structural similarity index measure (SSIM) [38], Frechet inception distance (FID) [39] as the primary metrics for both gesture recognition and user identification. ACC measures the confidence in predictions for each instance. SSIM assesses the similarity between two images, while FID evaluates the quality of images generated by generative models by measuring the distance between the feature distributions of generated and real images. In this paper, we treat our data as if it were an image, utilizing SSIM and FID metrics to represent the quality of data generation. Higher SSIM values and lower FID values indicate that the generated data is closer to the real data. To demonstrate the effectiveness of our proposed method in defending against MIA, we also present the attack success rate of the model. The SSIM and FID are calculated using the following formulas, respectively.

$$\text{SSIM}(x, y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)}, \quad (19)$$

where μ_x and μ_y are the mean value of input x and y , respectively. σ_x^2 and σ_y^2 are the variance of the input x and y , respectively. σ_{xy} is the covariance of inputs. C_1 and C_2 are two constant values.

$$\text{FID}(r, g) = \|\mu_r - \mu_g\|^2 + \text{Tr}(\Sigma_r + \Sigma_g - 2(\Sigma_r \Sigma_g)^{1/2}), \quad (20)$$

where μ_r and μ_g are the means of the features from real and generated data, Σ_r and Σ_g are the covariance matrices of real and generated data, and Tr denotes the trace of the matrix.

To demonstrate the defense capability of this method against MIA, we use the area under the receiver operating characteristic curve (AUCROC) and attack success rate (ASR) as evaluation metrics. AUCROC is a performance measurement for classification problems at various threshold settings, providing insight into the trade-off between true positive and false positive rates. ASR is commonly used in security and privacy contexts to evaluate the effectiveness of attacks on machine learning models, measuring the proportion of successful attacks out of the total number of attempts. The formulas for AUCROC and ASR can be expressed as follows:

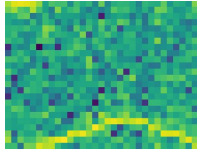
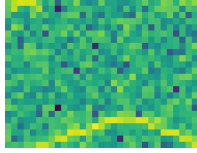
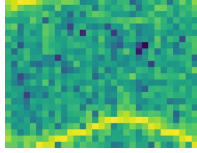
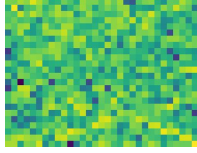
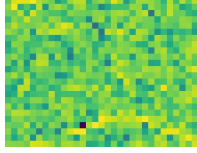
$$\text{AUC} = \int_0^1 \text{TPR}(\text{FPR}^{-1}(x)) dx, \quad (21)$$

$$\text{ASR} = \frac{\text{Number of Successful Attacks}}{\text{Total Number of Attacks}}, \quad (22)$$

AUCROC integrates the true positive rate (TPR) as a function of the false positive rate (FPR) across different thresholds, providing a single scalar value that summarizes the overall performance of the classifier. ASR, on the other hand, provides a direct measure of the vulnerability of the model to MIA, highlighting the need for effective defense mechanisms.

A. Overall Performance

TABLE I: Illustrative examples.

Methods	Example	SSIM	FID
Ground Truth		N/A	N/A
Ours		0.67	6.60
RF-Diffusion [25]		0.89	4.51
RF-Diffusion [25] + DP-SGD		0.38	9.13
CVAE [40]		0.22	11.28

We first evaluate the data quality of the models generated under different conditions. To express the generation effect of wireless data more intuitively, we convert the original Wi-Fi channel state information (CSI) data into a spectrogram after applying short-time fourier transform (STFT). The generation quality is quantitatively verified using SSIM and FID. The results are illustrated in Table I. The diffusion model used in the experiments is the RF-Diffusion designed in [25].

From Table I, the evaluation results demonstrate that our proposed hybrid training method generates Wi-Fi signals with high fidelity, achieving an average structural similarity of 67% relative to the ground truth and an average FID of 6.60. In comparison, the original diffusion model achieves an SSIM of 89% and an FID of 4.51, indicating a higher fidelity in the

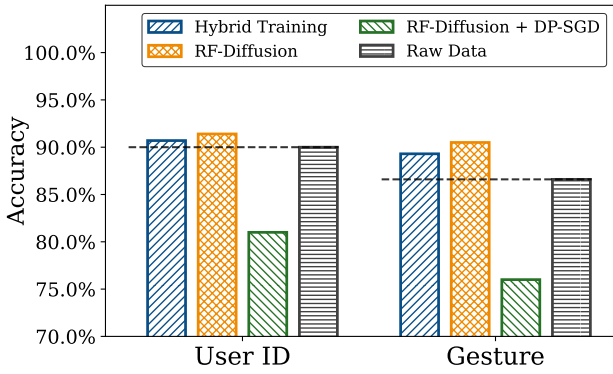


Fig. 2: In-domain experiments.

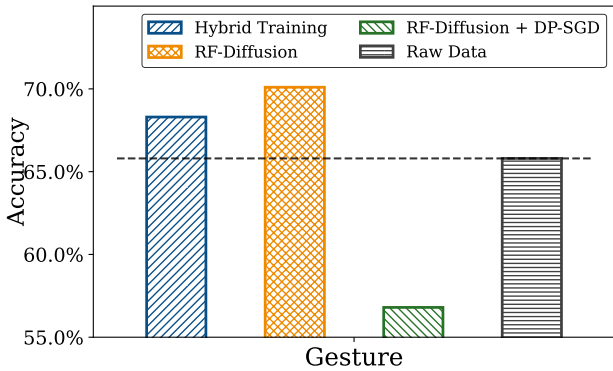


Fig. 3: Cross-domain experiments.

generated data. Additionally, we test the effect of adding DP-SGD directly to the diffusion model for model noise addition. This results in the worst performance, with an SSIM of 38% and an FID of 9.13, suggesting that the addition of DP-SGD negatively impacts the generation quality. We also conduct the complex variation autoencoder (CVAE) method [40] and find that its performance cannot compare to the diffusion model.

After generating several Wi-Fi datasets using different diffusion models, we conduct data augmentation experiments to verify the performance of these wireless datasets. These experiments included both in-domain and cross-domain classification tasks. For the in-domain experiments, we implement classification tasks for both gestures and users. In the cross-domain experiments, we select one user’s data as the test set and employ the remaining users’ data as the training set. The generated data is mixed with the original data in a 1 : 1 ratio. The classifiers used in the experiments are all based on the *ResNet18* architecture.

For in-domain experiments, the results are shown in Fig. 2. The evaluation results illustrate that the generated data has improved the performance of the classification tasks. Specifically, the original data achieves accuracies of 90.5% and 86.6% in user identification and gesture recognition tasks, respectively. When mixed with data generated by the original diffusion

model, the recognition accuracy increases to 91.4% and 90.5%, respectively. Wi-Fi data obtained through our hybrid training method further improves the experimental accuracy by 0.7% and 2.6%, respectively. However, when we initially use the diffusion model with added DP-SGD, the generated data does not improve model performance.

For cross-domain experiments, the results are shown in Fig. 3. It is noticed that the generated Wi-Fi data can improve the performance of cross-domain tasks. The original data achieves an accuracy of 65.2% in the gesture recognition task. When mixed with data generated by the original diffusion model, the recognition accuracy increases to 70.1%. Data obtained through our hybrid training method further boosts the experimental accuracy by 3.0%. However, if the diffusion model with added DP-SGD is used initially, the generated data still does not improve model performance.

From both the in-domain and cross-domain experiments, the results demonstrate that our proposed method is effective for data enhancement and ensures the reliability of the data. However, when using the diffusion model with the addition of DP-SGD, training from scratch results in data that is not effectively augmented. We believe this is because the noise addition significantly impacts the model, leading to substantial deviations from the original data, as shown in the generated results in Table I.

B. Privacy Performance

After testing the quality of the generated data, we conduct privacy protection experiments to assess the effectiveness of using the generated data to protect the original data. In this experiment, we employ a black-box MIA, assuming that the attacker can only access the target model and obtain its output, without knowing the data distribution or the internal structure of the model. The MIA model used in this study is based on the method described by Salem et al. in [41].

To verify that the original data is used for training, we utilize the generated data for classification training and then subjected it to MIA. The results are demonstrated in Table II. These results indicate that the original unprotected data, without any added protection, is highly susceptible to attacks, with the success rate reaching 97%. This highlights the vulnerability of the raw data to membership inference attacks. When DP-SGD noise is added to the diffusion model, the attack becomes ineffective, as indicated by a significantly lower attack success rate. However, as mentioned earlier, at this point, the data is also ineffective for data augmentation purposes. This presents a trade-off between privacy protection and data utility.

Our proposed hybrid training method successfully lowers the attack success rate to approximately 70%, balancing data utility with privacy protection. The evaluation results for different conditions are summarized in Table II, showing that our method achieves a lower AUCROC and ASR compared to other methods.

TABLE II: Attack AUCROC and ASR under different conditions.

	AUCROC	ASR
Raw data	0.99	0.97
Hybrid Training	0.80	0.72
RF-Diffusion	0.93	0.84
Diffusion + DP-SGD	0.52	0.50

C. Impact of Joint Optimizer

To further illustrate the impact of our proposed joint optimization approach on the model, we control the variable λ mentioned in Eq. 18 to explore its effect on data enhancement. The parameter λ controls the balance between different loss functions in the joint optimization process. Given that the weights of the loss functions are not the same, we normalize the output ranges of the two loss functions to ensure a fair comparison. This normalization ensures that adjusting the hyperparameter λ allows the model loss to be appropriately accounted for in the overall loss function.

The results, presented in Fig. 4 and Fig. 5, confirm our conjecture that as the weight of the denoising loss function increases, the model’s output gets closer to the original diffusion model output without additive protection. However, due to the simple structure of the joint optimization model, the denoising effect is not able to completely remove the noise. Consequently, the generated data still retains some level of noise, which creates a difference from the original signal. This retained noise is beneficial as it aids in protecting the data against MIA.

In Fig. 4, we observe that the model’s performance in terms of data enhancement improves as λ increases. This indicates that providing more weight to the denoising loss function allows the model to better approximate the original data distribution. However, this also means that the protection offered by the noise addition diminishes, making the data more vulnerable to MIAs.

Similarly, Fig. 5 illustrates the cross-domain performance of the model under different values of λ . As λ increases, the model’s ability to generalize across different domains improves, but again, the noise reduction makes the data less effective in defending against MIAs. These results highlight the trade-off between data fidelity and privacy protection, emphasizing the importance of carefully selecting λ to balance these two objectives.

While increasing λ enhances data fidelity, it also reduces the level of protection against MIAs. The joint optimization model, despite its simplicity, manages to strike a balance between denoising and retaining enough noise to protect the data. This balance is crucial for ensuring that the data remains useful for augmentation while providing adequate privacy protection.

VII. CONCLUSION

This paper proposes an effective defense approach, hybrid training, against MIA on generative models for Wi-Fi CSI

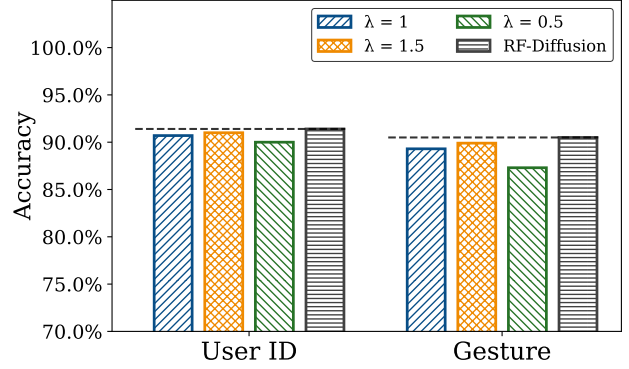


Fig. 4: Impact of joint optimizer on in-domain experiments.

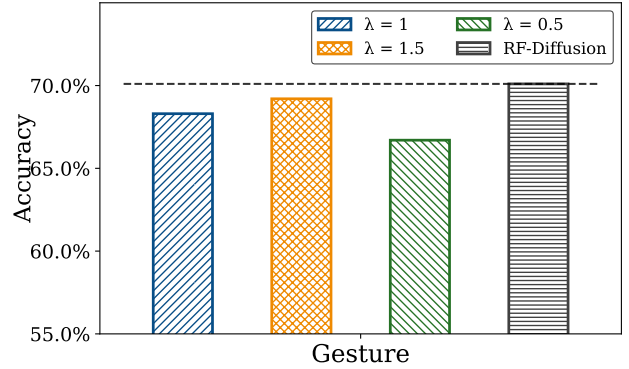


Fig. 5: Impact of joint optimizer on cross-domain experiments.

data. By modifying the training algorithm, hybrid training leverages the benefits of DP-SGD while mitigating its adverse effects on the generated data. We experimentally verify the feasibility of our method. The experimental results illustrate that our approach maintains data enhancement capabilities while addressing the privacy leakage problem of the original data. Our approach achieves an acceptable trade-off between data utility and privacy protection. Using the same MIA method, the data has a 97% attack success rate before privacy protection is added. Our approach successfully reduces the attack success rate to about 70%. The SSIM and FID are changed from 0.89 and 4.51 to 0.67 and 6.60, respectively. In the data enhancement experiment, the data generated by our method is mixed with the original data, resulting in a 2.6% improvement in recognition accuracy to 89%, which surpasses the performance achieved with the traditional DP method. These results confirm that our proposed hybrid training provides a robust solution to enhance data utility while effectively protecting against privacy attacks.

ACKNOWLEDGMENTS

This work is supported in part by the NSF (CNS-2415209, CNS-2321763, CNS-2317190, IIS-2306791, CNS-2319343, CNS-2107190, CNS-2415208, and CNS-2319342).

REFERENCES

- [1] J. Liu, H. Liu, Y. Chen, Y. Wang, and C. Wang, "Wireless sensing for human activity: A survey," *IEEE Communications Surveys & Tutorials*, vol. 22, no. 3, pp. 1629–1645, 2019.
- [2] T.-H. Vu, S. K. Jagatheesaperumal, M.-D. Nguyen, N. Van Huynh, S. Kim, and Q.-V. Pham, "Applications of generative AI (GAI) for mobile and wireless networking: A survey," *arXiv preprint arXiv:2405.20024*, 2024.
- [3] Z. Yang, Y. Zhang, and Q. Zhang, "Rethinking fall detection with Wi-Fi," *IEEE Transactions on Mobile Computing*, vol. 22, no. 10, pp. 6126–6143, 2022.
- [4] T. Wu, Z. Chen, D. He, L. Qian, Y. Xu, M. Tao, and W. Zhang, "CDDM: Channel denoising diffusion models for wireless communications," in *GLOBECOM 2023-2023 IEEE Global Communications Conference*. IEEE, 2023, pp. 7429–7434.
- [5] P. Dhariwal and A. Nichol, "Diffusion models beat gans on image synthesis," *Advances in neural information processing systems*, vol. 34, pp. 8780–8794, 2021.
- [6] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-resolution image synthesis with latent diffusion models," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 10 684–10 695.
- [7] A. Ramesh, P. Dhariwal, A. Nichol, C. Chu, and M. Chen, "Hierarchical text-conditional image generation with clip latents," *arXiv preprint arXiv:2204.06125*, vol. 1, no. 2, p. 3, 2022.
- [8] C. Saharia, W. Chan, S. Saxena, L. Li, J. Whang, E. L. Denton, K. Ghasemipour, R. Gontijo Lopes, B. Karagol Ayan, T. Salimans *et al.*, "Photorealistic text-to-image diffusion models with deep language understanding," *Advances in neural information processing systems*, vol. 35, pp. 36 479–36 494, 2022.
- [9] Y. Balaji, S. Nah, X. Huang, A. Vahdat, J. Song, Q. Zhang, K. Kreis, M. Aittala, T. Aila, S. Laine *et al.*, "ediff-i: Text-to-image diffusion models with an ensemble of expert denoisers," *arXiv preprint arXiv:2211.01324*, 2022.
- [10] R. Shokri, M. Stronati, C. Song, and V. Shmatikov, "Membership inference attacks against machine learning models," in *2017 IEEE symposium on security and privacy (SP)*. IEEE, 2017, pp. 3–18.
- [11] J. Hayes, L. Melis, G. Danezis, and E. De Cristofaro, "Logan: Membership inference attacks against generative models," *arXiv preprint arXiv:1705.07663*, 2017.
- [12] S. Mukherjee, Y. Xu, A. Trivedi, N. Patowary, and J. L. Ferres, "privGAN: Protecting GANs from membership inference attacks at low cost to utility," *Proceedings on Privacy Enhancing Technologies*, 2021.
- [13] Y. Shi and Y. E. Sagduyu, "Membership inference attack and defense for wireless signal classifiers with deep learning," *IEEE Transactions on Mobile Computing*, vol. 22, no. 7, pp. 4032–4043, 2022.
- [14] J. Jia, A. Salem, M. Backes, Y. Zhang, and N. Z. Gong, "Memguard: Defending against black-box membership inference attacks via adversarial examples," in *Proceedings of the 2019 ACM SIGSAC conference on computer and communications security*, 2019, pp. 259–274.
- [15] Z. Yang, B. Shao, B. Xuan, E.-C. Chang, and F. Zhang, "Defending model inversion and membership inference attacks via prediction purification," *arXiv preprint arXiv:2005.03915*, 2020.
- [16] B. K. Beaulieu-Jones, Z. S. Wu, C. Williams, R. Lee, S. P. Bhavnani, J. B. Byrd, and C. S. Greene, "Privacy-preserving generative deep neural networks support clinical data sharing," *Circulation: Cardiovascular Quality and Outcomes*, vol. 12, no. 7, p. e005122, 2019.
- [17] L. Fan, "A survey of differentially private generative adversarial networks," in *The AAAI Workshop on Privacy-Preserving Artificial Intelligence*, vol. 8, 2020.
- [18] C. Li, M. Liu, and Z. Cao, "WiHF: Enable user identified gesture recognition with WiFi," in *IEEE INFOCOM 2020-IEEE Conference on Computer Communications*. IEEE, 2020, pp. 586–595.
- [19] C. Li, Z. Cao, and Y. Liu, "Deep AI enabled ubiquitous wireless sensing: A survey," *ACM Computing Surveys (CSUR)*, vol. 54, no. 2, pp. 1–35, 2021.
- [20] R. Zhang, X. Jing, S. Wu, C. Jiang, J. Mu, and F. R. Yu, "Device-free wireless sensing for human detection: The deep learning perspective," *IEEE Internet of Things Journal*, vol. 8, no. 4, pp. 2517–2539, 2020.
- [21] M. Patel, X. Wang, and S. Mao, "Data augmentation with conditional GAN for automatic modulation classification," in *Proceedings of the 2nd ACM Workshop on wireless security and machine learning*, 2020, pp. 31–36.
- [22] H. Rizk, A. Shokry, and M. Youssef, "Effectiveness of data augmentation in cellular-based localization using deep learning," in *2019 IEEE Wireless Communications and Networking Conference (WCNC)*. IEEE, 2019, pp. 1–6.
- [23] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," *Advances in neural information processing systems*, vol. 33, pp. 6840–6851, 2020.
- [24] X. Chen and X. Zhang, "Rf genesis: Zero-shot generalization of mmwave sensing through simulation-based data synthesis and generative diffusion models," in *Proceedings of the 21st ACM Conference on Embedded Networked Sensor Systems*, 2023, pp. 28–42.
- [25] G. Chi, Z. Yang, C. Wu, J. Xu, Y. Gao, Y. Liu, and T. X. Han, "Rf-diffusion: Radio signal generation via time-frequency diffusion," in *Proceedings of the 30th Annual International Conference on Mobile Computing and Networking*, 2024, pp. 77–92.
- [26] B. Hilprecht, M. Härterich, and D. Bernau, "Monte carlo and reconstruction membership inference attacks against generative models," *Proceedings on Privacy Enhancing Technologies*, 2019.
- [27] D. Chen, N. Yu, Y. Zhang, and M. Fritz, "Gan-leaks: A taxonomy of membership inference attacks against generative models," in *Proceedings of the 2020 ACM SIGSAC conference on computer and communications security*, 2020, pp. 343–362.
- [28] S. Ben Hamida, H. Mrabet, and A. Jemai, "How differential privacy reinforces privacy of machine learning models?" in *International Conference on Computational Collective Intelligence*. Springer, 2022, pp. 661–673.
- [29] Z. Lin, V. Sekar, and G. Fanti, "On the privacy properties of gan-generated samples," in *International Conference on Artificial Intelligence and Statistics*. PMLR, 2021, pp. 1522–1530.
- [30] Z. Ji, Q. Hu, L. Xiang, and C. Zhou, "Mixup training for generative models to defend membership inference attacks," in *IEEE INFOCOM 2023-IEEE Conference on Computer Communications*. IEEE, 2023, pp. 1–10.
- [31] T. Dockhorn, T. Cao, A. Vahdat, and K. Kreis, "Differentially private diffusion models," *arXiv preprint arXiv:2210.09929*, 2022.
- [32] S. Lyu, M. F. Liu, M. Vinaroz, and M. Park, "Differentially private latent diffusion models," *arXiv preprint arXiv:2305.15759*, 2023.
- [33] M. Nasr, R. Shokri, and A. Houmansadr, "Machine learning with membership privacy using adversarial regularization," in *Proceedings of the 2018 ACM SIGSAC conference on computer and communications security*, 2018, pp. 634–646.
- [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: 24th Annual International Conference on the Theory and Applications of Cryptographic Techniques, St. Petersburg, Russia, May 28-June 1, 2006. Proceedings 25*. Springer, 2006, pp. 486–503.
- [35] C. Dwork, A. Roth *et al.*, "The algorithmic foundations of differential privacy," *Foundations and Trends® in Theoretical Computer Science*, vol. 9, no. 3–4, pp. 211–407, 2014.
- [36] 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*, 2016, pp. 308–318.
- [37] Y. Zheng, Y. Zhang, K. Qian, G. Zhang, Y. Liu, C. Wu, and Z. Yang, "Zero-effort cross-domain gesture recognition with Wi-Fi," in *Proceedings of the 17th annual international conference on mobile systems, applications, and services*, 2019, pp. 313–325.
- [38] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE transactions on image processing*, vol. 13, no. 4, pp. 600–612, 2004.
- [39] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, "Gans trained by a two time-scale update rule converge to a local nash equilibrium," *Advances in neural information processing systems*, vol. 30, 2017.
- [40] K. Sohn, H. Lee, and X. Yan, "Learning structured output representation using deep conditional generative models," *Advances in neural information processing systems*, vol. 28, 2015.
- [41] A. Salem, Y. Zhang, M. Humbert, P. Berrang, M. Fritz, and M. Backes, "MI-leaks: Model and data independent membership inference attacks and defenses on machine learning models," *arXiv preprint arXiv:1806.01246*, 2018.