Generative Adversarial Networks: A Survey Toward Private and Secure Applications

ZHIPENG CAI, ZUOBIN XIONG, HONGHUI XU, PENG WANG, and WEI LI,
Georgia State University
YI PAN, Shenzhen Institute of Advanced Technology, CAS, China and Georgia State University

Generative Adversarial Networks (GANs) have promoted a variety of applications in computer vision and natural language processing, among others, due to its generative model's compelling ability to generate realistic examples plausibly drawn from an existing distribution of samples. GAN not only provides impressive performance on data generation-based tasks but also stimulates fertilization for privacy and security oriented research because of its game theoretic optimization strategy. Unfortunately, there are no comprehensive surveys on GAN in privacy and security, which motivates this survey to summarize systematically. The existing works are classified into proper categories based on privacy and security functions, and this survey conducts a comprehensive analysis of their advantages and drawbacks. Considering that GAN in privacy and security is still at a very initial stage and has imposed unique challenges that are yet to be well addressed, this article also sheds light on some potential privacy and security applications with GAN and elaborates on some future research directions.

CCS Concepts: • **General and reference** → **Surveys and overviews**; • **Security and privacy** → **Privacy-preserving protocols**; • **Computing methodologies** → **Machine learning;**

Additional Key Words and Phrases: Generative adversarial networks, deep learning, privacy and security

ACM Reference format:
Zhipeng Cai, Zuobin Xiong, Honghui Xu, Peng Wang, Wei Li, and Yi Pan. 2021. Generative Adversarial Networks: A Survey Toward Private and Secure Applications. *ACM Comput. Surv.* 54, 6, Article 132 (July 2021), 38 pages.
https://doi.org/10.1145/3459992

1 INTRODUCTION

The technological breakthrough brought by **Generative Adversarial Networks (GANs)** has rapidly produced a revolutionary impact on machine learning and its related fields, and this impact has already flourished to various of research areas and applications. As a powerful generative

This work was partially supported by the National Science Foundation under grants 1741277, 1829674, 1704287, and 2011845.
This work is done when Yi Pan was located Shenzhen Institute of Advanced Technology.

Authors' addresses: Z. Cai, Z. Xiong, H. Xu, P. Wang, and W. Li (corresponding author), Georgia State University, Department of Computer Science, 25 Park Place, Atlanta, Georgia, USA, 30303; emails: zcai@gsu.edu, {zxiong2, hxu16, wpeng12}@student.gsu.edu, wli28@gsu.edu; Y. Pan (corresponding author), Shenzhen Institute of Advanced Technology, CAS, 1068 Xueyuan Avenue, Shenzhen, China, 518055, and Georgia State University, Department of Computer Science, 25 Park Place, Atlanta, Georgia, USA, 30303; email: yipan@gsu.edu.

0360-0300/2021/07-ART132 $15.00
https://doi.org/10.1145/3459992

framework, GAN has significantly promoted many applications with complex tasks, such as image generation, super-resolution, and text data manipulations. Most recently, exploiting GAN to work out elegant solutions to severe privacy and security problems has become increasingly popular in both academia and industry due to its game theoretic optimization strategy. This survey aims to provide a comprehensive review and an in-depth summary of the state-of-the-art technologies and discuss some promising future research directions for GAN in the area of privacy and security. We start our survey with a brief introduction to GAN.

1.1 Generative Adversarial Networks

GAN was first proposed by Goodfellow et al. [40] to serve as a generative model bridging between supervised learning and unsupervised learning in 2014, which is highly praised as "the most interesting idea in the last 10 years in Machine Learning" by Yann LeCun, the winner of the 2018 Turing Award. Typically, a generative model takes a training dataset drawn from a particular distribution as input and tries to produce an estimated probability distribution to mimic a given real data distribution. In particular, a zero-sum game between the generator and the discriminator is designed to achieve realistic data generation. In other words, the generator of GAN is trained to fool the discriminator whose goal is to distinguish the real data from the generated data.

GAN has promoted many emerging data-driven applications related to Big Data and Smart Cities thanks to its fantastic properties: (i) the design of generative models offers an excellent way to capture a high-dimensional probability distribution that is an important research focus in mathematics and engineering domains; (ii) a well-trained generative model can break through the imprisonment of data shortage for technical innovation and performance improvement in many fields, especially for deep learning (e.g., the high-quality generated data can be incorporated into semi-supervised learning, for which the influence of missing data could be mitigated to some extent); and (iii) generative models (particularly GAN) enable learning algorithms to work well with multi-modal outputs, in which more than one correct output may be obtained from a single input for a task (e.g., the next frame prediction) [79].

Prior to GAN, several generative models stemming from the maximum likelihood estimation existed, each of which was state of the art at the time it was proposed. These prior generative models are either explicit density based or implicit density based, depending on whether the underlying distribution can be explicitly pre-defined. Some well-known explicit density-based models, including Restricted Boltzmann Machine (RBM) [5], Fully Visible Belief Networks (FVBN) [37], **Gaussian Mixture Model (GMM)** [102], Naive Bayes Model (NBM) [54], and Hidden Markov Model (HMM) [100], can specialize some specific problems and scenarios but are limited by their common weakness—when the explicitly defined probability density function has intensive parameters and complex dimensions, the computational tractability issue happens where the maximum likelihood estimation may not be able to represent the complexity of the sample data and therefore cannot learn the high-dimension data distribution well. In addition, the majority of prior generative models, such as implicit density-based Markov Chain models, require an assumption of Markov Chain that has an ambiguous distribution and can be mixed between patterns. On the contrary, GAN gets rid of the high-dimension constraint and the Markov Chain dependence. The generator of GAN uses a pre-defined low-dimension latent code as input and then maps its input to the target data dimension. In addition, GAN is a non-parametric method and does not require any approximate distribution or Markov Chain property, which endows GAN with the ability to represent the generated data in a lower dimension using fewer parameters. Most importantly, GAN is more like an adversarial training framework instead of a rigorous formulation. Thus, it is more flexible and extensible to be transformed into many variants according to different requirements, such as **Wasserstein Generative Adversarial Network (WGAN)** [7], **Information Maximizing**

Generative Adversarial Network (InfoGAN) [21], and CycleGAN [160]. Motivated by these characteristics, novel research benefits from GAN in a widespread way.

1.2 The Most Recent Research on GAN

Currently, two mainstream kinds of research on GAN are being conducted concurrently: application-oriented study and theory-oriented study. As the restrictions of previous generative models are overcome by GAN, the applications related to data generation are thoroughly investigated for different data formats, such as image generation [83, 95, 131, 145], **Natural Language Processing (NLP)** [24, 32, 53, 149], time series data generation [17, 28, 30, 42], and semantic segmentation [82, 99, 118, 161]. These application scenarios can be further divided into fine-grained subcategories, including image-to-image translation, image super-resolution, image in-painting, face aging, human pose synthesis, object detection, sketch synthesis, text synthesis, medical data generation, texture synthesis, language and speech synthesis, video and music generation, and so on. Those prosperous applications demonstrate the extraordinary capability and widespread popularity of GAN.

Meanwhile, to push GAN's capability to a higher level, theoretical methodologies proceed to tackle essential issues including non-stable training, mode collapse, gradient vanish, lack of proper evaluation metrics, and so on. Some feasible solutions have been proposed, such as feature matching, unrolled GAN, mini-batch discrimination, the **Self-Attention Generative Adversarial Network (SAGAN)**, label smoothing, proper optimizer, gradient penalty, and alternative loss function [84].

1.3 GAN in Privacy and Security

With individuals' increasing privacy concerns and governments' gradually strengthening privacy regulations, thwarting security and privacy threats has been put in a critical place when designing applications, such as medical image analysis, street-view image sharing, and face recognition.

Thanks to the characteristics of adversarial training, GAN and its variants can be exploited to investigate the privacy and security issues without any pre-determined assumptions of opponents' capabilities that are often hard to be determined in traditional attacks and defense mechanisms. As the adversarial training process can capture the interactions between an attacker and a defender in a min-max game, the GAN-based methods can be formulated to either launch an attack to break a solid defense or implement protection to defend against strong attackers. For an attack model, the generator is modeled as an attacker aiming at fooling a defender (i.e., the discriminator) [10, 38, 47, 158]. In a defense model, the generator is modeled as a defender to resist a powerful attacker (i.e., the discriminator), such as Generative Adversarial Privacy (GAP) [51], Privacy Preserving Adversarial Networks (PPANs) [127], Compressive Adversarial Privacy (CAP) [22], and Reconstructive Adversarial Network (RAN) [77].

In a nutshell, the existing GAN-based privacy and security methods mainly differ in their configurations of GAN models and formulations of loss functions. However, from the perspective of application scenarios, model design, and data utilization, there is plenty of room for taking maximum advantage of GAN, leaving lots of research blanks for further enhancements. Those potential directions are elaborated at the end of this survey.

The organization of this survey is illustrated in Figure 1. We present the preliminaries about GAN and its variants in Section 2. The applications of GAN for privacy, including data privacy and model privacy, are reviewed in Section 3 and Section 4, respectively. Security-related applications are described in Section 5, and promising future works are discussed in Section 6. Finally, we conclude the survey in Section 7.

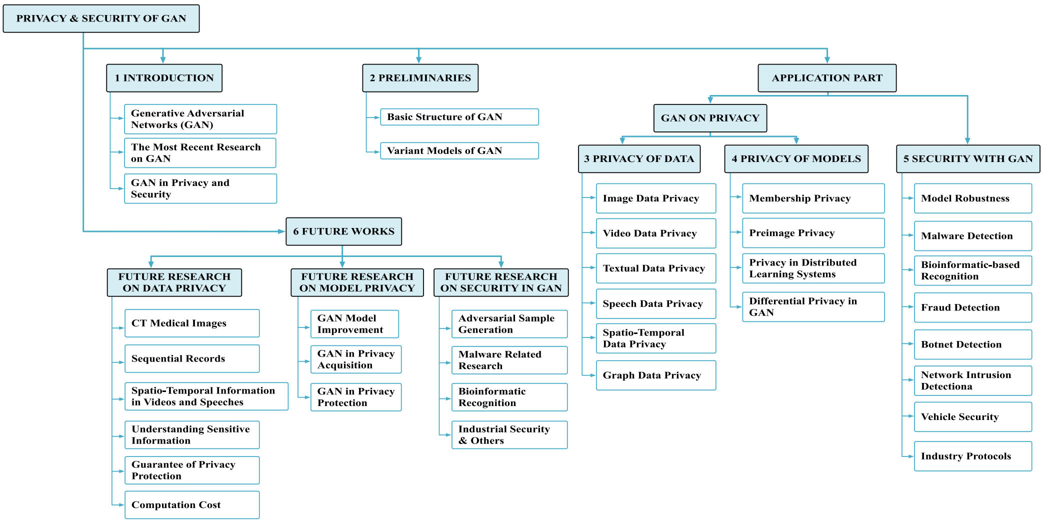
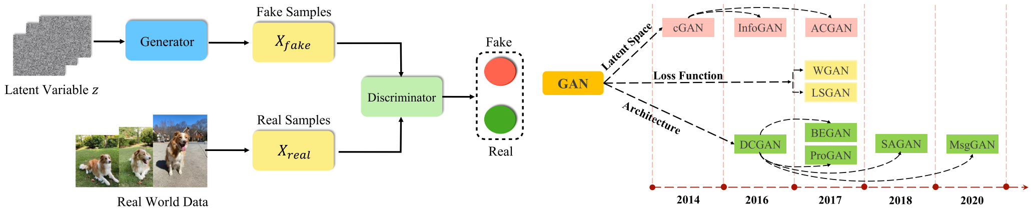

PRIVACY & SECURITY OF GAN
1 INTRODUCTION
Generative Adversarial Networks (GAN)
The Most Recent Research on GAN
GAN in Privacy and Security
2 PRELIMINARIES
Basic Structure of GAN
Variant Models of GAN
APPLICATION PART
GAN ON PRIVACY
3 PRIVACY OF DATA
Image Data Privacy
Video Data Privacy
Textual Data Privacy
Speech Data Privacy
Spatio-Temporal Data Privacy
Graph Data Privacy
4 PRIVACY OF MODELS
Membership Privacy
Preimage Privacy
Privacy in Distributed Learning Systems
Differential Privacy in GAN
5 SECURITY WITH GAN
Model Robustness
Malware Detection
Bioinformatic-based Recognition
Fraud Detection
Botnet Detection
Network Intrusion Detectiona
Vehicle Security
Industry Protocols
6 FUTURE WORKS
FUTURE RESEARCH ON DATA PRIVACY
CT Medical Images
Sequential Records
Spatio-Temporal Information in Videos and Speeches
Understanding Sensitive Information
Guarantee of Privacy Protection
Computation Cost
FUTURE RESEARCH ON MODEL PRIVACY
GAN Model Improvement
GAN in Privacy Acquisition
GAN in Privacy Protection
FUTURE RESEARCH ON SECURITY IN GAN
Adversarial Sample Generation
Malware Related Research
Bioinformatic Recognition
Industrial Security & Others

Fig. 1. The organization of this survey and its taxonomy.



(a) The Architecture of Basic GAN (b) The Evolution of GAN Models

Fig. 2. The architecture of GAN and its variants.

2 PRELIMINARIES

In this section, we review the basic structure of GAN and its variant models.

2.1 Basic Structure of GAN

The basic idea of GAN was first proposed by Goodfellow et al. [40], where a generator can be well trained under an adversarial training framework. As shown in Figure 2(a), GAN consists of a generator G and a discriminator D. G is a function of operating a latent space z to generate real-like fake data X_{fake}, whereas D is a function to distinguish X_{fake} and real data X_{real}. The training process of G is terminated until X_{fake} and X_{real} are indistinguishable by D [48]. The interactions between G and D in the adversarial training scenario can be modeled as a min-max game with the following objective:

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

where $x \sim p_{data}$ denotes the distribution of X_{real}, and $z \sim p_z$ denotes the distribution of z.

2.2 Variant Models of GAN

Inspired by the initial design of GAN, a number of variant models have been proposed for various scenarios. In the following, we introduce several popular ones.

2.2.1 Wasserstein GAN. WGAN was developed to solve the problem of mode collapse in a training process to some extent [7]. To generate real-looking data that can fool a discriminator, WGAN is trained for minimizing the Wasserstein distance between the real-like data distribution p_g and the real data distribution p_{data}.

2.2.2 Least Squares GAN. To tackle the issue of gradients vanishing in the training process of GAN, the *a-b* coding scheme was utilized in the least squares method to formulate the loss function of discriminator in **Least Squares Generative Adversarial Network (LSGAN)** [84]. Accordingly, the objective functions of the discriminator and the generator are expressed as follows respectively:

$$\min_D \frac{1}{2}\mathbb{E}_{x\sim p_{data}}[(D(x)-b)^2] + \frac{1}{2}\mathbb{E}_{z\sim p_z}[(D(G(x))-a)^2], \tag{2}$$

$$\min_G \frac{1}{2}\mathbb{E}_{z\sim p_z}[(D(G(x))-\delta)^2], \tag{3}$$

where δ represents the value that G wants D to classify on fake data.

2.2.3 Conditional GAN. Considering that auxiliary information can also assist in generating data, it is natural to extend GAN to a conditional version named **Conditional Generative Adversarial Network (cGAN)** that provides both the generator and the discriminator with auxiliary information [90, 91]. In cGAN, the latent space z and the auxiliary information y (e.g., class labels and data from other modalities) are combined as the conditional input of the generator to make the conditional fake data as similar as the conditional real data. Accordingly, the objective function of cGAN can be expressed by Equation (4):

$$\min_G \max_D \mathbb{E}_{x\sim p_{data}}[\log D(x|y)] + \mathbb{E}_{z\sim p_z}[\log(1-D(G(z|y)))]. \tag{4}$$

2.2.4 Information Maximizing GAN. InfoGAN attempts to learn representations with the idea of maximizing the mutual information between labels and the generative data [21]. To this end, InfoGAN introduces another classifier Q to predict y given by $G(z|y)$ based on cGAN. Thus, the objective function of InfoGAN is a regularization of cGAN's objective function, shown as follows:

$$\min_G \max_D V(D,G) - \lambda I(G,Q), \tag{5}$$

where $V(D,G)$ is the objective function of cGAN but the discriminator does not take y as input, $I(\cdot)$ is the mutual information, and λ is a positive hyperparameter.

2.2.5 Auxiliary Classifier GAN. The **Auxiliary Classifier Generative Adversarial Network (ACGAN)** is a variant of the basic GAN with an auxiliary classifier [95], which attempts to learn a representation for z with a class label. In ACGAN, every generated sample also has a corresponding class label c. However, the discriminator is trained to classify its input as real or fake. The auxiliary classifier is used to obtain a probability distribution over the class labels. Hence, there are two loss functions for training ACGAN: the log-likelihood of the correct source shown in Equation (6) and the log-likelihood of the correct label presented in Equation (7):

$$L_S = \mathbb{E}[\log P(S = real|X_{real})] + \mathbb{E}[\log P(S = fake|X_{fake})], \tag{6}$$

$$L_C = \mathbb{E}[\log P(C = c|X_{real})] + \mathbb{E}[\log P(C = c|X_{fake})]. \tag{7}$$

In Equation (6) and Equation (7), $P(S = real|X_{real})$ is the probability of determining the data sample to be real when it is real, and $P(C = c|X_{real})$ is the probability of determining the correct class when the data sample is real. In ACGAN, the generator is trained via maximizing $L_C - L_S$, and the discriminator is trained via maximizing $L_C + L_S$.

2.2.6 Deep Convolutional GAN (DCGAN). With the success of deep learning models, especially the **Convolutional Neural Network (CNN)** [64], the **Deep Convolutional Generative Adversarial Network (DCGAN)** has been proposed to generate images and videos efficiently by setting both the generator and the discriminator as CNNs. DCGAN can even produce higher visual quality images with the help of a CNN-based generator and discriminator [119].

2.2.7 Boundary Equilibrium GAN. By configuring the discriminator as an Autoencoder, the **Boundary Equilibrium Generative Adversarial Network (BEGAN)** was developed in the work of Berthelot et al. [13]. To prevent the discriminator from beating the generator easily, BEGAN learns the Autoencoder loss distributions using a loss derived from the Wasserstein distance instead of learning data distributions directly.

In BEGAN, the generator is trained to minimize the loss of image generation in Equation (8), and the discriminator is trained in Equation (9) to minimize the reconstruction loss of the real data and maximize the reconstruction loss of the generated images:

$$L_G = L(G(z)), \tag{8}$$

$$L_D = L(D(x)) - k_t L(D(G(z))). \tag{9}$$

In Equation (9), $k_t = k_{t-1} + \lambda_k(\beta L(D(x)) - L(D(G(z)))$ is a variable that controls the weight of $L(D(G(z)))$ in L_D, where λ_k is the learning rate at the k-th iteration in the training process, and $\beta = \frac{\mathbb{E}[L(D(G(z)))]}{\mathbb{E}[L(D(x))]}$ balances the efforts allocated to the generator and the discriminator.

2.2.8 Progressive-Growing GAN. The **Progressive-Growing Generative Adversarial Network (ProGAN)** [57] is built based on DCGAN, where both G and D start training with low-resolution images. It gradually increases the model depth by adding new layers to G and D during the training process and ends with the generation of high-resolution images.

2.2.9 Self-Attention GAN. Traditional CNNs only focus on local spatial information due to the limited receptive field of CNNs, making it difficult for CNN-based GANs to learn multi-class image datasets. SAGAN [155] is derived from DCGAN to ensure a large receptive field in G and D via a self-attention mechanism so that SAGAN can be used to learn global long-range dependencies for generating multi-class images better.

2.2.10 Multi-Scale Gradients GAN. When there is not enough overlap in the supports of the real and fake data distributions, gradients passing from D to G become uninformative, making it difficult to exploit different datasets using GAN models. The **Multi-Scale Gradient Generative Adversarial Network (MsgGAN)** [56] overcomes this problem by connecting latent space of G and D while training GAN on multiple datasets, in which more information is shared between G and D to make MsgGAN applicable to different datasets.

Summary. The variants of GAN mentioned previously can be categorized into three categories based on their improvement focus, whose evolution is presented in Figure 2(b): (1) *Latent space*: In cGAN, the labels can be used in the latent space as a kind of extra information for better generation and discrimination of labeled data. Based on cGAN, InfoGAN expects to learn representation by maximizing the mutual information between labels and the generative data, whereas ACGAN tries to learn representation with labels by using an auxiliary classifier. (2) *Loss function*: WGAN uses Wasserstein distance to calculate the loss to solve the problem of mode collapse in GAN, whereas LSGAN applies the *a-b* coding scheme in the least squares method to the design of D's loss to solve the problem of gradient vanish in GAN. (3) *Architecture*: The generator in DCGAN is a deep CNN-based architecture to generate more real-like images and videos with high visual quality.

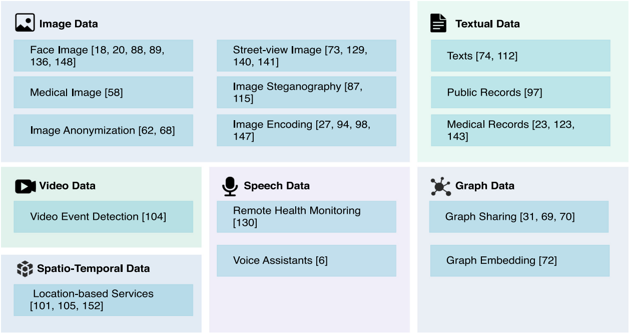


Image Data
Face Image [18, 20, 88, 89, 136, 148]
Street-view Image [73, 129, 140, 141]
Medical Image [58]
Image Steganography [87, 115]
Image Anonymization [62, 68]
Image Encoding [27, 94, 98, 147]
Textual Data
Texts [74, 112]
Public Records [97]
Medical Records [23, 123, 143]
Video Data
Video Event Detection [104]
Speech Data
Remote Health Monitoring [130]
Voice Assistants [6]
Graph Data
Graph Sharing [31, 69, 70]
Graph Embedding [72]
Spatio-Temporal Data
Location-based Services [101, 105, 152]

Fig. 3. The applications of GAN-based data privacy protection.

The discriminator in BEGAN is an autoencoder-based architecture to prevent the discriminator from easily beating the generator at the early training stage for fair adversarial training. Based on DCGAN, ProGAN gradually increases the depth of G and D in the training process of GAN to generate high-resolution images, SAGAN relies on a self-attention mechanism to obtain global long-range dependency to generate multi-class images, and MsgGAN connects the latent space of G and D while training on multiple datasets to ensure it can be exploited to different datasets.

3 PRIVACY OF DATA

According to data type, the mainstream applications of GAN in data privacy protection can be classified into six major categories, including image data privacy, video data privacy, textual data privacy, speech data privacy, spatio-temporal data privacy, and graph data privacy, for which a more detailed classification is presented in Figure 3. Technically speaking, on the one hand, the generator is designed as a perturbation function to hide the private information and/or trained by one or more discriminators for privacy-preserving data generation. On the other hand, the discriminator is employed to ensure data similarity so that the generated privacy-preserving data is still usable in real applications but is hard to be distinguished from the real data by attackers.

3.1 Image Data Privacy

As the most popular images used in deep learning, face images contain various individuals' sensitive information, easily causing privacy leakage, and thus have received lots of research attention [15, 16, 18, 20, 26, 88, 136, 148]. In addition, the privacy of medical images [58] and street-view images [129, 141] have captured research interest in recent years. Furthermore, a number of GAN-based schemes have been developed for image steganography [87, 115], image anonymization [62, 68, 122], and image encoding [27, 94, 98, 147], which indeed can be exploited on any type of image besides face/medical/street-view images. Currently, the study of face images and medical images focuses on a single object, such as one face and one human organ, whereas the study of street-view images deals with multiple objects, including pedestrians, vehicles, buildings, and so on. In the following, the existing works on face images, medical images, street-view images, image steganography, image anonymization, and image encoding are introduced in order.

3.1.1 Face Images. Chen et al. [20] proposed a method of image representation learning based on the **Variational Generative Adversarial Network (VGAN)** for privacy-preserving facial expression recognition, where **Variational Autoencoder (VAE)** [80] and cGAN [90] are combined to create an identity-preserving representation of facial images while generating an expression-preserving realistic vision. In VGAN, the generator (i.e., the encoder-decoder pair in VAE) takes a real image I and its class label c (that indicates a user's identity) as inputs to synthesize a face image. Three discriminators in this VGAN model are designed with different functionalities: (i) D_1 is used for image quality (i.e., the synthesized face images should be similar to the real ones), (ii) D_2 is employed to identity recognition (i.e., the identity of the synthetic image should be determined incorrectly by the person identifier); and (iii) D_3 is exploited for expression recognition (i.e., the facial expression in the synthesized data should be guaranteed). During the training process, three parameters are set to control the weights of image quality, identity recognition, and expression recognition so that a balance between privacy and utility can be achieved in the synthesized images. In addition, to generate identity-preserving face images, Yang et al. [148] also developed a targeted identity-protection iterative method (TIP-IM) using GAN to generate adversarial identity masks for face images to alleviate the identity leakage of face images without sacrificing the visual quality of these face images.

Multi-view identity-preserving face image synthesis (i.e., 3D identity-preserving face image generation) has also been studied by Cao et al. [18]. They proposed a DCGAN-based approach to produce realistic 3D photos while preserving identity of multi-view results, in which a face normalizer and an editor are set as the generators to synthesize the 3D photos, and their corresponding discriminators are used to ensure similarity between the synthesized data and the real data. This proposed method was demonstrated to dramatically improve the pose-invariant face recognition and generate multi-view face images while preventing the leakage of individuals' identification.

As is well known, real-world recognition systems depend on high-resolution images, which can be used to infer users' identities and biometric information like age, gender, race and health condition through a soft biometric classifier. Mirjalili et al. [88] proposed a model based on AC-GAN [95] to hide the gender information in images for privacy protection. In their work [88], Autoencoder [80] is used as the generator, which is the state-of-the-art method of image generation. The discriminator consists of a 0–1 classifier making the perturbed images to be real-like face images, an auxiliary gender classifier ensuring that the gender attribute of face images is confounded, and a face matcher mitigating the impact on the performance of other biometric recognition.

Based on cGAN, Wu et al. [136] designed a model, called **Privacy-Protective-GAN (PP-GAN)**, to preserve soft-biometric attributes during the generation of realistic face with identification. Compared with ACGAN, PP-GAN aims at hiding more soft-biometric attributes instead of gender information. Moreover, Mirjalili et al. [89] proposed a multi-attribute face privacy model, PrivacyNet, based on GAN to provide controllable soft-biometric privacy protection. PrivacyNet allows us to modify an input face image to obfuscate targeted soft-biometric attributes while maintaining the recognition capability on the generated face images.

3.1.2 *Medical Images.*

Today, medical data has been widely applied to medical research but possibly suffers from the leakage of individuals' identification in medical image analysis. To solve this issue, an adversarial training framework of identity-obfuscated segmentation has been proposed by Kim et al. [58]. Their novel DCGAN-based architecture contains three entities: (i) a deep encoder network used as the generator to remove identity features of medical images with the help of additional noise, (ii) a 0–1 classifier used as a discriminator to guarantee similarity between the encoded images and the original images, and (iii) a CNN-based medical image analysis network used as another discriminator to analyze image segmentation content. This design integrates an encoder, a 0–1 classifier, and a segmentation analysis network to protect medical data privacy and simultaneously maintain medical image segmentation performance.

3.1.3 *Street-View Images.*

Street-view services, such as Google Street View and Bing Maps Streetside, typically serve users through collecting millions of images, which some individuals often refuse due to serious privacy concerns [65]. Uittenbogaard et al. [129] designed a multi-view

GAN model based on DCGAN, where the generator is used to detect, remove, and paint in moving objects by using multi-view imagery, and the discriminator is used to make the generated images photorealistic. With these settings, the multi-view GAN removes private regions and is able to retain the utility of the synthesized street-view images. Similarly, Li et al. [73] proposed the PicPrivacy model [73] to segment and erase sensitive information, such as human portraits, from street-view images while repairing blank regions based on GAN to maintain the performance of 3D construction. In addition, to defend against location inference attacks on vehicular camera data, Xiong et al. [140, 141] proposed three **Auto-Driving Generative Adversarial Network (ADGAN)** models based on DCGAN to generate privacy-preserving vehicular images and videos for autonomous vehicles. The core idea of their three methods is to prevent the location-related background information in images/videos from being identified by attackers and maintain data utility simultaneously. The generator takes original data as input and outputs the privacy-preserving data, and multiple discriminators are constructed following the convolutional 0–1 classifier structure with different filter sizes to distinguish real/fake data more efficiently. Additionally, for the trade-off between privacy and utility, customized privacy loss and utility loss are calculated through the difference between the original data and the generated data. To improve model performance and data quality, an extra target model was added in the work of Xiong et al. [140] to provide more accurate feedback on data generation.

3.1.4 Image Stegangraphy. With the widespread **Internet of Things (IoT)** applications in recent years, the risk of privacy leakage has increased. Traditionally, steganography is a critical method to find the trade-off between personal privacy disclosure and covert communication. A new steganography algorithm is developed based on image-to-image translation using a cyclic DCGAN framework, where G_1 is a steganography module transferring data from x_1-domain to x_2-domain, and G_2 is another steganography module transferring data from x_2-domain to x_1-domain. These two steganography modules are used as two generators in Steganography-CycleGAN so that the stego images generated by the proposed method will be close to the cover images. Two discriminators D_{x_2} and D_{x_1} are used to make sure that not only the stego images from x_1-domain to x_2-domain but also the stego images from x_2-domain to x_1-domain are similar to the real ones. In addition, to resist detection by the steganalysis module, one steganalysis module D_S is deployed to realize the concealment and security in transmission by an adversarial training. Similarly, Shu et al. [115] applied GAN to successfully achieve the encrypted rich-data steganography during transmission in wireless networks.

3.1.5 Image Anonymization. Protecting individuals' data privacy is an essential task for public data collection and publication. In the work of Kim and Yang [62], a privacy-preserving adversarial protector network (PPAPNet) was designed as a DCGAN-based anonymization method that converts a sensitive image into a high-quality and attack-immune synthetic image. Under PPAPNet, the generator is initialized as a protector with pre-trained Autoencoder, the discriminator is the WGAN [43] critic with a gradient to guide the protector to generate realistic images that can defend model inversion attacks, and a noise amplifier inside the protector plays a vital role in noise optimization for effective image anonymization. Similarly, Lee et al. [68] implemented face anonymization on the drone patrol systems to hide the sensitive information in face images by converting a sensitive image into another synthetic image based on DCGAN.

3.1.6 Image Encoding. A formula was established for learning an encoding function based on DCGAN in the work of Pittaluga et al. [98]. The encoder is trained to prevent privacy inference and maintain the utility of predicting non-private attributes. In this adversarial framework, the generator is an encoding function that outputs limited information of private attributes while preserving non-private attributes, and the discriminators are neural network–based estimators for privacy protection. Similarly, other works [27, 94, 147] attempted to learn representations or dimension-reduced features from raw data based on DCGAN, in which the desired variables are maintained for utility representations, and the sensitive variables are hidden for privacy protection. Especially, these representations encoded from users' data can exhibit the predictive ability and protect privacy.

3.2 Video Data Privacy

There is an increasing concern in computer vision devices invading users' privacy by recording unwanted videos [87, 107, 134, 141]. On the one hand, some previous works that focus on image privacy can be applied to hide sensitive content in videos by simply considering a video as a sequence of image frames [62, 147]. On the other hand, videos can also be used to recognize important events and assist humans' daily lives by advanced deep learning models, which differ from image recognition applications. Thus, it is expected that in videos, individuals' privacy should not be intruded and the efficiency of detecting continuous action should be kept at the same time. A video face anonymizer is built based on the DCGAN model by Ren et al. [104], where the generator is designed as a modifier of the face in videos, and two discriminators, including one 0–1 classifier and one face classifier, are deployed. Specifically, the 0–1 classifier is applied for adversarial training, and the face classifier is used for face detection. As a result, the video face anonymizer can finally hide the faces but maintain the information used for action recognition.

3.3 Textual Data Privacy

To implement privacy protection using GAN for textual data, various schemes for anonymous text synthesis [74, 112] and privacy-preserving public/medical records release [23, 66, 97, 123, 143] have been proposed.

3.3.1 Texts.

NLP [52] enables author identification of anonymous texts by analyzing the texts' stylistic properties, which has been already applied to describe users by determining their private attributes like age and/or gender. Shetty et al. [112] proposed an **Author Adversarial Attribute Anonymous Neural Translation (A4NT)** with a basis of DCGAN to defend NLP-based adversaries. The objective of A4NT is to fool the identity classifier by altering the semantics of the input text in person while maintaining semantic consistency. To this end, the A4NT network is designed as a style-transfer network that transforms texts into a target style based on **long short-term memory (LSTM)** [120] and fools the attribute classifier simultaneously. The generator in an A4NT network transforms the input texts from a source attribute class to generate the style of a different attribute class. One discriminator uses a 0–1 classifier to help the generator hide the author's identity, and the other discriminator makes the text semantic consistent by minimizing the semantic and language loss. With a similar idea, Li et al. [74] presented a method to learn text representations instead of texts, preserve users' personal information, and retain text representation utility.

3.3.2 Public Records.

When sharing records with partners and/or releasing records to the public, traditional approaches perform privacy protection by removing identifiers, altering quasi-identifiers, and perturbing values. In the work of Park et al. [97], the DCGAN architecture with an auxiliary classifier is exploited to develop a model called *AC-DCGAN*, where the generator produces synthetic records to hide sensitive information. Through adversarial training, the auxiliary classifier is used to predict synthetic records' labels such that the records' identification cannot be re-identified, and the discriminator is trained to ensure that the synthetic records have similar distributions of real records. Thus, the fake and synthetic records can defend re-identification attacks [15] while achieving high utility.

3.3.3 Medical Records. Accessing **Electronic Health Record (EHR)** records data has promoted computational advances in medical research and raised people's privacy concerns about their EHR data. Choi et al. [23] constructed a **Medical Generative Adversarial Network (medGAN)** based on the basic structure of GAN to generate privacy-preserving synthetic patient records. MedGAN can generate high-dimensional discrete variables, in which an autoencoder network is used as the generator to produce the synthetic medical data with the help of additional noise, and a 0–1 classifier is used as the discriminator to ensure data similarity. As a result, the synthetic medical data is applicable to distribution statistics, predictive modeling, medical expert review, and other medical applications. A limited privacy risk in both identity and attributes can be achieved using medGAN. Moreover, to improve the performance of privacy protection of medGAN, the evaluation of privacy-preserving medical records of medGAN was investigated by Yale et al. [143]. CorGAN was developed by Torfi and Fox [123] taking into account the correlations of medical records, and a dual-autoencoder was configured as the generator in medGAN to generate sequential EHRs instead of discrete records for higher predictive accuracy to assist medical experts [66].

3.4 Speech Data Privacy

The works on privacy-preserving speech data based on GAN mainly focus on two fields: remote health monitoring [130] and voice assistants in IoT systems [6].

3.4.1 Remote Health Monitoring. Remote health monitoring has been introduced as a solution to continuous diagnosis and trace of subjects' condition with less effort, which can be partially achieved by passive audio recording technology that may disclose subjects' privacy. Vatanparvar et al. [130] designed a GAN-based speech obfuscation mechanism for passive audio recording when using remote health monitoring. In this speech obfuscation model, the generator is employed to map the audio recording into the distribution of human speech audio and filter the private background voice, and the discriminator is one 0–1 classifier to determine the probability of human speech presence within the audio. After the adversarial training, the synthetic audio recording can be obtained to match the human speech distribution for medical diagnosis and avoid the trace of private information.

3.4.2 Voice Assistance. Voice-enabled interactions provide more human-like experiences in many popular IoT systems. Currently, many speech recognition techniques are developed to offer speech analysis services by extracting useful information from voice inputs as the voice signal is a rich resource containing various states of speakers, such as emotional states, confidence and stress levels, and physical conditions. With the voice signal, service providers can build a very accurate profile for a user through the voice, which, however, may lead to privacy leakage. In the work of Aloufi et al. [6], a cyclic GAN model was built to translate voice from one domain into another domain to hide the users' emotional states in voice, in which the generators are used to do voice translation, and the discriminators are used to force generators to produce the synthetic voice with high quality. The synthetic voice can still be successfully exploited to perform speech recognition for voice-controlled IoT services while resisting inference on users' emotional states.

3.5 Spatio-Temporal Data Privacy

The popularity of edge computing accelerates the emergence and innovation of IoT applications and services. Since various spatio-temporal data need to be collected from IoT devices (e.g., GPS)

Table 1. Comparison of GAN-Based Mechanisms for Data Privacy Protection

Literature	Application	Input	Output	Model	Data Utility	Data Privacy
[20]	Expression Recognition	Face Images	Synthetic Face Images	VGAN	Expression Recognition	Identity
[148]	Face Image Synthesis	Face Images	Synthetic Face Images	TIP-IM	Face Synthesis	Identity
[18]	3D Face Image Synthesis	Face Images	3D Synthetic Face Images	ADGAN	3D Face Synthesis	Identity
[88]	Face Recognition	Face Images	Synthetic Face Images	ACGAN	Face Recognition	Gender
[89, 136]	Face Recognition	Face Images	Synthetic Face Images	PP-GAN	Face Recognition	Soft-Biometric Attributes
[58]	Medical Image Analysis	Medical Images	Synthetic Medical Images	DCGAN	Image Segmentation	Identity
[73, 129]	Street Image Synthesis	Street Images	Inpainted Street Images	DCGAN	Street Image Synthesis	Private Regions
[140, 141]	Autonomous Vehicles	Camera Data	Perturbed Camera Data	ADGAN	Camera Data Synthesis	Location
[87, 115]	Image Steganography	Images	Steganographic Images	Cyclic DCGAN	—	–
[62, 68]	Image Anonymization	Images	Anonymized Images	DCGAN	—	–
[27, 94, 98, 147]	Image Encoding	Images	Image Representations	DCGAN	—	–
[104]	Action Detection	Video	Face-Anonymized Video	DCGAN	Action Detection	Face
[112]	Text Synthesis	Texts	Synthetic Texts	A4NT	Text Synthesis	Identity
[74]	Text Representation	Texts	Text Features	LSTM-GAN	Text Representation	Identity
[97]	Record Release	Public Records	Synthetic Records	AC-DCGAN	Record Synthesis	Identity
[23, 123, 143]	Medical Record Sharing	EHR Records	Synthetic EHR Records	MedGAN	Record Synthesis	Identity
[130]	Health Monitoring	Audio	Synthetic Audio	Obfuscation	Audio Synthesis	Background Audio
[6]	Voice Assistance	Voice Signal	Synthetic Voice	Cyclic GAN	Voice Synthesis	Emotional States
[105, 152]	Data Sharing	Mobile Data	Synthetic Mobile Data	Perturbation	Mobile Data Synthesis	Sensitive Information
[101]	Location-Based Services	Trajectories	Synthetic Trajectories	LTSM-TrajGAN	Trajectories Synthesis	Identity
[31, 69, 70]	Graph Sharing	Graph	Anonymized Graph	Perturbation	Graph Synthesis	Communities
[72]	Graph Embedding	Graph	Graph Representations	APGE	Representations Synthesis	Private Attributes

and the data contains a lot of users' sensitive information, privacy issues are raised unavoidably [105, 152].

In the work of Yin and Yang [152], a GAN-based training framework was designed to protect data privacy in two real-world mobile datasets, where the generator is trained to learn the features of data for privacy-preserving sharing, and the discriminator is used to guarantee the utility of the synthesized data. Considering the limited computation capacity of IoT devices, Rezaei et al. [105] created a privacy-preserving perturbation method that can efficiently run on IoT devices by combining a deep learning network and the basic structure of GAN. They implemented one generator to add noise and two discriminative classifiers (including a target classifier and a sensitive classifier) for adversarial training. More concretely, the target classifier attempts to maintain the utility of the mobile data. The sensitive classifier tries to help hide sensitive information during the data generation process, aiming to find a good trade-off between utility and privacy for mobile data in IoT. In location-based services, the privacy of spatio-temporal trajectories submitted from IoT devices was studied by Rao et al. [101]. The authors proposed an LTSM-TrajGAN model based on DCGAN to generate privacy-preserving synthetic trajectory data, in which the generator is based on an LSTM recurrent neural network trained by minimizing the spatial and temporal similarity loss and the discriminator is a 0–1 classifier for performing the adversarial training. LTSM-TrajGAN can produce privacy-preserving synthetic trajectory data to prevent reidentification of users and preserve the essential spatial-temporal characteristics of trajectory data.

3.6 Graph Data Privacy

The graph data (e.g., social networks) promotes the research and applications of data mining, but privacy leakage in graph data is also becoming more serious during data processing and sharing procedures. Although the traditional anonymization methods for the graph data can balance data utility and data privacy to some extent, these methods are vulnerable to the state-of-the-art inference approaches using graph neural networks [71]. Therefore, more powerful strategies are desired to defend inference attacks for graph data.

3.6.1 Graph Sharing. Fang et al. [31] developed a Graph Data Anonymization using the Generative Adversarial Network (GDAGAN) that exploits the LSTM-based generator for data generation and the 0–1 classifier-based discriminator for utility guarantee. In addition, Laplace noise is added into the synthetic graph for perturbation to protect privacy before publishing graph data to the public. The idea of an adversarial graph has been extended in the work of Li et al. [70] to consider both the problems of imperceptible data generation and community detection for an enhanced privacy protection. The proposed GAN-based model of Li et al. [70] has three critical components: (i) a constrained graph generator based on a graph neural network to generate an adversarial graph, (ii) a 0–1 classifier working as the discriminator to make the synthetic graph real-like for maintaining utility, and (iii) a community detection model that helps the adversarial graph prevent community detection attacks. Similarly, the graph feature learning model of Li et al. [69] was designed based on GAN to perturb a probability adjacency matrix with the help of Laplace noise in the graph reconstruction process to obtain an anonymous graph. This reconstructed anonymous graph maintains the utility of link prediction due to GAN's good feature learning ability, and can be used to defend community detection and de-anonymization attack owing to the utilization of Laplace noise.

3.6.2 Graph Embedding. It is well known that graph embedding is useful to learn low-dimension feature representations for various prediction tasks. Adversaries can also infer sensitive information from these graph node representations, resulting in privacy leakage. Li et al. [72] designed an **Adversarial Privacy Graph Embedding (APGE)** training framework based GAN to remove users' private information from the learned representations of graph data. In APGE, one autoencoder-based generator is used to learn graph node representations while implementing disentangling, and purging mechanisms. During the process of adversarial training, one 0–1 classifier is employed to make the synthetic representations real-like, and one non-private attribute prediction model and one private attribute prediction model are designed to keep data utility and protect users' privacy, respectively.

The comparison of surveyed approaches for data privacy is summarized in Table 1.

4 PRIVACY OF MODELS

In the previous section, we discussed the works on the privacy issues of various sensitive data. It is worth noting that privacy can be inferred not only through data but also through the adopted models, especially in **Machine Learning as a Service (MLaaS)** [106]. As analyzed in the work of Fredrikson et al. [36], a model's privacy breaches if an adversary can use the model's output to infer the private attributes used to train the model. This section will survey how to steal privacy from the learning models and how to protect the learning model privacy using GAN-based approaches.

4.1 Membership Privacy

Membership inference attacks can be launched toward a machine learning and/or deep learning model to determine if a specific data point is in the given model's training dataset or not [114]. Typically, after a model is trained, an attacker feeds data into the model and gets the corresponding prediction results that can be used as additional knowledge to perform black-box membership inference attacks. Such an attack will cause privacy leakage and even other severe consequences. For instance, with a patient's medical records and a predictive model trained for a disease, an attacker can know whether the patient has a certain disease by implementing membership inference attacks. To defend against such attacks, the techniques of GAN, anonymization, and obfuscation have been exploited to design countermeasures [4, 29].

4.1.1 Attacks on Membership Privacy. Membership privacy of generative models was studied for the first time by Hayes et al. [46], in which a model named *LOGAN* was designed to attack a generative model through either black-box or white-box via released API in MLaaS. In white-box attacks, an attacker is assumed to know the target GAN model's structure and parameter consisting of a generator and a discriminator. It is known that the discriminator is able to assign a higher probability to a data point that is in the training dataset. Accordingly, an attacker inputs several data points into the discriminator obtaining the corresponding probabilities and selects the n most probable data points as the n members of the training dataset. For black-box attacks, membership privacy can be inferred without or with auxiliary knowledge. If there is no auxiliary knowledge, an attacker uses the generator of the target GAN from query API to produce enough generated data labeled as "real" for training a local GAN such that he/she can get a parameterized discriminator. Then, the attacker performs the aforementioned white-box attacks on his/her discriminator to learn membership privacy. If the attacker knows some auxiliary information, such as the data only from the training dataset or the data from both the training and test datasets, the attacker's discriminator can be better trained and used in while-box attacks for membership inference. However, there are too many assumptions in black-box attacks, which may be impractical in real applications.

Sum et al. [76] proposed "co-membership" attack toward generative models. Unlike the previous works that infer the membership of a single data point each time, the co-membership attack aims to simultaneously decide whether n ($n \geq 1$) data points are in the training dataset of the target generative model. The implementation of co-membership attacks comes from the intuitive understanding of GAN: if GAN is powerful enough, it should be able to generate any data from a latent vector or reconstruct any data from its latent representation. To accomplish such attacks, a neural network is trained on the attacker side with the following objective: $\min_\gamma \frac{1}{n} \sum_i^n \Delta(x_i, G(A_\gamma(x_i)))$, where γ is a network parameter, G is the generator of GAN that takes a latent vector z as input and outputs generated data $G(z)$, and A_γ is the attacker's network that takes original data x_i as input and outputs a low-dimensional vector with a shape same as z. After the training process is finished, the attacker gets a distance (e.g., L_2 distance) $\Delta(x_i, G(A_\gamma(x_i)))$ between x_i and $G(A_\gamma(x_i))$. If the distance is greater than a threshold, the reconstruction of x_i from z is not associated with the original x_i, which means x_i may not be a member of the training dataset; otherwise, x_i is a member of the training dataset. The proposed attack method has some fatal flaws: (i) for a large training dataset, if the pre-determined value of n is small, the attack accuracy is lower than that of the traditional membership attacks, and (ii) for each victim model, an attacker needs to train a different attack network A_γ from randomly initialized weights. So the attack efficiency is not as high as expected. In addition, the proposed co-membership attack is a kind of white-box attack and is impossible to be used in a black-box scenario.

4.1.2 *Protection of Membership Privacy.*

To prevent membership inference attacks, Nasr et al. [93] proposed an end-to-end method that trains a machine learning model with membership privacy protection using adversarial regularization based on GAN, which enables a user to train a privacy-preserving predictive model on the MLaaS platforms (e.g., Google, Amazon, and Microsoft). Their proposed method contains two parties: an attacker h and a defensive classifier f. The attacker's privacy gain from the victim model f is defined as follows:

$$G_f(h) = \mathop{\mathbb{E}}_{(x,y)\in D(X,Y)}[\log(h(x,y,f(x)))] + \mathop{\mathbb{E}}_{(x,y)\in D'(X,Y)}[\log(1 - h(x,y,f(x)))], \tag{10}$$

where (x, y) is a data point, $D(X, Y)$ is a training dataset, and $D'(X, Y)$ is the set of data that is not in $D(X, Y)$. By maximizing Equation (10), an attacker can obtain an accurate prediction on all data points and know if they are in the training dataset. A defender's loss function is formulated as $\min_f(L_D(f) + \lambda \max_h G_f(h))$, where L_D is the normal loss when training a classifier (e.g., cross entripy), and λ is a parameter to adjust the trade-off between utility and privacy. In this method, the defensive classifier has two objectives: (i) minimizing the normal loss function of f and

(ii) reducing the inference gain $G_f(h)$. Moreover, $G_f(h)$ also works as a regularization term controlled by λ, improving the generalization capability of classifier f. The min-max optimization can train a private classifier even if the attacker has the strongest inference gain. Nevertheless, the proposed method has its disadvantages. As shown in the experiments, the trained classifier's classification accuracy is decreased by around 3% compared with the classifiers without an adversarial regularization term. Another flaw is that the training process of the proposed method requires much data as a reference dataset. In practice, it is hard to obtain so much data with the same distribution as the data given by users in MLaaS, which lowers the applicability of the proposed method.

Wu et al. [135] investigated the generalization ability of GAN from a novel perspective of privacy protection. They theoretically analyzed the connection between the generalization gap and the membership privacy for a series of GAN models. Motivated by a well-known intuition [150], "the smaller the generalization gap is, the less information of the training dataset will be revealed," they linked the stability-based theory and differential privacy [29], which illustrates that a differentially private training mechanism not only reduces the membership privacy leakage but also improves the generalization capability of the model.

4.2 Preimage Privacy

Some other attacks (e.g., model inversion attacks [35] and data reconstruction attacks [34]) move one more step and can cause more serious damage to machine learning models. In model inversion attacks, given a target model f and a label y_t, the purpose of an attacker is to retrieve the input x of the target model f such that $f(x) = y_t$. Similarly, in data reconstruction attacks, an attacker focuses on recovering the raw data in the training dataset of a given model f with the help of additional information. The objective of these two types of attacks is to find private information of input data of learning models, for which we propose a new term called *preimage privacy* to depict model inversion attacks and data reconstruction attacks.

4.2.1 *Attacks on Preimage Privacy.*

Model inversion attacks have been successfully conducted as a severe threat under the white-box setting, whereas for the black-box scenario, there are no impressive works before the birth of GAN. In the work of Aïvodji et al. [3], a model inversion attack framework was built under the black-box setting. Given a target model f and a label y_t, an attacker aims at characterizing data x_t belonging to y_t. To achieve the goal, an attacker trains a GAN framework using adaptive loss in BEGAN [13], where the generator G works as a data inverter (i.e., $G : z \rightarrow x$), and the discriminator D is replaced by a neural network classifier taking x as input and predicting a label y as output. Since there is no real data, a randomly sampled dataset $X_D \sim \mathcal{N}(0, 1)$ is used for self-adaptive updating based on BEGAN. The training process of an attack is expressed as follows:

$$\min_G H(f(G(z)), y_t), \tag{11}$$

$$\min_D H(D(X_D), f(X_D)) - k_t H(D(G(z)), f(G(z))). \tag{12}$$

In Equation (11) and Equation (12), H denotes the cross-entropy loss, and k_t is a parameter of self-adaptive updating. This attack method can efficiently attack the black-box model even though the model is trained with a differential privacy mechanism, providing much inspiration to future research. But this method has an essential flaw that is common for all black-box attacks: an attacker has to issue lots of queries to get a predicted label of input data as auxiliary training information, bringing a huge cost. For example, in this work, 1,280,000 queries are needed to achieve the desired attack performance. Such a frequent and intensive query operation may be detected by the target model easily. Thus, there should be some other ways to improve the attack method.

Moreover, when a target model (e.g., a neural network) has high-dimension input, it is difficult to obtain an optimal solution only with the given label y for the attack model, making the resulted x like a random noise in high-dimension space. Thus, the results of GAN-based model inversion attacks usually lead to unrecognizable representations that are not useful to attackers in reality.

To address this problem, Basu et al. [11] proposed another white-box attack method, where an attacker is able to access the target model and know the domain information of the target model (e.g., the target model is trained on a human face or optical character recognition). With the domain information, an attacker can establish a GAN model to search correct representations in a quite low-dimension space by the generator G, formulated in Equation (13):

$$\hat{z} = \arg\min_{z} L(f(G(z)), y) + \lambda R(z), \tag{13}$$

where L is the loss function of the target model, λ is a parameter, and $R(\cdot)$ is a regularization term. When an attacker learns the domain information, he can grab sufficient data in that domain as the training data from public data sources, such as the Internet. The grabbed data is used as the real dataset to train a traditional GAN model. After being trained, the generator G is used in Equation (13) to obtain an optimal low-dimension input z. Essentially, GAN acts as a transmitter to transfer a high-dimension problem into a low-dimension problem. Then the model inversion result can be generated with $\hat{x} = G(\hat{z})$ quickly. This method's attack efficiency is impressive as shown by the authors and does solve the problem of unrecognized representations.

Later, an improved version of model inversion attacks was developed in the work of Zhang et al. [157]: **Generative Model Inversion (GMI)**, which is similar to that in the work of Basu et al. [11] but more powerful. The major merits of GMI lie in two aspects: (i) it can perform inversion attacks successfully even if the distribution of the attacker's prior information is different from that of the training dataset, and (ii) an improved objective function makes attackers stronger. The implementation of GMI has two phases: *public information distillation* and *secret revelation*. In the first phase, an attacker trains a WGAN model on public information so that the trained generator can be used to recover realistic data. In the second phase, an attacker optimizes the latent vector z (i.e., the input of the generator) via $\hat{z} = \arg\min_z L_{prior}(z) + \lambda L_{id}(z)$, where $L_{prior}(z)$ is used to penalize unrealistic data, and $L_{id}(z)$ encourages the generated images to have maximum likelihood under the target model. Additionally, it has been proved that the more accurate a model is, the easier it is to be attacked, which indicates a trade-off between accuracy and security vulnerability for a learning model.

4.2.2 Protection of Preimage Privacy. To preserve preimage privacy in the MLaaS scenarios, GAN-based mechanisms have been utilized to preprocess private data before the training stage. As shown in Figure 4, the **Compressive Privacy Generative Adversarial Network (CPGAN)** [128] contains three modules: (i) the generator G that is a privatization mechanism for generating privacy-preserving data, (ii) the service module S providing prediction service with the predicted label Y as utility, and (iii) the attacker module A that is a mimic attacker aiming at getting the reconstructed data X' using Z. More specifically, producing Z in G requires that the prediction service $S(Z)$ should perform well and the reconstruction error of $A(Z)$ should be large even if the attacker A is the strongest, based on which the objective of CPGAN can be formulated by Equation (14):

$$\max_{G}[\min_{A} L_A(X, A(G(Z))) - \lambda \min_{S} L_S(S(G(X)), Y)]. \tag{14}$$

CPGAN can defend preimage privacy attacks in MLaaS because the input data of S does not contain any sensitive information. However, the generator G directly accesses sensitive data, which introduces potential privacy threats to G. Furthermore, the unstable training property of GAN makes the optimization process hard to converge.

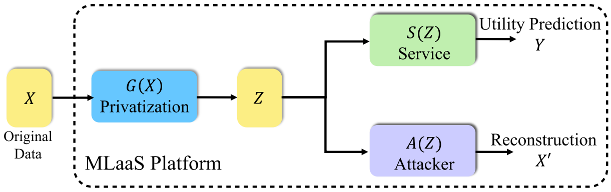


Fig. 4. The framework of CPGAN.

4.3 Privacy in Distributed Learning Systems

As aforementioned, *membership privacy* and *preimage privacy* can be maliciously inferred in an end-to-end centralized machine learning system [8, 36] because all sensitive information is held by a central server that is likely to suffer single point failure. To address this issue, decentralized learning systems become promising solutions, in which the geographically distributed data is trained by different participants locally without data sharing. Two most popular distributed learning schemes are **distributed selective SGD (DSSGD)** [113] and federated learning [85]. In DSSGD, local models only share and exchange a small fraction of parameters through a remote server. In federated learning, the server aggregates parameters using the submitted locally trained parameters. Both methods can train a global model built on a server with an accuracy comparable to that in the centralized learning system [113]. Although distributed learning can protect data privacy to some extent as no one could have a global view of all training data, it is still far from perfect.

4.3.1 Privacy Attacks in Distributed Learning Systems. Since the generator of GAN can mimic data distribution, a well-designed GAN-based model can threaten data privacy in distributed learning scenarios. Besides membership privacy and preimage privacy, more challenging privacy issues should be handled for distributed learning. For example, attackers can invade a server or local users to steal parameters, performing malicious attacks. Under such situation, more private information needs to be protected, e.g., which data belongs to which local user, which user participate in the distributed training process, and how to identify/defend malicious servers or local users who pretend to be trusted.

The first attack method targeting DSSGD was proposed by Hitaj et al. [47] using GAN as an attacker, in which an attacker pretends to be an honest local model within the distributed training system with a goal of recovering sensitive information of a specific label that he does not have. Without compromising the central server or any local models, the attacker only uses parameters shared by other models and some common information (i.e., all class labels in a training dataset) to build a local dynamic GAN. Unlike model inversion attacks, the attacker can update its GAN in real time to adjust attack performance as long as the entire training is not stopped. The attacker's key principle is to share a crafted gradient to the central server to push a victim model to upload more local data information.

In distributed learning systems, the central server may be untrusted as well. As studied in the work of Wang et al. [133], user-level privacy in the distributed learning systems can be revealed invisibly by the malicious server via training a multi-task GAN with auxiliary identification (mGAN-AI) without affecting system performance. In mGAN-AI, G is a conditional generator outputting fake data with random noise and data label as input, and D is a multi-task classifier built from the shared model in the distributed learning systems. When D is under training, except for the last layer, the shared model with additional three parallel fully connected layers is copied to D for the purposes of data generation, categorization, and identification. To find an optimal solution, G is trained by minimizing the loss of data generation, classification, and victim identification, whereas

D is trained by maximizing the loss of data generation and victim identification. After the training process, the attacker can use *G* to generate sensitive information of any target victim model. In addition, a more powerful active attack is provided, in which the malicious server allocates an isolated model to a victim model without sharing any model. In this case, *D* in the attacker's GAN model is exactly the same as the model uploaded by the victim. As a result, the attacker can reconstruct *G* to produce more accurate data without influence on other local models. But this active model actually affects the protocol of distributed learning and decreases learning performance compared with original attacks. Moreover, there is a very strong assumption: the server has a global dataset. If this is true, there is no need to use GAN for attack implementation.

4.3.2 Privacy Protection in Distributed Learning Systems. Privacy protection in distributed learning systems also deserves our attention. Yan et al. [144] designed a protection mechanism against two different attack behaviors, including stealing user information and attacking server parameters. In their protection mechanism, except for the attacker, every local model is embedded with an additional layer called the *buried point layer* and all of its weights are set to be 0. When an attacker starts to attack, for the sake of an unknown buried point layer, the parameters uploaded to the server should be different from harmless local models. At the server, a detection module is used to detect abnormal changes. If the attacker uploads parameters to the server, the detection module immediately discovers the intrusion. When an intrusion is detected for the first time, the link between the attacker and the server is awaited for a check, and when an intrusion is detected for the second time, the connection is blacklisted.

A few current works focus on the extension of GAN to a federated scenario. Due to the constraint that no raw data can leave its local dataset, federated learning is somehow restricted to train classifiers only and thus cannot be used in other important applications, such as data generation and reinforcement learning, especially on small datasets. The integration of GAN and federated learning can realize distributed data generation, improving traditional federated learning's applicability. Generally speaking, federated GAN's objective is to obtain a global generator at the server to produce realistic data following the data distribution of local clients without privacy leakage. In the work of Hardy et al. [45], the generated data and corresponding errors are exchanged for data generation among a generator at the server and distributed discriminators at local clients. Similar updating rules are adopted by Yonetani et al. [153] for data generation in a non-i.i.d. setting by assigning different weights to local discriminators at the aggregation stage. Rasouli et al. [103] built the federated GAN in another way that trains both the generator and the discriminator locally on each private dataset and employs the server as a parameter aggregator and distributor.

4.4 Differential Privacy in GAN

According to the analysis in Sections 4.1 through 4.3 and the conclusion of Song et al. [116], we can find that the root cause of privacy leakage of models is that machine learning models remember too much. In other words, during the training process of a model, the model parameters are optimized to fit the underlying training dataset, which implies that the information of training data (e.g., distribution, features, membership) is embedded into the model parameters. Therefore, the adversary can unveil private information by exploiting the model parameters.

So far, two types of solutions have been proposed to overcome this vulnerability in learning models. The first method is adding a regularization term in a loss function to avoid overfitting during the training process. The regularization item can improve robustness and generalize a model to work on the data that the model has never seen. For example, this method can be used to defend membership inference attack [93] as described in Section 4.1.2. The second method is to add acceptable noise into model parameters to hinder privacy inference attacks. Such type of

Table 2. Comparison of GAN-Based Mechanisms for Model Privacy Protection

Literature	Purpose	White/Black Box	Scenario	GAN Model	Requirement	Model Privacy
[46]	Attack	Both	MLaaS	GAN	Partial training data	Membership
[76]	Attack	White	Centralized	GAN	Partial training data	Membership
[93]	Protection	–	MLaaS	GAN+Regularization	Large amount of data	Membership
[135]	Protection	–	Centralized	WGAN	Large amount of data	Membership
[3]	Attack	Black	MLaas/Centralized	BEGAN	Multiple query+random dataset	Preimage
[11]	Attack	White	Centralized	GAN	Domain information	Preimage
[157]	Attack	White	Centralized	WGAN	Public information	Preimage
[128]	Protection	–	MLaaS	GAN	Large amount of data	Preimage
[47]	Attack	White	Decentralized	GAN	Target label	Data feature
[133]	Attack	White	Decentralized	cGAN	Malicious server	User information
[45, 103, 153]	Protection	–	Decentralized	GAN	Local training data	Local data privacy
[2, 154]	Protection	–	Centralized	GAN	k, public data	Membership
[78, 139]	Protection	–	Centralized	WGAN	DP	Data feature
[142, 156]	Protection	–	Centralized	WGAN	Public dataset, DP	Data feature
[124]	Protection	–	Centralized	cGAN	Target labels, DP	Membership
[9, 126]	Protection	–	Decentralized	GAN	DP	Data feature
[81]	Protection	–	Centralized	GAN	New defined metrics	Data feature

obfuscation method (e.g., k-anonymity, l-diversity, t-closeness, and differential privacy) has attracted lots of research interests for privacy protection, especially the combination of differential privacy [19] and neural networks [1]. Notably, recent research [135, 150] has illustrated the relation between differential privacy and the overfitting problem: introducing differential privacy noise into model parameters could reduce overfitting, thereby mitigating privacy leakage.

Acs et al. [2] presented a first-of-its-kind attempt to build private generative models based on GAN. In their method, GAN is trained to generate unlimited data for data release with a differential privacy guarantee, which improves the generative model's performance significantly and eliminates the constraints of limited data sources. The proposed method divides the whole training dataset into k disjoint sub-datasets using the differentially private k-means algorithm and trains the local generative models on each sub-dataset separately. In the work of Yoon et al. [154], a PATE mechanism-based model named *PATE-GAN* also adopted a dataset dividing strategy. In PATE-GAN, there is a generator G, a teacher T with k teachers trained on k disjoint datasets, and a student S. First, T is trained with public data to differentiate real/fake as the discriminator. Then T is used as a noisy label generator to produce a differentially private dataset for training student S that is the real discriminator. Especially, G and S work as a couple of networks to generating realistic data with a privacy guarantee. The noticeable thing is that during the adversarial training of GAN, T is also updated with G and S at each iteration, which is better for the discrimination capability of T. The training performance of the preceding approaches is dependent on the value of k. If an appropriate k is chosen, the training process would be benefited; otherwise, the training process would suffer because an inappropriate k is possible to induce uneven clustering and a large privacy budget. To get rid of the restriction of k, Xie et al. [139] and Liu *et al.* [78] proposed another more straightforward differentially private GAN model (DPGAN) by carefully adding designed noise into gradients during the training procedure. Based on the training process of GAN, DPGAN makes several improvements for privacy protection. First, the loss function of WGAN is adopted to generate better results and resist model collapse, which is crucial for considering a trade-off between privacy and utility. Then, at the training stage, Gaussian noise is added into gradient calculation. After that, the updated weights are clipped by an upper bound, which helps achieve a smaller privacy loss. Since the generator has no way to access the original data, there is a waste of privacy budget for adding noise on the generator. Thus, the privacy loss of DPGAN can be further reduced.

To this end, Zhang et al. [156] and Xu et al. [142] proposed two methods adding differential noise on the discriminator only. The basic idea of these two methods is too similar, so we review them here together. At the beginning of training, WGAN is used for pursing stability of the training process. During the training procedure, the gradients of the discriminator are bounded and perturbed using the methods of DPGAN [139]. However, the generated data has low quality, and the proposed models converge slower than the traditional GAN, resulting in excessive privacy loss. The following three solutions were proposed to improve data quality, convergence rate, training stability, and scalability. First, parameter grouping is a common scheme used in differential privacy [44]. A balance between convergence rate and privacy cost could be achieved by carefully grouping the training parameters and clipping over different groups. Second, adaptive clipping can enhance the data quality of using random clipping. Assume that a public dataset can be used to dynamically adjust the clipping bounds to achieve faster convergence and stronger privacy protection. Third, warm starting can save privacy budget for critical iterations, in which a public dataset is used to initialize the models with a good starting point. This work's shortage is that the assumption of an available public dataset is too strong in such a private data release scenario. Until now, all of the generative models with differential privacy focused on unlabeled data, and the work on labeled data had not been addressed.

Triastcyn and Faltings [124] developed a cGAN-based framework for the generation of labeled data. The proposed framework can achieve differential privacy for both the generator and the discriminator by only injecting noise into the discriminator. This framework's advantage is that with the class label as auxiliary information in cGAN, the quality of generated data is extremely high with a low privacy loss. However, since the generative model observes the label information, more information may be leaked to attackers when the model is published. As a result, this framework might only be able to prevent membership inference attacks but fail to protect preimage privacy.

As discussed in Section 4.3, for the distributed GAN, the privacy of generators in federated setting also needs to be protected because the physical separation of data is not enough. More analysis of differential privacy in the private federated GAN models was presented elsewhere [9, 126], where the authors proposed to add differential privacy during the training process of local GAN models. The noise mechanisms also utilize clipping and noisy gradients, which is similar to other methods [139, 142, 156]. To achieve a better trade-off between utility and privacy, a relaxed expected privacy loss is adopted by Triastcyn and Faltings [126], whereas the work of Augenstein et al. [9] utilizes the original *Moments Accountant* strategy. In a nutshell, differential privacy is a classic standard for privacy protection and an efficient approach for preserving the membership and preimage privacy of GAN. But the protection efficacy of differential privacy is greatly determined by the noise scale, which may introduce utility loss and needs more research endeavors.

Not stopping here, a study performed by Lu et al. [81] pointed out that even though there had been much research on private data release by generative models, the adopted quality metrics are not quite suitable. For example, adopting differential privacy may impair data utility more or less. Thus, the users (e.g., companies) often do not adopt the strict privacy guarantee in academic areas. Instead, they only require the privacy level of some mechanisms to be slightly better than the government regulation even in the boundary of law. For those users, a more practical measurement could be considered. In the work of Lu et al. [81], the experiments were set up to evaluate new defined metrics in addition to the privacy budget of differential privacy, such as hitting rate, record linkage, and Euclidean distance. The extensive experiments demonstrate that for formal privacy definition (i.e., differential privacy), even though it achieves strict privacy protection, it loses more data utility. For industrial and commercial applications, informal privacy guarantees, such as GAN-based methods, can meet privacy requirements and have better data utility.

We compare the preceding reviewed methods in Table 2 from different perspectives.

5 SECURITY WITH GAN

Besides privacy issues, a variety of security issues also exist in GAN. In this section, the research findings on security for GAN are introduced from the aspects of model robustness, malware, fraud detection, vehicle security, industry protocol, and so on, and a comparison is presented in Table 3.

5.1 Model Robustness

For machine learning models, one of the most severe secure threats is adversarial sample attack. Let X be the feature space of data and Y be the class space. Suppose for the original data $x \in X$, its ground-truth label is $y \in Y$. For a given classifier $f : X \rightarrow Y$, an adversarial example attack intends to manipulate a sample x' through unperceptive modification to mislead classification, which can be mathematically formulated as the following optimization problem:

$$\min L(x', x), \tag{15a}$$

$$\text{s.t. } f(x') \neq f(x) = y. \tag{15b}$$

Equation (15a) implies the objective of minimizing the distance, $L(x', x)$, between the adversarial data and the original data, where $L(\cdot, \cdot)$ is a pre-defined distance metric (e.g., L_2 norm). Equation (15b) indicates the incorrect classification result on the adversarial data. There are two kinds of adversarial attacks: non-targeted and targeted. Non-targeted attacks only require $f(x') \neq f(x)$, and targeted attacks expect $f(x') = y_t \neq f(x)$ with $y_t \in Y$ being a target label pre-determined by an attacker.

5.1.1 Adversarial Sample Attacks. There have been many works on targeted attacks [92, 96] and non-targeted attacks [41, 138] based on traditional optimization methods. Baluja and Fischer [10], for the first time, generated targeted adversarial samples using GAN to attack machine learning models. Unlike previous works that produce adversarial samples by optimizing a noise δ added into the original data x, the idea of Baluja and Fischer [10] is to train a neural network to obtain an adversarial sample x' directly from the original data x. Such an adversarial generator aims to simultaneously minimize the distance loss in the feature space and the classification loss in the prediction space. Particularly, in the implementation of targeted attacks, an attacker hires a reranking function to resort to the predicted labels such that the target label has a maximum probability and the other labels maintain their original order. Compared with the traditional optimization-based methods, this generative attack mechanism is extremely fast and efficient once the neural network has been trained. However, this attack mechanism is model dependent, which means that it does not perform well in the black-box scenario and lacks transferability.

To deal with the preceding weakness, Zhao et al. [158] suggested using a GAN-based model plus a data inverter to enhance attack capability. They adopted WGAN to generate vivid data from random noise $z \sim \mathcal{N}(0, 1)$ and used the dense representation z to produce realistic adversarial samples, in which a data inverter I was designed to map normal data to the corresponding dense representations $z \sim \mathcal{N}(0, 1)$. With the help of this inverter I, any normal data can be transferred into its corresponding representation $I(x)$ used as input to produce an adversarial dense representation $\tilde{z}$ for sample generation in WGAN. The workflow of this attack is briefly summarized in Figure 5.

In the work of Xiao et al. [137], a target classifier model was seamlessly integrated into GAN to train a stronger attacker in the black-box scenario. The proposed model aims at minimizing the target prediction loss, normal form of GAN loss, and the noise scale simultaneously. When performing a black-box attack, the whole network is updated to obtain a strong classifier f via two steps: (i) fix the target classifier and update the generator and the discriminator in GAN according to the objective, and (ii) fix GAN and update the classifier satisfying

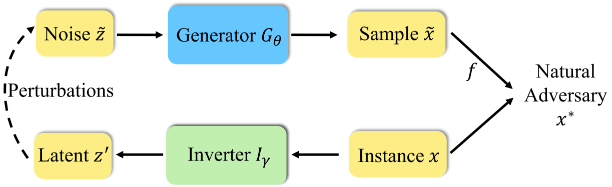


Fig. 5. The structure of a GAN inverter.

arg $\min_f \mathbb{E}_x H(f(x), b(x)) + H(f(x + G(x)), b(x + G(x)))$, where b is the target black-box classifier, f is a shadow model built from observation of b, and $H(\cdot,\cdot)$ is the cross-entropy function. After the training process, the generator in GAN can be used to produce adversarial sample $x + G(x)$. Since the attack model is dynamically trained based on the target classifier's changes, the attack capability is enhanced, and the attack success rate is increased.

Unlike the previous attack methods requiring small adversarial noise for accurate human perception and classification, an unrestricted adversarial attack method was proposed by Song et al. [117] to mislead the victim classifier to the target label y_t using a modified ACGAN model of Odena et al. [95]. In the beginning, a generator $G(z, y)$ is trained to generate synthetic labeled data from random noise z and data label y. To generate the adversarial samples with target label y_t, an attacker tries to optimize z to $\tilde{z}$ until $f(G(\tilde{z}, y)) = y_t$, where f is a given victim classifier. This attack is achieved by two optimization procedures in ACGAN: (i) for the victim classifier f, minimize the loss between $f(G(\tilde{z}, y))$ and y_t, and (ii) for the additional classifier C, minimize the loss between $C(G(\tilde{z}, y))$ and y. The advantage of this attack is that it can generate numerous adversarial samples that could be even different from the original data without being detected by a human.

Similar to Odena et al. [95], Wang et al. [132] designed an AT-GAN model using a pre-trained generator G and ACGAN to transfer G to G_{attack} that can be used to produce adversarial samples directly. In the beginning, an attacker trains $G(z, y)$ to mimic real data distribution with ACGAN. Based on the training results of $G(z, y)$, the attacker slightly modifies $G(z, y)$ to $G_{attack}(z, y)$ by minimizing $||G_{attack}(z, y) - G(z, y)||$ such that $G_{attack}(z, y)$ can produce data with the target label y_t, where the prediction loss, $L(f(G(z, y)), y_t)$, and the distance loss, $||G_{attack}(z, y) - G(z, y)||$, are taken into account for simultaneous minimization. For the desired adversarial samples, the prediction loss assures the modified samples can be classified to the target class, and the distance loss controls perturbation magnitude between real and fake data.

However, as a generative model, GAN itself also faces the danger of being attacked. In the work of Kos et al. [63], a framework was given to show how to design adversarial samples to attack generative models, such as GAN and VAE. Apart from the original adversarial samples in classification models, an adversarial sample x' in the generative model is achieved by minimizing the distance $L(x', x)$ such that $f(G(x')) = y_t \neq f(G(x))$. To realize the attack purpose, the entire loss function, $\lambda L(x, x') + H(f(G(x')), y_t)$, should be minimized, in which $L(\cdot,\cdot)$ controls the distance between x and x', and $H(\cdot,\cdot)$ compels $G(x')$ to be classified to the y_t class. This is the first work to design an attack model for the generative models. However, the proposed attack model can only handle white-box attack implementation, making it infeasible to perform attacks without prior knowledge.

5.1.2 Adversarial Sample Defense. The existing strategies against adversarial sample attacks can be briefly classified into three categories: denoising, adversarial training, and detection. The GAN-based defense methods will be summarized from the preceding categories.

The first denoising method based on GAN was designed by Shen et al. [111] with the fundamental idea that the generator can take adversarial samples to output normal data. Accordingly, a conditional GAN structure was established, where the generator $G(x')$ takes adversarial sample x' as training data and learns the normal data distribution with $G(x') = x$. In other words, the distribution of the generated data should be the same as that of the real data to make the output become clean data. In this GAN framework, the loss function of the discriminator is the same as that in the original GAN, whereas the generator's goal consists of a distance loss that constraints the distance between x and $G(x')$ to be small and an adversarial loss that requires $D(G(x'))$ to have a higher score. A similar idea was exploited by Samangouei et al. [108] to develop a "Defense-GAN" framework that de-noises adversarial samples to obtain normal samples through the generator of GAN. The slight difference is that in their work [108], the denoising generator is trained on clean data from noise z to minimize the distance between input data and generated data. After the optimized z^* is obtained, a reconstructed data $G(z^*)$ is fed into f expecting that the output $f(G(z^*))$ is the same as normal data. The denoising method can be used in conjunction with any classifier and does not need to modify the classifier structure. Thus, it will not decrease the performance of the trained model. In addition, it is independent of any attack method, which means that it can be used as a defense against any attack.

To tackle the issue of insufficient adversarial data in adversarial training, a generative adversarial training method was proposed by using GAN [67]. In this method, the generator $G(\nabla)$ takes the gradient ∇ of normal data x as input and outputs adversarial noise to perturb x. The loss function of G is $L_G(\nabla, y) = H(f(x+G(\nabla)), y) + \lambda\|G(\nabla)\|^2$, where $H(f(x+G(\nabla)), y)$ encourages the generated data $x + G(\nabla)$ to be classified correctly by the classifier f, and $\lambda\|G(\nabla)\|^2$ requests the generated noise $G(\nabla)$ to be small and imperceptible. The desired classifier f was configured as a GAN's discriminator to reduce the classification loss for both the normal data and the generated data. The loss function of the classifier f is $L_f = \alpha H(f(x), y) + (1-\alpha)H(f(x+G(\nabla)), y)$, where $H(\cdot, \cdot)$ is defined as cross entropy. This article presents the first work to apply GAN to adversarial training and provides a robust classifier so that enough adversarial data can be used to enhance the desired model's regularization power effectively. It is worth pointing out that the proposed method is model dependent and can only work on some specific models due to the classifier f's involvement in the adversarial training.

An improvement has been made by Liu et al. [75], who proposed a model-independent method named *GanDef* that can be a defense for many different classifiers. According to their analysis, the misclassification of deep models is caused before the soft-max layer. The input of soft-max layer would be different from adversarial samples and normal samples through forward propagation, which explains why the final output of deep models is different. Moreover, the normal samples hold a property of "invariant features"; in other words, if a set of data is classified into correct classes, the input of soft-max should follow the same distribution. The failure of classification on the adversarial samples indicates that the adversarial samples and the normal samples do not have the same invariant features. Thus, for correct classification, a classifier and a discriminator are deployed in a GAN framework to modify adversarial samples such that their invariant features are the same as normal data. Following the training process of the original GAN, the classifier and the discriminator are iteratively updated. Finally, the classifier is supposed to have the ability to make the adversarial sample's invariant features similar to (ideally, even the same as) the normal samples. Thus, any data can be processed before the soft-max layer with this mechanism and get the correct prediction.

5.2 Malware Detection

Hu and Tan [49] developed MalGAN to generate adversarial malware examples, where GAN was employed as a binary malware feature generator to attack a malware detector. The malware only extracts the program output of detection with the assumption that the attacker only knows the detector's features without the machine learning algorithm the detector uses and the parameters of the trained model. As the trained model details are unkown, a substituted detector is used to fit the black-box detector and provide gradient information. The adopted substituted detector's training data has two parts: the set of adversarial malware examples from the generator of GAN and the set of examples from an additional benign dataset.

In the work of Shahpasand et al. [110], GAN was applied for detecting Android malware. The detection focuses on black-box attacks, where attackers cannot access the inner details of the network (including network architecture and parameters) but can get the classifier's output and alter the malware codes based on detection results. Generally, given a malware x with the true label $f(x) = 1$, the attacker aims to avoiding malware detection via generating an adversarial version $x' = x + \delta$ such that $f(x') = 0$, where $f(\cdot)$ is a malware detector. The loss function includes the similarity between the generated and the benign samples (denoted by L_{GAN}) and misclassification rate of the adversarial malware samples (denoted by $L_{adv.Mal}$), which can be mathematically expressed as $L = \alpha L_{GAN} + (1 - \alpha) L_{adv.Mal}$, where L_{GAN} represents discriminative loss of GAN on real and fake data and $L_{adv.Mal} = l_f(x + G(z), 0)$ is supposed that an adversarial sample can bypass the targeted classifier f by manipulating itself as a benign class.

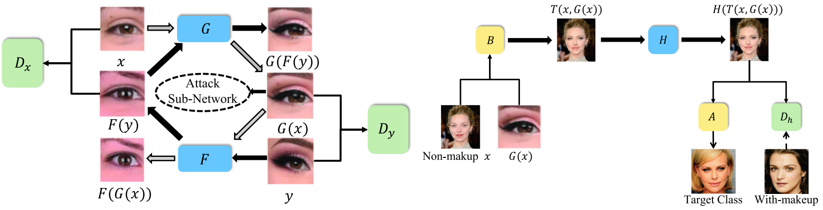
Furthermore, GAN can be utilized for analyzing Linux and Windows malware. Kargaard et al. [55] brought GAN to the analysis of malware detection, where the malware binaries are converted to images for training in GAN. In particular, the malware is collected via a honeypots system and contains WannaCry ransomware, Linux SMB trojan, and MySQL Trojan, and so on. All of the malware binary files are converted to greyscale images with a size of 32×32 for processing. Taheri et al. [121] proposed using federated GAN to defend attack and enable the devices to communicate with each other efficiently and securely. As illustrated in this work, the proposed method is much more effective and reliable than the previous methods.

Among various malware, zero-day malware is outstandingly difficult to be detected because it cannot be removed by antivirus systems that mainly use the characteristics of stored malware for detection. Kim et al. [60, 61] proposed a transferred deep-convolutional generative adversarial network (tDCGAN) that applies a deep Autoencoder to learn malware characteristics and transfers the characteristics to train the generator of GAN. The architecture of tDCGAN has three parts: (i) data compression and reconstruction, in which preprocessed data is input to compress and reconstruct the malware data; (ii) fake malware generation, in which deep Autoencoder is used to reconstruct malware data and the decoder is transferred to the generator of GAN; and (iii) malware detection, in which the generator is given the probability distribution to produce fake malware. Then the discriminator of GAN is transferred to the malware detector.

5.3 Bioinformatic-Based Recognition

Bio-information (e.g., fingerprints and iris) recognition systems have been widely deployed in many areas, such as banking, criminal investigation, and national security. However, the bio-information gathering process is expensive and time consuming. In addition, due to privacy protection legislation, publishing a bio-information database is not easy. Fortunately, GAN provides a novel way to construct bio-information systems for authentication.

Bontrager et al. [14] used GAN to generate synthetic fingerprints for a fingerprint verification system identifying different people. In their work, two methods are designed based on GAN, with the first one applying evolutionary optimization in the space of latent variables and the second one using gradient-based search. In the work of Kim et al. [59], the fingerprints, namely the master minutia set, were generated from a two-stage GAN. The two-stage GAN is composed of two GANs: the first GAN is for generally describing fingerprints, and the second GAN uses the outputs from



(a) Makeup Transfer Sub-network (b) Adversarial Attack Sub-network

Fig. 6. The framework of adversarial attack networks. Source: http://iprobe.cse.msu.edu/datasets.php.

the first GAN to create fingerprint images. Then the minutiae extracted from the second GAN's outputs and a feature extraction algorithm is used to do extraction.

For face recognition, there are also more and more attack methods that generate adversarial examples to violate the deep learning based face recognition models. Zhu et al. [162] applied GAN to make up adversarial example attack on well-trained face recognition models. The proposed method consists of two GAN-based sub-networks, including a "makeup transfer sub-network" that transfers face images from non-makeup domain to makeup domain and an "adversarial attack sub-network" that generates adversarial examples. The configuration of "makeup transfer sub-network" follows CycleGAN [160] as shown in Figure 6(a), in which x is the real non-makeup input, y is the real makeup input, the generator G outputs $G(x)$ to obfuscate the discriminator D_y, and the network $F : Y \rightarrow X$ generates results of non-makeup faces to obfuscate the discriminator D_x. In the "makeup transfer sub-network," one generator adds makeup effect to non-makeup images, and the other removes makeup effect while still maintaining the original identity. However, D_x distinguishes between the real non-makeup photos and the generated ones, and D_y distinguishes between the real makeup photos and the generated ones. The structure of "adversarial attack sub-networks" is presented in Figure 6(b), where a transformation $T(\cdot, \cdot)$ is input with the blend of original image x and the generated output $G(x)$ of "makeup transfer sub-network." Then, $T(x, G(x))$ serves as the input of the generator H, which is used to generate images $H(T(x, G(x)))$ that can deceive both the target network A and the discriminator D_h. In the "adversarial attack sub-networks," the discriminator D_h functions similarly as D_y, and D_y is also used as a pre-trained model parameter to initialize D_h.

In addition, Damer et al. [25] proposed MorGAN to launch a realistic morphing attack by considering the representation loss. MorGAN is inspired by the idea of adversarially learned inference but uses a VAE instead of simple Autoencoder. By doing so, MorGAN can avoid the possibilities of non-continuous latent space and further lead to more realistic output from the interpolation between encoded vectors.

5.4 Fraud Detection

Financial fraud detection problems, including credit card fraud, telecom fraud, and insurance fraud, and so on, are well known for highly non-linear and complex solutions. The artificial neural network, which simulates interacting neurons' properties, has been successfully utilized to solve such problems. However, in real applications for fraud detection, the artificial neural network faces two problems: (i) the receiving bank cannot get access to detailed information about the sending accounts when the transaction happens between different banks, and (ii) the receiving bank also cannot obtain call records of the recipients of transfers from the telecommunication provider.

To deal with these problems, in the work of Zheng et al. [159], GAN was applied to telecom fraud detection under the bank receiving scenario, which is also useful for many other anomaly detection problems when the training dataset is limited. In this work, the authors coupled GAN, Autoencoder, and GMMs for fraud detection. Particularly, the encoder together with a GMM was used as the discriminator, and the decoder was used as the generator. Then the encoder and another GMM output the classification results (i.e., whether a given sample is normal or fraud).

5.5 Botnet Detection

As one of the most formidable threats to cybersecurity, a botnet is often enrolled in launching large-scale attack sabotage [33]. The network-based detection mainly studies the abnormal characteristics of botnets based on network flows. Some other classic detection approaches are based on the extraction and selection of features using statistical analysis, machine learning, data mining, and other methods. However, these traditional detection schemes have two main shortcomings. On the one hand, most of the existing network-based methods for botnet detection are limited to the packet inspection level and focus on partial characteristics of network flows, which cannot fully characterize botnets' abnormal behaviors. On the other hand, botnets keep pace with the times and take advantage of advanced ideas and technologies to escape detection, raising insurmountable challenges to the traditional detection schemes.

The discriminator is essentially a binary classifier that classifies the samples into real or fake categories. Similarly, the real samples can be further categorized into normal traffic or abnormal traffic for a botnet detector. Inspired by these observations, Yin et al. [151] proposed a GAN-based botnet detection framework, which is suitable for augmenting the original detection model. In their work [151], the discriminator was replaced with a botnet detector, and the corresponding binary output (i.e., normal and anomaly) was transformed into a triple output (i.e., normal, anomaly, and fake) for detection using the soft-max function.

5.6 Network Intrusion Detection

Network environment is very complex and time varying, so it is difficult to use traditional methods to extract accurate features of intrusion behaviors from the high-dimensional data samples and process the high-volume data efficiently. Even worse, the network intrusion samples are submerged into many normal data packets, leading to insufficient samples for model training.

Yang et al. [146] proposed a DCGAN-based method to extract features directly from the raw data and then generated new training datasets by learning from the raw data, in which LSTM [39] was applied to learn the features of network intrusion behaviors automatically. The generator G was configured with CNN, where the pooling layer was replaced with the fractional stride convolutions. Then, the fully connected layer was removed, ReLU was applied for all layers except the output layer, and tanh is used at the output layer. In addition, batchnorm was utilized to solve the poor initialization problem and propagate each layer's gradient. The discriminator was also constructed using CNN that contains a pooling layer without any fully connected layer, and LeakyReLU is used at all layers. Batchnorm was used to propagate the gradient to each layer to avoid the generator converging all samples to the same point.

5.7 Vehicle Security

Seo et al. [109] proposed a **GAN-based Intrusion Detection System (GIDS)** for vehicular networks. A **Controller Area Network (CAN)** bus in the networks is an efficient standard bus enabling communication between all Electronic Control Units. However, CAN itself is vulnerable due to the lack of security features. GIDS aims at detecting unknown attacks on CAN with two discriminative models. The first discriminator receives normal and abnormal CAN images extracted from the actual vehicle. Because the first discriminator uses attack data in the training process, the type of attacks that can be detected may be limited to the attack used for training. The generator and the second discriminator are trained simultaneously in an adversarial process, where the generator

Table 3. Comparison of GAN-Based Mechanisms for Security

Literature	Purpose	White/Black Box	Application	GAN Model	Strategy	Target
[10]	Attack	White	Break robustness	GAN	Generating adversarial samples	Misclassification
[158]	Attack	Black	Break robustness	WGAN	Explore latent space z	Misclassification
[137]	Attack	Black	Break robustness	GAN+shadow model	Minimize noise scale by classifier	Misclassification
[117]	Attack	White	Break robustness	ACGAN	Explore latent space z	Misclassification
[132]	Attack	Black	Break robustness	ATGAN	Perturb normal generator	misclassification
[63]	Attack	White	Data generation	VAE-GAN	Modify latent representation	Incorrect generation
[108, 111]	Defense	Black	Data sanitizing	cGAN	Use generator to clean adversarial data	Denoising
[67]	Defense	White	Adversarial training	GAN	Generating adversarial samples	Enhance classifier
[75]	Defense	White	Data generation	GAN	Modify invariant features in adversarial samples	Enhance classifier
[49]	Attack	Black	Malware detection	GAN	Generate adversarial malware examples	Invade detector
[110]	Defense	Black	Malware detection	GAN	Generate noise for malware	Enhance detector
[55]	Defense	White	Malware detection	GAN	Convert malware into greyscale images	Enhance detector
[60, 61]	Attack	White	Malware detection	DCGAN	Generate malware	Invade detector
[14, 59]	Defense	White	Bioinfo recognition	GAN	Generate fingerprints	Bypass verification
[162]	Attack	White	Bioinfo recognition	CycleGAN	Modify face without changing ID	Misclassification
[159]	Defense	White	Fraud detection	GAN	Generate more fraud data for training	Enhance detector
[151]	Attack	White	Botnet detection	GAN	Modify discriminator to a detector	Enhance detector
[146, 162]	Defense	White	Network intrusion detection	DCGAN	Generating intrusion behavior data	Data Augmentation
[50]	Defense	White	Industry protocols	GAN	Generate data for industrial protocol fuzzing	Data Augmentation

generates fake images by using random noise and the second discriminator determines whether its input images are real CAN images or fake images generated by the generator. In GIDS, the second discriminator ultimately beats the generator so that the second discriminator can detect even the fake images that are similar to real CAN images.

5.8 Industry Protocols

In industry, people often use fuzz to detect whether the industrial network protocols are secure. Traditionally, to generate the fuzzing data effectively, the guidance of protocol grammar is applied to the generating process, where the grammar is extracted from interpreting the protocol specifications and reversing engineering in network traces.

The work of Hu et al. [50] employed GAN to train the generation model on a set of real protocol messages for industrial network protocol fuzzing. Specifically, in the GAN framework, an RNN with LSTM cells is used as the generative model, and a CNN was set as the discriminative model. They used the trained generative model to produce fake messages, based on which an automatic fuzzing framework was built to test industrial network protocols. Their experiments showed that since the proposed framework does not rely on any specified protocols, the proposed framework outperforms many previous frameworks. Moreover, some errors and vulnerabilities were identified successfully in a test on several simulators of the Modbus-TCP protocol.

6 FUTURE WORKS

The survey of the state-of-the-art GAN models displays the innovative contributions of GAN to solving the issues of privacy and security in various fields. As a preliminary attempt, GAN's potentials have not been fully and deeply explored by the existing GAN-based approaches yet, leaving many unsolved challenging problems. This section provides a comprehensive discussion to address these challenges and promising directions for future research.

6.1 Future Research on Data Privacy

6.1.1 CT Medical Images. In medical image analysis, there exists one work that implements the GAN-based model on MR images to generate privacy-preserving synthetic medical images while maintaining segmentation performance. However, in reality, CT images are more widely used in medical analysis than MR images. To leverage the advantages of GAN to further benefit medical image analysis in real applications, protecting private information in CT images with a performance guarantee would deserve researchers' attention.

6.1.2 Sequential Records. Although some methods have been proposed to protect public records' privacy before collaborative use, they take these records as discrete data for privacy-preserving processing. These records actually are a kind of sequential data as the relation between two words in a textual sentence is not ignorable. Thus, separating one sentence into words may cause performance loss for data generation and privacy protection. In other words, it is indispensable to take these relations into account when generating privacy-preserving public records, which is one promising research direction for the effectiveness improvement in practice.

6.1.3 Spatio-Temporal Information in Videos and Speeches. In the prior works on privacy protection in videos and speeches, the sensitive information is hidden by noise added via a generator and the data utility is maintained by a discriminator through an adversarial training. A video is treated as a sequence of image frames for generation, ignoring the spatio-temporal relation between frames; a speech is only considered as a distribution of voice information, overlooking the spatio-temporal relation between voice segments. Notably, such spatio-temporal relations can be exploited as side-channel information to mine individuals' private information, especially with the development of deep learning models. As a result, the challenge of incorporating spatio-temporal relations into privacy protection for videos and speeches should be overcome.

6.1.4 Understanding Sensitive Information. Typically, the current GAN-based models add noise into the synthesized data to hide sensitive information through the generator and demonstrate the capability of privacy protection through the experimental results of reduced prediction/classification accuracy. However, in these models, it is still unknown what type of sensitive information is hidden and where the noise is added. This is because the generator is a black-box function of data synthesis, only relying on the discriminator's feedback in the adversarial training process. Such a kind of training mechanism makes the privacy protection inefficient when facing the privacy detectors that are not taken into account by the discriminator(s) in the training process. Therefore, understanding the privacy-related features in source data will be helpful to strengthen privacy protection in GAN.

6.1.5 Guarantee of Privacy Protection. When applied to privacy-preserving data generation, all existing GAN-based methods fail to provide a theoretical guarantee of privacy protection. The root cause is that the generator is a black-box function of data synthesis, and its synthesis performance is mainly determined by the feedback of the discriminator during the adversarial training process. What is worse is that the adversarial training is not stable, resulting in the generator's unpredictable capabilities and the discriminator. To further promote GAN' development and application, technique breakthrough is desired to offer a theoretical guarantee of privacy protection.

6.1.6 Computation Cost. With the increased popularity of emerging applications, such as IoT and edge/fog computing, a large amount of user data is shared through connected devices (e.g., mobile phones), resulting in serious privacy leakage during transmission. To protect data privacy in the connected devices timely before transmission, light-weighted GAN models are expected such that the computation cost (e.g., time and energy) is affordable for these connected devices, in which balancing the trade-off between computation performance and computation cost is an unavoidable challenge.

6.2 Future Research on Model Privacy

6.2.1 GAN Model Improvement. Future works on model privacy are tightly related to the GAN model's improvement, which is a mainstream research direction in the learning field and will continue to attract more research interest. For example, the investigation on the convergence rate and mode collapse of GAN will definitely enhance GAN's efficacy on both attack and defense aspects.

Increasing the convergence rate would save more time and cost for the attacker, as well as provide a more efficient way to train a defense mechanism. Rectifying the problem of mode collapse in GAN can yield a stronger generator with more representation capability, which can either encourage the attacker to recover data with higher quality when stealing preimage privacy or produce a more powerful denoising module for better defense. To understand how to accelerate convergence and how to avoid mode collapse, conducting fundamental theoretical research is essential.

6.2.2 GAN in Privacy Acquisition. Using GAN to proceed privacy-related attack always faces a realistic problem: it requires a lot of real data as prior knowledge for training a well-performed GAN model, which is a strong assumption and is hard to be satisfied in practice. Possible solutions to this problem include *transfer learning* and *probably approximately correct learning*.

Transfer learning. In the real world, some private information is not accessible for the public, but lots of available related data can be used to bridge to privacy. Transfer learning is a promising paradigm to perform knowledge transfer between public data and private data. In light of this idea, transfer learning and GAN can be integrated to establish an approximate GAN model that is very close to the GAN model trained on unknown private data.

Probably approximately correct learning. Until now, GAN has been applied to various attack methods, but there is no theoretical analysis illustrating why these methods can work and/or how good they can be. It is probably approximately correct learning to provide a novel way of carrying out theoretical analysis on the preceding problems, filling in the blank in the literature. As an elegant framework, probably approximately correct learning explores the mathematical analysis of machine learning and computational learning theory. Especially, it can quantitatively analyze some parameters in learning algorithms, including the approximate correctness, the probability of getting approximate correctness, the number of sample needed in learning, and so on. Since no work has studied probably approximately correct learning in GAN, there are many open questions for further investigation. One novel idea is using series theory to derive the necessary conditions/requirement for launching privacy-related attack with GAN, which could tell us if it is worth launching an attack or not, such as what the minimum size of the training dataset is to train a good GAN model with a certain attack success probability, and when there is limited training dataset locally, what the upper bound of the attack accuracy is.

6.2.3 GAN in Privacy Protection. Originally, GAN was born with a private feature: model separation to protect privacy. In other words, during the training process of GAN, only the discriminator can access data directly while the generator is innocent with data. However, with the development of different attack methods, the privacy threats to GAN have been increased.

Differential privacy. Differential privacy has been treated as a golden defensive mechanism but has its defects, such as lower data utility when noise is accumulated. In future work, investigating differential privacy deeply for GAN is still an attractive topic, for which some possible directions are briefly addressed in the following. First, according to the application requirements, different kinds of differential privacy could be explored rather than only using (ϵ, δ)-differential privacy. For example, Renyi-differential privacy and concentrated differential privacy are possible choices. In the work of Beaulieu-Jones [12], Renyi-differential privacy was proposed to achieve a more tighter bound on privacy loss. As pointed by Triastcyn and Faltings [125], Bayesian differential privacy in federated learning is flexible and can save privacy loss significantly for utility critical applications. Second, the convergence analysis of GAN in a differentially private setting should be studied. Current works mainly focus on the final results of trained models and rarely consider the convergence issues, leading GAN's research on an experiment-driven path. In the long term, GAN's development needs the guidance of fundamental theory for accelerated progress in privacy protection.

Last but not least, the use of differential privacy should be granulated at an appropriate level for fine-grained protection, such as for the instance level and the client level. It has been shown in previous work [47] that instance-level differential privacy fails to protect preimage privacy. Thus, in the design of differentially private GAN, more details should be considered for performance improvement.

Federated learning. Federated learning [86] can help relieve privacy leakage when facing a powerful attack. Since the training data in federated learning is geographically distributed among all local non-contact clients, single failure issues can be avoided. In addition, integrating federated learning and differential privacy is not a hard job, which can further improve GAN's capability of privacy protection.

6.3 Future Research on Security in GAN

6.3.1 Adversarial Sample Generation. The adversarial samples generated by GAN can be used to either launch an attack or implement a defense, which has been intensively studied so far. *Attack.*

It is worth mentioning that the actual attack success rate and training cost of GAN-based attack methods are not as good as those of the traditional optimization-based attack methods. Improving these two metrics is closely related to GAN's fundamental research direction (i.e., convergence and mode collapse). The faster the convergence rate, the less is the training cost, and the lower the mode collapse probability, the higher is the attack success rate. Another reason for the low attack success rate is the lack of dedicated objective functions. Among the generated results of a trained generator, some samples are benign while some are adversarial. This is because the space of generative samples is a multi-fold space, where only a partial region may have adversarial attributes. In other words, even though the recall of adversarial samples is high, the precision of being adversarial in the entire space is quite small. Thus, analyzing the generative space to restrict the probability of adversarial samples falling into the benign region from a mathematic perspective is a problem worth thinking about, which is non-trivial and requires thorough investigation. In addition, when the number of training samples is limited, the generated adversarial examples become worse in the presence of a pure black-box classifier because black-box provides less useful information for training GAN. To tackle this problem, as illustrated in Section 6.2.2, probably approximately correct learning is a promising scheme to measure how much data is needed and/or how good the attack can be.

Defense. Among the techniques of adversarial training, detection, and de-noising, adversarial sample detection might be the most practical approach as it can be deployed in a plug-and-play mode without involving the trained models. Motivated by this observation, GAN can be used to detect adversarial samples in various ways. For example, we can configure G as a feature squeezer that transforms data point x to latent space z (i.e., $G(x) = z$) and then train the discriminator D as a binary classifier to check whether the latent vector comes from real data $G(x)$ or adversarial sample $G(x')$. Finally, an adversarial sample could be easily detected through $D(G(\cdot))$ operation.

6.3.2 Malware-Related Research. Most of the current malware-related research based on GAN focuses on Android malware because the security protection mechanism is relatively consummate in the OS like Windows, Linux, and MacOS. This indicates that in the area of those OS, there still exist many research opportunities to overcome the unsolved challenges, such as analysis combining statistic and dynamic parameters, defensive programming, white-box attack, and code reviewing.

Statistic analysis. The majority of the previous works are about statistic analysis that effectively identifies the existing malware. With the help of GAN, the issue of insufficient malware samples can be fixed. In addition, another drawback of statistic analysis on malware is its relatively weak performance of detecting unknowing malware. GAN is able to generate unknowing malware based on the known attacks but still needs experts to extract features by hand. How to detect malware against virtualization and extract statistic data more effectively make the issues worth exploring.

Dynamic analysis. Dynamic analysis is more robust than statistic analysis, but the existing dynamic analysis tools and techniques are imperfect. GAN can help enhance the performance of malware detection by applying dynamic analysis via generating more adversarial samples. However, only several dynamic features are actually utilized in the current works. As virtualization is getting widespread, dynamic analysis on VMM and hypervisor may be a promising direction. Moreover, dynamic analysis on the side channel of attack behaviors is still left blank now, which should be filled to advance malware detection techniques.

White-box attacks. Black-box attacks and semi white-box attacks are the focus of most of the existing works. Compared with black-box attacks, white-box attacks require higher transparency of the target systems. Usually, white-box attacks are related to code review, which has a high requirement for experts' experience. In addition, there is a lack of datasets for the white-box attack, which is a good application scenario for GAN.

6.3.3 Bioinformatic Recognition. Bioinformatic-based recognition has been widely used in various areas, some of which are related to the applications requiring a high degree of security protection, such as bioinformatic authentication. Slight modifications on bioinformatics may cause the results of recognition change thoroughly. Current works are limited to fingerprint and face recognition, ignoring other important bioinformatics like iris. As a result, more efforts are needed to design GAN-based methods for different bioinformatic recognition systems.

6.3.4 Industrial Security and Others. Industrial security is more complex and challenging and thus has higher requirements. Here, using GAN to generate adversarial data from limited data is helpful in enhancing industrial security. However, GAN also has its limitation: the probability distribution of adversarial data may be unbalanced when there is too little original data. Especially in a situation where the specific context of a test is missing, combining transfer learning and GAN may enable the generation of adversarial samples from limited data resources.

7 CONCLUSION

This survey intensively reviews the state-of-the-art approaches using GAN for privacy and security in a broad spectrum of applications, including image generation, video event detection, records publishing, distributed learning, malware detection, fraud detection, and so on. For the different purposes of attack and defense, these existing approaches establish problem formulation based on the variants of GAN framework, taking into account attack success rate, classification/prediction accuracy, data utility, and other performance metrics. After a thorough analysis, the unsolved challenges and promising research directions are provided for further discussion from perspectives of application scenario, model design, and data utilization.

REFERENCES

[1] Martin Abadi, Andy Chu, Ian Goodfellow, H. Brendan McMahan, Ilya Mironov, Kunal Talwar, and Li Zhang. 2016. Deep learning with differential privacy. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security.* ACM, New York, NY, 308–318.

[2] Gergely Acs, Luca Melis, Claude Castelluccia, and Emiliano De Cristofaro. 2018. Differentially private mixture of generative neural networks. *IEEE Transactions on Knowledge and Data Engineering* 31, 6 (2018), 1109–1121.

[3] Ulrich Aïvodji, Sébastien Gambs, and Timon Ther. 2019. GAMIN: An adversarial approach to black-box model inversion. arXiv:1909.11835.

[4] Mohammad Al-Rubaie and J. Morris Chang. 2019. Privacy-preserving machine learning: Threats and solutions. *IEEE Security & Privacy* 17 (2019), 49–58.

[5] Constantin F. Aliferis, Ioannis Tsamardinos, and Alexander Statnikov. 2003. HITON: A novel Markov blanket algorithm for optimal variable selection. In *Proceedings of the AMIA Annual Symposium*. 21–25.

[6] Ranya Aloufi, Hamed Haddadi, and David Boyle. 2019. Emotionless: Privacy-preserving speech analysis for voice assistants. arXiv:1908.03632.

[7] Martín Arjovsky, Soumith Chintala, and Léon Bottou. 2017. Wasserstein GAN. arXiv:1701.07875

[8] Giuseppe Ateniese, Giovanni Felici, Luigi Mancini, Angelo Spognardi, Antonio Villani, and Domenico Vitali. 2015. Hacking smart machines with smarter ones: How to extract meaningful data from machine learning classifiers. *International Journal of Security and Networks* 10 (2015), 137–150.

[9] Sean Augenstein, H. Brendan McMahan, Daniel Ramage, Swaroop Ramaswamy, Peter Kairouz, Mingqing Chen, Rajiv Mathews, and Blaise Agüera y Arcas. 2019. Generative models for effective ML on private, decentralized datasets. arXiv:1911.06679

[10] Shumeet Baluja and Ian Fischer. 2017. Adversarial transformation networks: Learning to generate adversarial examples. arXiv:1703.09387

[11] Samyadeep Basu, Rauf Izmailov, and Chris Mesterharm. 2019. Membership model inversion attacks for deep networks. arXiv:1910.04257

[12] Brett K. Beaulieu-Jones, Zhiwei Steven Wu, Chris Williams, Ran Lee, Sanjeev P. Bhavnani, James Brian Byrd, and Casey S. Greene. 2019. Privacy-preserving generative deep neural networks support clinical data sharing. *Circulation: Cardiovascular Quality and Outcomes* 12, 7 (2019), e005122.

[13] David Berthelot, Tom Schumm, and Luke Metz. 2017. BEGAN: Boundary equilibrium generative adversarial networks. arXiv:1703.10717

[14] Philip Bontrager, Julian Togelius, and Nasir D. Memon. 2017. DeepMasterPrint: Generating fingerprints for presentation attacks. arXiv:1705.07386

[15] Karla Brkić, Tomislav Hrkać, Zoran Kalafatić, and Ivan Sikirić. 2017. Face, hairstyle and clothing colour de-identification in video sequences. *IET Signal Processing* 11, 9 (2017), 1062–1068.

[16] Karla Brkic, Ivan Sikiric, Tomislav Hrkac, and Zoran Kalafatic. 2017. I know that person: Generative full body and face de-identification of people in images. In *Proceedings of the 2017 IEEE CVPR Workshops*. IEEE, Los Alamitos, CA, 1319–1328.

[17] Eoin Brophy, Zhengwei Wang, and Tomas E. Ward. 2019. Quick and easy time series generation with established image-based GANs. arXiv:1902.05624

[18] Jie Cao, Yibo Hu, Bing Yu, Ran He, and Zhenan Sun. 2019. 3D Aided Duet GANs for multi-view face image synthesis. *IEEE Transactions on Information Forensics and Security* 14, 8 (2019), 2028–2042.

[19] Kamalika Chaudhuri, Jacob Imola, and Ashwin Machanavajjhala. 2019. Capacity bounded differential privacy. In *Advances in Neural Information Processing Systems*, Vol. 32. Curran Associates, Red Hook, NY, 1–10.

[20] Jiawei Chen, Janusz Konrad, and Prakash Ishwar. 2018. VGAN-based image representation learning for privacy-preserving facial expression recognition. In *Proceedings of the IEEE CVPR Workshops*. IEEE, Los Alamitos, CA, 1570–1579.

[21] Xi Chen, Yan Duan, Rein Houthooft, John Schulman, Ilya Sutskever, and Pieter Abbeel. 2016. InfoGAN: Interpretable representation learning by information maximizing generative adversarial nets. In *Proceedings of the 30th Conference on Neural Information Processing Systems*. 2172–2180.

[22] Xiao Chen, Peter Kairouz, and Ram Rajagopal. 2018. Understanding compressive adversarial privacy. In *Proceedings of the 2018 IEEE Conference on Decision and Control (CDC'18)*. IEEE, Los Alamitos, CA, 6824–6831.

[23] Edward Choi, Siddharth Biswal, Bradley Malin, Jon Duke, Walter F. Stewart, and Jimeng Sun. 2017. Generating multi-label discrete patient records using generative adversarial networks. In *Proceedings of the Machine Learning for Healthcare Conference*. 286–305.

[24] Zihang Dai, Zhilin Yang, Fan Yang, William W. Cohen, and Russ R. Salakhutdinov. 2017. Good semi-supervised learning that requires a bad GAN. In *Proceedings of the 31st Conference on Neural Information Processing Systems*. 6510–6520.

[25] Naser Damer, Alexandra Mosegui Saladie, Andreas Braun, and Arjan Kuijper. 2018. MorGAN: Recognition vulnerability and attack detectability of face morphing attacks created by generative adversarial network. In *Proceedings of the 9th IEEE International Conference on Biometrics Theory, Applications, and Systems*. IEEE, Los Alamitos, CA, 1–10.

[26] Debayan Deb, Jianbang Zhang, and Anil K. Jain. 2019. AdvFaces: Adversarial face synthesis. arXiv:1908.05008

[27] Xiaofeng Ding, Hongbiao Fang, Zhilin Zhang, Kim-Kwang Raymond Choo, and Hai Jin. 2020. Privacy-preserving feature extraction via adversarial training. *IEEE Transactions on Knowledge and Data Engineering* 1 (2020), 1–10.

[28] Chris Donahue, Julian J. McAuley, and Miller S. Puckette. 2018. Synthesizing audio with generative adversarial networks. arXiv:1802.04208

[29] Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. 2006. Calibrating noise to sensitivity in private data analysis. In *Proceedings of the Theory of Cryptography Conference*. 265–284.

[30] Cristóbal Esteban, Stephanie L. Hyland, and Gunnar Rätsch. 2017. Real-valued (medical) time series generation with recurrent conditional GANs. arXiv:1706.02633

[31] Junbin Fang, Aiping Li, and Qianyue Jiang. 2019. GDAGAN: An anonymization method for graph data publishing using generative adversarial network. In *Proceedings of the 2019 6th International Conference on Information Science and Control Engineering*. IEEE, Los Alamitos, CA, 309–313.

[32] William Fedus, Ian J. Goodfellow, and Andrew M. Dai. 2018. MaskGAN: Better text generation via filling in the ______. arXiv:1801.07736

[33] Maryam Feily, Alireza Shahrestani, and Sureswaran Ramadass. 2009. A survey of botnet and botnet detection. In *Proceedings of the 2009 3rd International Conference on Emerging Security Information, Systems, and Technologies*. IEEE, Los Alamitos, CA, 268–273.

[34] Jianjiang Feng and Anil K. Jain. 2010. Fingerprint reconstruction: From minutiae to phase. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 33 (2010), 209–223.

[35] Matt Fredrikson, Somesh Jha, and Thomas Ristenpart. 2015. Model inversion attacks that exploit confidence information and basic countermeasures. In *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*. ACM, New York, NY, 1322–1333.

[36] Matthew Fredrikson, Eric Lantz, Somesh Jha, Simon Lin, David Page, and Thomas Ristenpart. 2014. Privacy in pharmacogenetics: An end-to-end case study of personalized warfarin dosing. In *Proceedings of the 23rd USENIX Security Symposium*. 17–32.

[37] Brendan J. Frey, Geoffrey E. Hinton, and Peter Dayan. 1996. Does the wake-sleep algorithm produce good density estimators? In *Advances in Neural Information Processing Systems*. MIT Press, Cambridge, MA, 661–667.

[38] Yutong Gao and Yi Pan. 2020. Improved detection of adversarial images using deep neural networks. arXiv:2007.05573

[39] Felix A. Gers, Jürgen Schmidhuber, and Fred A. Cummins. 2000. Learning to forget: Continual prediction with LSTM. *Neural Computation* 12, 10 (2000), 2451–2471.

[40] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative adversarial nets. In *Advances in Neural Information Processing Systems*. Curran Associates, Red Hook, NY, 2672–2680.

[41] Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. 2015. Explaining and harnessing adversarial examples. arXiv:1412.6572

[42] Kay Gregor Hartmann, Robin Tibor Schirrmeister, and Tonio Ball. 2018. EEG-GAN: generative adversarial networks for electroencephalograhic (EEG) brain signals. arXiv:1806.01875

[43] Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron C. Courville. 2017. Improved training of Wasserstein GANs. In *Advances in Neural Information Processing Systems*. 5767–5777.

[44] Qilong Han, Zuobin Xiong, and Kejia Zhang. 2018. Research on trajectory data releasing method via differential privacy based on spatial partition. *Security and Communication Networks* 2018 (2018), Article 4248092.

[45] Corentin Hardy, Erwan Le Merrer, and Bruno Sericola. 2019. MD-GAN: Multi-discriminator generative adversarial networks for distributed datasets. In *Proceedings of the 2019 IEEE International Parallel and Distributed Processing Symposium*. IEEE, Los Alamitos, CA, 866–877.

[46] Jamie Hayes, Luca Melis, George Danezis, and Emiliano De Cristofaro. 2019. LOGAN: Membership inference attacks against generative models. *Proceedings on Privacy Enhancing Technologies* 2019 (2019), 133–152.

[47] Briland Hitaj, Giuseppe Ateniese, and Fernando Pérez-Cruz. 2017. Deep models under the GAN: Information leakage from collaborative deep learning. In *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*. ACM, New York, NY, 603–618.

[48] Yongjun Hong, Uiwon Hwang, Jaeyoon Yoo, and Sungroh Yoon. 2019. How generative adversarial networks and their variants work: An overview. *ACM Computing Surveys* 52, 1 (2019), 10.

[49] Weiwei Hu and Ying Tan. 2017. Generating adversarial malware examples for black-box attacks based on GAN. arXiv:1702.05983

[50] Zhicheng Hu, Jianqi Shi, YanHong Huang, Jiawen Xiong, and Xiangxing Bu. 2018. GANFuzz: A GAN-based industrial network protocol fuzzing framework. In *Proceedings of the 15th ACM International Conference on Computing Frontiers*. ACM, New York, NY, 138–145.

[51] Chong Huang, Peter Kairouz, Xiao Chen, Lalitha Sankar, and Ram Rajagopal. 2017. Context-aware generative adversarial privacy. *Entropy* 19, 12 (2017), 656.

[52] Mohd Ibrahim and Rodina Ahmad. 2010. Class diagram extraction from textual requirements using natural language processing (NLP) techniques. In *Proceedings of the International Conference on Computer Research and Development*. IEEE, Los Alamitos, CA, 200–204.

[53] Nikolay Jetchev, Urs Bergmann, and Roland Vollgraf. 2016. Texture synthesis with spatial generative adversarial networks. arXiv:1611.08207

[54] Liangxiao Jiang, Harry Zhang, and Zhihua Cai. 2008. A novel Bayes model: Hidden naive bayes. *IEEE Transactions on Knowledge and Data Engineering* 21 (2008), 1361–1371.

[55] Joakim Kargaard, Tom Drange, Ah-Lian Kor, Hissam Twafik, and Emlyn Butterfield. 2018. Defending IT systems against intelligent malware. In *Proceedings of the 2018 IEEE 9th International Conference on Dependable Systems, Services, and Technologies*. IEEE, Los Alamitos, CA, 411–417.

[56] Animesh Karnewar and Oliver Wang. 2020. MSG-GAN: Multi-scale gradients for generative adversarial networks. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition*. IEEE, Los Alamitos, CA, 7799–7808.

[57] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. 2017. Progressive growing of GANs for improved quality, stability, and variation. arXiv:1710.10196

[58] Bach Ngoc Kim, Christian Desrosiers, Jose Dolz, and Pierre-Marc Jodoin. 2019. Privacy-Net: An adversarial approach for identity-obfuscated segmentation. arXiv:1909.04087

[59] Hakil Kim, Xuenan Cui, Man-Gyu Kim, and Thi Hai Binh Nguyen. 2019. Fingerprint generation and presentation attack detection using deep neural networks. In *Proceedings of the 2019 IEEE Conference on Multimedia Information Processing and Retrieval*. IEEE, Los Alamitos, CA, 375–378.

[60] Jin-Young Kim, Seok-Jun Bu, and Sung-Bae Cho. 2017. Malware detection using deep transferred generative adversarial networks. In *Proceedings of the International Conference on Neural Information Processing Systems*. 556–564.

[61] Jin-Young Kim, Seok-Jun Bu, and Sung-Bae Cho. 2018. Zero-day malware detection using transferred generative adversarial networks based on deep autoencoders. *Information Sciences* 460 (2018), 83–102.

[62] Taehoon Kim and Jihoon Yang. 2019. Latent-space-level image anonymization with adversarial protector networks. *IEEE Access* 7 (2019), 84992–84999.

[63] Jernej Kos, Ian Fischer, and Dawn Song. 2018. Adversarial examples for generative models. In *Proceedings of the 2018 IEEE Security and Privacy Workshops*. IEEE, Los Alamitos, CA, 36–42.

[64] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. 2012. ImageNet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems*. Curran Associates, Red Hook, NY, 1097–1105.

[65] Martha Larson, Zhuoran Liu, S. F. B. Brugman, and Zhengyu Zhao. 2018. Pixel privacy: Increasing image appeal while blocking automatic inference of sensitive scene information. In *Working Notes Proceedings of the MediaEval 2018 Workshop*, Vol. 2283.

[66] Dongha Lee, Hwanjo Yu, Xiaoqian Jiang, Deevakar Rogith, Meghana Gudala, Mubeen Tejani, Qiuchen Zhang, and Li Xiong. 2020. Generating sequential electronic health records using dual adversarial autoencoder. *Journal of the American Medical Informatics Association* 27, 9 (2020), 1411–1419.

[67] Hyeungill Lee, Sungyeob Han, and Jungwoo Lee. 2017. Generative adversarial trainer: Defense to adversarial perturbations with GAN. arXiv:1705.03387

[68] Harim Lee, Myeung Un Kim, Yeong-Jun Kim, Hyeonsu Lyu, and Hyun Jong Yang. 2020. Privacy-protection drone patrol system based on face anonymization. arXiv:2005.14390

[69] Aiping Li, Junbin Fang, Qianye Jiang, Bin Zhou, and Yan Jia. 2020. A graph data privacy-preserving method based on generative adversarial networks. In *Proceedings of the International Conference on Web Information Systems Engineering*. 227–239.

[70] Jia Li, Honglei Zhang, Zhichao Han, Yu Rong, Hong Cheng, and Junzhou Huang. 2020. Adversarial attack on community detection by hiding individuals. In *Proceedings of the Web Conference 2020*. ACM, New York, NY, 917–927.

[71] Kaiyang Li, Guoming Lu, Guangchun Luo, and Zhipeng Cai. 2020. Seed-free graph de-anonymization with adversarial learning. In *Proceedings of the 29th ACM International Conference on Information and Knowledge Management*. ACM, New York, NY, 745–754.

[72] Kaiyang Li, Guangchun Luo, Yang Ye, Wei Li, Shihao Ji, and Zhipeng Cai. 2020. Adversarial privacy preserving graph embedding against inference attack. arXiv:2008.13072

[73] Qinya Li, Zhenzhe Zheng, Fan Wu, and Guihai Chen. 2020. Generative adversarial networks-based privacy-preserving 3D reconstruction. In *Proceedings of the 2020 IEEE/ACM 28th International Symposium on Quality of Service*. IEEE, Los Alamitos, CA, 1–10.

[74] Yitong Li, Timothy Baldwin, and Trevor Cohn. 2018. Towards robust and privacy-preserving text representations. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*. 25–30.

[75] Guanxiong Liu, Issa Khalil, and Abdallah Khreishah. 2019. GanDef: A GAN based adversarial training defense for neural network classifier. In *Proceedings of the International Conference on ICT Systems Security and Privacy Protection*. 19–32.

[76] Kin Sum Liu, Bo Li, and Jiexin Gao. 2019. Performing co-membership attacks against deep generative models. In *Proceedings of the 2019 IEEE International Conference on Data Mining*. IEEE, Los Alamitos, CA, 459–467.

[77] Sicong Liu, Anshumali Shrivastava, Junzhao Du, and Lin Zhong. 2019. Better accuracy with quantified privacy: Representations learned via reconstructive adversarial network. arXiv:1901.08730

[78] Yi Liu, Jialiang Peng, J. Q. James, and Yi Wu. 2019. PPGAN: Privacy-preserving generative adversarial network. In *Proceedings of the 2019 IEEE 25th International Conference on Parallel and Distributed Systems (ICPADS'19)*. IEEE, Los Alamitos, CA, 985–989.

[79] William Lotter, Gabriel Kreiman, and David D. Cox. 2015. Unsupervised learning of visual structure using predictive generative networks. arXiv:1511.06380

[80] Christos Louizos, Kevin Swersky, Yujia Li, Max Welling, and Richard S. Zemel. 2016. The variational fair autoencoder. In *Proceedings of the 4th International Conference on Learning Representations*. 31–40.

[81] Pei-Hsuan Lu, Pang-Chieh Wang, and Chia-Mu Yu. 2019. Empirical evaluation on synthetic data generation with generative adversarial network. In *Proceedings of the 9th International Conference on Web Intelligence, Mining, and Semantics*. ACM, New York, NY, 1–6.

[82] Pauline Luc, Camille Couprie, Soumith Chintala, and Jakob Verbeek. 2016. Semantic segmentation using adversarial networks. arXiv:1611.08408

[83] Liqian Ma, Xu Jia, Qianru Sun, Bernt Schiele, Tinne Tuytelaars, and Luc Van Gool. 2017. Pose guided person image generation. In *Proceedings of the 31st Conference on Neural Information Processing Systems*. 406–416.

[84] Xudong Mao, Qing Li, Haoran Xie, Raymond Y. K. Lau, Zhen Wang, and Stephen Paul Smolley. 2017. Least squares generative adversarial networks. In *Proceedings of the IEEE International Conference on Computer Vision*. 2794–2802.

[85] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. 2017. Communication-efficient learning of deep networks from decentralized data. In *Artificial Intelligence and Statistics*. PMLR, 1273–1282.

[86] H. Brendan McMahan, Daniel Ramage, Kunal Talwar, and Li Zhang. 2017. Learning differentially private language models without losing accuracy. arXiv:1710.06963

[87] Ruohan Meng, Qi Cui, Zhili Zhou, Zhangjie Fu, and Xingming Sun. 2019. A steganography algorithm based on CycleGAN for covert communication in the Internet of Things. *IEEE Access* 7 (2019), 90574–90584.

[88] Vahid Mirjalili, Sebastian Raschka, Anoop Namboodiri, and Arun Ross. 2018. Semi-adversarial networks: Convolutional autoencoders for imparting privacy to face images. In *Proceedings of the 2018 International Conference on Biometrics*. IEEE, Los Alamitos, CA, 82–89.

[89] Vahid Mirjalili, Sebastian Raschka, and Arun Ross. 2020. PrivacyNet: Semi-adversarial networks for multi-attribute face privacy. *IEEE Transactions on Image Process* 29 (2020), 9400–9412.

[90] Mehdi Mirza and Simon Osindero. 2014. Conditional generative adversarial nets. arXiv:1411.1784

[91] Takeru Miyato and Masanori Koyama. 2018. cGANs with projection discriminator. arXiv:1802.05637

[92] Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, and Pascal Frossard. 2016. DeepFool: A simple and accurate method to fool deep neural networks. In *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, Los Alamitos, CA, 2574–2582.

[93] Milad Nasr, Reza Shokri, and Amir Houmansadr. 2018. Machine learning with membership privacy using adversarial regularization. In *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*. ACM, New York, NY, 634–646.

[94] Hung Nguyen, Di Zhuang, Pei-Yuan Wu, and Morris Chang. 2020. AutoGAN-based dimension reduction for privacy preservation. *Neurocomputing* 384 (2020), 94–103.

[95] Augustus Odena, Christopher Olah, and Jonathon Shlens. 2017. Conditional image synthesis with auxiliary classifier GANs. In *Proceedings of the 34th International Conference on Machine Learning*. 2642–2651.

[96] Nicolas Papernot, Patrick McDaniel, Ian Goodfellow, Somesh Jha, Z. Berkay Celik, and Ananthram Swami. 2017. Practical black-box attacks against machine learning. In *Proceedings of the 2017 ACM on Asia Conference on Computer and Communications Security*. ACM, New York, NY, 506–519.

[97] Noseong Park, Mahmoud Mohammadi, Kshitij Gorde, Sushil Jajodia, Hongkyu Park, and Youngmin Kim. 2018. Data synthesis based on generative adversarial networks. *Proceedings of the VLDB Endowment* 11, 10 (2018), 1071–1083.

[98] Francesco Pittaluga, Sanjeev Koppal, and Ayan Chakrabarti. 2019. Learning privacy preserving encodings through adversarial training. In *Proceedings of the 2019 IEEE Winter Conference on Applications of Computer Vision*. IEEE, Los Alamitos, CA, 791–799.

[99] Zhaofan Qiu, Yingwei Pan, Ting Yao, and Tao Mei. 2017. Deep semantic hashing with generative adversarial networks. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM, New York, NY, 225–234.

[100] Lawrence R. Rabiner. 1989. A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE* 77 (1989), 257–286.

[101] Jinmeng Rao, Song Gao, Yuhao Kang, and Qunying Huang. 2021. LSTM-TrajGAN: A deep learning approach to trajectory privacy protection. In *Proceedings of the 11th International Conference on Geographic Information Science (GIScience'21)*, Vol. 177. Article 12, 17 pages.

[102] Carl Edward Rasmussen. 1999. The infinite Gaussian mixture model. In *Advances in Neural Information Processing Systems*. MIT Press, Cambridge, MA, 554–560.

[103] Mohammad Rasouli, Tao Sun, and Ram Rajagopal. 2020. FedGAN: Federated generative adversarial networks for distributed data. arXiv:2006.07228

[104] Zhongzheng Ren, Yong Jae Lee, and Michael S. Ryoo. 2018. Learning to anonymize faces for privacy preserving action detection. In *Proceedings of the European Conference on Computer Vision*. 620–636.

[105] Aria Rezaei, Chaowei Xiao, Jie Gao, and Bo Li. 2018. Protecting sensitive attributes via generative adversarial networks. arXiv:1812.10193

[106] Mauro Ribeiro, Katarina Grolinger, and Miriam A. M. Capretz. 2015. MLaaS: Machine Learning as a Service. In *Proceedings of the 2015 IEEE 14th International Conference on Machine Learning and Applications*. IEEE, Los Alamitos, CA, 896–902.

[107] Michael S. Ryoo, Brandon Rothrock, Charles Fleming, and Hyun Jong Yang. 2017. Privacy-preserving human activity recognition from extreme low resolution. In *Proceedings of the 31st Conference on Artificial Intelligence*. 4255–4262.

[108] Pouya Samangouei, Maya Kabkab, and Rama Chellappa. 2018. Defense-GAN: Protecting classifiers against adversarial attacks using generative models. arXiv:1805.06605

[109] Eunbi Seo, Hyun Min Song, and Huy Kang Kim. 2018. GIDS: GAN based intrusion detection system for in-vehicle network. In *Proceedings of the 2018 16th Annual Conference on Privacy, Security and Trust*. IEEE, Los Alamitos, CA, 1–6.

[110] Maryam Shahpasand, Len Hamey, Dinusha Vatsalan, and Minhui Xue. 2019. Adversarial attacks on mobile malware detection. In *Proceedings of the 2019 IEEE 1st International Workshop on Artificial Intelligence for Mobile*. IEEE, Los Alamitos, CA, 17–20.

[111] Shiwei Shen, Guoqing Jin, Ke Gao, and Yongdong Zhang. 2017. APE-GAN: Adversarial perturbation elimination with GAN. arXiv:1707.05474

[112] Rakshith Shetty, Bernt Schiele, and Mario Fritz. 2018. A4NT: Author attribute anonymity by adversarial training of neural machine translation. In *Proceedings of the 27th USENIX Security Symposium (USENIX'18)*. 1633–1650.

[113] Reza Shokri and Vitaly Shmatikov. 2015. Privacy-preserving deep learning. In *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*. ACM, New York, NY, 1310–1321.

[114] Reza Shokri, Marco Stronati, Congzheng Song, and Vitaly Shmatikov. 2017. Membership inference attacks against machine learning models. In *Proceedings of the 2017 IEEE Symposium on Security and Privacy*. IEEE, Los Alamitos, CA, 3–18.

[115] Dule Shu, Weilin Cong, Jiaming Chai, and Conrad S. Tucker. 2020. Encrypted rich-data steganography using generative adversarial networks. In *Proceedings of the 2nd ACM Workshop on Wireless Security and Machine Learning*. ACM, New York, NY, 55–60.

[116] Congzheng Song, Thomas Ristenpart, and Vitaly Shmatikov. 2017. Machine learning models that remember too much. In *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*. ACM, New York, NY, 587–601.

[117] Yang Song, Rui Shu, Nate Kushman, and Stefano Ermon. 2018. Constructing unrestricted adversarial examples with generative models. In *Advances in Neural Information Processing Systems*. Curran Associates, Red Hook, NY, 8312–8323.

[118] Nasim Souly, Concetto Spampinato, and Mubarak Shah. 2017. Semi supervised semantic segmentation using generative adversarial network. In *Proceedings of the IEEE International Conference on Computer Vision*. IEEE, Los Alamitos, CA, 5688–5696.

[119] Patricia L. Suárez, Angel D. Sappa, and Boris X. Vintimilla. 2017. Infrared image colorization based on a triplet DCGAN architecture. In *Proceedings of the IEEE CVPR Workshops*. IEEE, Los Alamitos, CA, 18–23.

[120] Martin Sundermeyer, Ralf Schlüter, and Hermann Ney. 2012. LSTM neural networks for language modeling. In *Proceedings of the 13th Annual Conference of the International Speech Communication Association*. 194–197.

[121] R. Taheri, M. Shojafar, M. Alazab, and R. Tafazolli. 2020. FED-IIoT: A robust federated malware detection architecture in Industrial IoT. *IEEE Transactions on Industrial Informatics*. Early access. December 9, 2020.

[122] Weixuan Tang, Shunquan Tan, Bin Li, and Jiwu Huang. 2017. Automatic steganographic distortion learning using a generative adversarial network. *IEEE Signal Processing Letters* 24, 10 (2017), 1547–1551.

[123] Amirsina Torfi and Edward A. Fox. 2020. CorGAN: Correlation-capturing convolutional generative adversarial networks for generating synthetic healthcare records. In *Proceedings of the 33th International Florida Artificial Intelligence Research Society Conference*, Roman Barták and Eric Bell (Eds.). AAAI Press, 335–340.

[124] Aleksei Triastcyn and Boi Faltings. 2018. Generating differentially private datasets using GANs. arXiv:1803.03148

[125] Aleksei Triastcyn and Boi Faltings. 2019. Federated learning with Bayesian differential privacy. In *Proceedings of the 2019 IEEE International Conference on Big Data*. IEEE, Los Alamitos, CA, 2587–2596.

[126] Aleksei Triastcyn and Boi Faltings. 2020. Federated generative privacy. *IEEE Intelligent Systems* 35, 4 (2020), 50–57.

[127] Ardhendu Tripathy, Ye Wang, and Prakash Ishwar. 2019. Privacy-preserving adversarial networks. In *Proceedings of the 2019 57th Annual Allerton Conference on Communication, Control, and Computing*. IEEE, Los Alamitos, CA, 495–505.

[128] Bo-Wei Tseng and Pei-Yuan Wu. 2020. Compressive privacy generative adversarial network. *IEEE Transactions on Information Forensics and Security* 15 (2020), 2499–2513.

[129] Ries Uittenbogaard, Clint Sebastian, Julien Vijverberg, Bas Boom, Dariu M. Gavrila, and Peter H. N. de With. 2019. Privacy protection in street-view panoramas using depth and multi-view imagery. In *Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition*. IEEE, Los Alamitos, CA, 10581–10590.

[130] Korosh Vatanparvar, Viswam Nathan, Ebrahim Nemati, Md Mahbubur Rahman, and Jilong Kuang. 2020. Adapting to noise in speech obfuscation by audio profiling using generative models for passive health monitoring. In *Proceedings of the 2020 42nd Annual International Conference of the IEEE Engineering in Medicine & Biology Society*. IEEE, Los Alamitos, CA, 5700–5704.

[131] Carl Vondrick, Hamed Pirsiavash, and Antonio Torralba. 2016. Generating videos with scene dynamics. In *Proceedings of the 30th Conference on Neural Information Processing Systems*. 613–621.

[132] Xiaosen Wang, Kun He, and John E. Hopcroft. 2019. AT-GAN: A Generative attack model for adversarial transferring on generative adversarial nets. arXiv:1904.07793

[133] Zhibo Wang, Song Mengkai, Zhifei Zhang, Yang Song, Qian Wang, and Hairong Qi. 2019. Beyond inferring class representatives: User-level privacy leakage from federated learning. In *Proceedings of the IEEE Conference on Computer Communications*. IEEE, Los Alamitos, CA, 2512–2520.

[134] Zihao W. Wang, Vibhav Vineet, Francesco Pittaluga, Sudipta N. Sinha, Oliver Cossairt, and Sing Bing Kang. 2019. Privacy-preserving action recognition using coded aperture videos. In *Proceedings of the IEEE CVPR Workshops*. IEEE, Los Alamitos, CA, 1–10.

[135] Bingzhe Wu, Shiwan Zhao, Chaochao Chen, Haoyang Xu, Li Wang, Xiaolu Zhang, Guangyu Sun, and Jun Zhou. 2019. Generalization in generative adversarial networks: A novel perspective from privacy protection. In *Proceedings of the 33rd Conference on Neural Information Processing Systems*. 307–317.

[136] Yifan Wu, Fan Yang, Yong Xu, and Haibin Ling. 2019. Privacy-Protective-GAN for privacy preserving face de-identification. *Journal of Computer Science and Technology* 34, 1 (2019), 47–60.

[137] Chaowei Xiao, Bo Li, Jun-Yan Zhu, Warren He, Mingyan Liu, and Dawn Song. 2018. Generating adversarial examples with adversarial networks. arXiv:1801.02610

[138] Cihang Xie and Alan L. Yuille. 2019. Intriguing properties of adversarial training. arXiv:1906.03787

[139] Liyang Xie, Kaixiang Lin, Shu Wang, Fei Wang, and Jiayu Zhou. 2018. Differentially private generative adversarial network. arXiv:1802.06739

[140] Zuobin Xiong, Zhipeng Cai, Qilong Han, Arwa Alrawais, and Wei Li. 2020. ADGAN: Protect your location privacy in camera data of auto-driving vehicles. *IEEE Transactions on Industrial Informatics*. Early access. October 20, 2020.

[141] Zuobin Xiong, Wei Li, Qilong Han, and Zhipeng Cai. 2019. Privacy-preserving auto-driving: A GAN-based approach to protect vehicular camera data. In *Proceedings of the 2019 IEEE International Conference on Data Mining*. IEEE, Los Alamitos, CA, 668–677.

[142] Chugui Xu, Ju Ren, Deyu Zhang, Yaoxue Zhang, Zhan Qin, and Kui Ren. 2019. GANobfuscator: Mitigating information leakage under GAN via differential privacy. *IEEE Transactions on Information Forensics and Security* 14, 9 (2019), 2358–2371.

[143] Andrew Yale, Saloni Dash, Ritik Dutta, Isabelle Guyon, Adrien Pavao, and Kristin P. Bennett. 2020. Generation and evaluation of privacy preserving synthetic health data. *Neurocomputing* 416 (2020), 244–255.

[144] X. Yan, B. Cui, Y. Xu, P. Shi, and Z. Wang. 2019. A method of information protection for collaborative deep learning under GAN model attack. *IEEE/ACM Transactions on Computational Biology and Bioinformatics* 1 (2019), 1.

[145] Chao Yang, Xin Lu, Zhe Lin, Eli Shechtman, Oliver Wang, and Hao Li. 2017. High-resolution image inpainting using multi-scale neural patch synthesis. In *Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition*. 6721–6729.

[146] Jin Yang, Tao Li, Gang Liang, Wenbo He, and Yue Zhao. 2019. A simple recurrent unit model based intrusion detection system with DCGAN. *IEEE Access* 7 (2019), 83286–83296.

[147] Tsung-Yen Yang, Christopher Brinton, Prateek Mittal, Mung Chiang, and Andrew Lan. 2018. Learning Informative and private representations via generative adversarial networks. In *Proceedings of the 2018 IEEE International Conference on Big Data*. IEEE, Los Alamitos, CA, 1534–1543.

[148] Xiao Yang, Yinpeng Dong, Tianyu Pang, Jun Zhu, and Hang Su. 2020. Towards privacy protection by generating adversarial identity masks. arXiv:2003.06814

[149] Zhilin Yang, Junjie Hu, Ruslan Salakhutdinov, and William W. Cohen. 2017. Semi-supervised QA with generative domain-adaptive nets. arXiv:1702.02206

[150] Samuel Yeom, Irene Giacomelli, Matt Fredrikson, and Somesh Jha. 2018. Privacy risk in machine learning: Analyzing the connection to overfitting. In *Proceedings of the 2018 IEEE 31st Computer Security Foundations Symposium*. IEEE, Los Alamitos, CA, 268–282.

[151] Chuanlong Yin, Yuefei Zhu, Shengli Liu, Jinlong Fei, and Hetong Zhang. 2018. An enhancing framework for botnet detection using generative adversarial networks. In *Proceedings of the 2018 International Conference on Artificial Intelligence and Big Data*. IEEE, Los Alamitos, CA, 228–234.

[152] Dan Yin and Qing Yang. 2018. GANs based density distribution privacy-preservation on mobility data. *Security and Communication Networks* 2018 (2018), Article 9203076.

[153] Ryo Yonetani, Tomohiro Takahashi, Atsushi Hashimoto, and Yoshitaka Ushiku. 2019. Decentralized learning of generative adversarial networks from multi-client non-iid data. arXiv:1905.09684

[154] Jinsung Yoon, James Jordon, and Mihaela van der Schaar. 2019. PATE-GAN: Generating synthetic data with differential privacy guarantees. In *Proceedings of the International Conference on Learning Representations*. IEEE, Los Alamitos, CA, 536–545.

[155] Han Zhang, Ian Goodfellow, Dimitris Metaxas, and Augustus Odena. 2019. Self-Attention generative adversarial networks. In *Proceedings of the International Conference on Machine Learning*. 7354–7363.

[156] Xinyang Zhang, Shouling Ji, and Ting Wang. 2018. Differentially private releasing via deep generative model. arXiv:1801.01594

[157] Yuheng Zhang, Ruoxi Jia, Hengzhi Pei, Wenxiao Wang, Bo Li, and Dawn Song. 2020. The secret revealer: Generative model-inversion attacks against deep neural networks. In *Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition*. IEEE, Los Alamitos, CA, 253–261.

[158] Zhengli Zhao, Dheeru Dua, and Sameer Singh. 2017. Generating natural adversarial examples. arXiv:1710.11342

[159] Yu-Jun Zheng, Xiao-Han Zhou, Wei-Guo Sheng, Yu Xue, and Sheng-Yong Chen. 2018. Generative adversarial network based telecom fraud detection at the receiving bank. *Neural Networks* 102 (2018), 78–86.

[160] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A. Efros. 2017. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV'17)*. IEEE, Los Alamitos, CA, 2223–2232.

[161] Wentao Zhu and Xiaohui Xie. 2016. Adversarial deep structural networks for mammographic mass segmentation. arXiv:1612.05970

[162] Zheng-An Zhu, Yun-Zhong Lu, and Chen-Kuo Chiang. 2019. Generating adversarial examples by makeup attacks on face recognition. In *Proceedings of the 2019 IEEE International Conference on Image Processing*. IEEE, Los Alamitos, CA, 2516–2520.

Received December 2020; revised March 2020; accepted April 2021