Artificial Intelligence Review (2025) 58:33
https://doi.org/10.1007/s10462-024-11024-6

Generative AI model privacy: a survey

Yihao Liu[1] · Jinhe Huang[1] · Yanjie Li[1] · Dong Wang[1] · Bin Xiao[1]

Accepted: 6 November 2024

Abstract
The rapid progress of generative AI models has yielded substantial breakthroughs in AI, facilitating the generation of realistic synthetic data across various modalities. However, these advancements also introduce significant privacy risks, as the models may inadvertently expose sensitive information from their training data. Currently, there is no comprehensive survey work investigating privacy issues, e.g., attacking and defending privacy in generative AI models. We strive to identify existing attack techniques and mitigation strategies and to offer a summary of the current research landscape. Our survey encompasses a wide array of generative AI models, including language models, Generative Adversarial Networks, diffusion models, and their multi-modal counterparts. It indicates the critical need for continued research and development in privacy-preserving techniques for generative AI models. Furthermore, we offer insights into the challenges and discuss the open problems in the intersection of privacy and generative AI models.

Keywords Generative AI models · Privacy issues · Language models · Generative adversarial networks · Attack and defense

✉ Bin Xiao
b.xiao@polyu.edu.hk

Yihao Liu
yihao5.liu@connect.polyu.hk

Jinhe Huang
jinhe.huang@connect.polyu.hk

Yanjie Li
yanjie.li@connect.polyu.hk

Dong Wang
dong-comp.wang@connect.polyu.hk

[1] The Hong Kong Polytechnic University, Hung Hom, Hong Kong

Published online: 04 December 2024

1 Introduction

In an era where data is akin to currency, safeguarding the privacy of personal information has become an overriding priority. The widespread adoption of digital technology has resulted in a surge of personal data, a significant portion of which is collected and processed by artificial intelligence models, particularly generative AI models. These models are distinguished by their capacity to produce synthetic data that closely mimics real-world data. This encompasses language models that can produce human-like text and vision models that can create lifelike images. As these models find their way into a variety of applications, from generating content to analyzing data, they raise a fundamental challenge: privacy.

This survey embarks on methodical research of the privacy landscape within generative AI models, including language models, Generative Adversarial Networks, diffusion models, and multi-modal models. We start by offering an introduction to generative AI models and discussing their architectures and training methodologies. We then narrow our focus to the privacy these generative AI models raise. Membership privacy is a key concern, where the goal is to prevent the inference of whether or not a specific individual's data was utilized for training a model. We explore the techniques used to launch membership inference attacks and the defenses that have been developed to counteract them. Next, we discuss model inversion attacks, which aim to use the model itself to rebuild or deduce important training data properties. Another significant portion of this survey is dedicated to the privacy considerations in distributed learning systems. With the rise of collaborative AI systems, where models are trained across decentralized data sources, the threat landscape has evolved. We discuss the innovative attacks that exploit vulnerabilities in these systems, such as leveraging gradients and model updates to infer sensitive information. Lastly, we review differential privacy (DP), which ensures privacy by injecting noise into data or model outputs. We consider the use of DP in generative AI models and how well it preserves privacy without sacrificing usefulness.

The objective of this paper is to organize and synthesize the most recent advancements in privacy attacks and mitigation strategies against generative AI models. This will help researchers create stronger privacy attacks to evaluate model robustness, develop more resilient generative AI models, and ensure privacy in real-world applications. To identify relevant literature for the review, we initially define the scope of the review and conduct a meticulous search of papers published in related top conferences and journals over the past few years. We then choose highly cited or notable works that pertain to privacy attacks and defenses on generative AI models.[1]

1.1 Related work

In the Machine Learning (ML) domain, with the continuous advancements of algorithms and models, especially in handling and analyzing large amounts of personal data, privacy protection issues are increasingly prominent. This is why many works (Liu et al. 2021; Rigaki and Garcia 2023; Cristofaro 2020) have sought to summarize the most recent pertinent research in ML privacy. Among them, Hu et al. (2022b) specifically explore the impact of membership inference attacks in ML. Deep learning, a subfield of ML, has made notable

[1] GPT-4 was utilized to polish and translate during the initial drafting phase of this manuscript to improve language expression. The authors have conducted a comprehensive review of the AI-edited content.

advances in domains like Natural Language Processing (NLP) through its powerful data processing capabilities. However, these deep learning models have also raised serious concerns about privacy breaches when handling sensitive textual data. To better comprehend this, some work (Mireshghallah et al. 2020; Bae et al. 2018; Liu et al. 2020) investigate privacy and security issues in deep learning from different perspectives. Moreover, Boulemtafes et al. (2020) specialize in collecting various privacy-preserving techniques for deep learning. Recently, Golda et al. (2024) provided a comprehensive introduction to many privacy protection algorithms from the perspective of optimization algorithms, but they lack explanation from the model perspective.

For generative AI models, some work (Brown et al. 2022; Sousa and Kern 2023; Huang et al. 2024) conducted a survey on the privacy issues and countermeasures associated with language models. On the other hand, Cai et al. (2021); Zhang et al. (2022a) perform an investigation into the privacy challenges and defensive strategies associated with Generative Adversarial Networks.

In contrast to the related surveys above, this work encompasses a wider scope of topics to review privacy attacks and defenses surrounding generative AI models. This is demonstrated in Table.1, which compares this review against other studies in the literature addressing privacy issues related to generative AI models. For instance, a few studies have concentrated solely on the privacy concerns surrounding specific models, such as GANs (Hu et al. 2022b) or language models (Brown et al. 2022; Sousa and Kern 2023; Huang et al. 2024). However, our work broadens this scope to encompass additional generative AI

Table 1 A comparative analysis of this review with pertinent surveys and their respective content scopes

Work	Year	GAIM		Detailed privacy attack	Detailed privacy defense
		NLP	CV		
Bae et al. (2018)	2018		Specific		General
Cristofaro (2020)	2020		Specific		
Mireshghallah et al. (2020)	2020		Specific	Specific	Specific
Liu et al. (2020)	2020		Specific		
Cai et al. (2021)	2021		Specific	General	General
Liu et al. (2021)	2021		Specific		
Brown et al. (2022)	2022	General			General
Zhang et al. (2022a)	2022		Specific		
Hu et al. (2022b)	2022		General	Specific	General
Sousa and Kern (2023)	2023	General			General
Rigaki and Garcia (2023)	2023		Specific	General	General
Golda et al. (2024)	2024	Specific	Specific		
This review	2024	General	General	General	General

models, including diffusion models. We even pay attention to privacy issues in multi-modal models. Additionally, there is a lack of in-depth exploration of various privacy attack and defense methods on generative AI models, which prompted us to write this survey.

1.2 Contributions

(1) To the best of our knowledge, we present the first comprehensive technical survey of the privacy works associated with generative AI models.
(2) We identify the existing privacy attack techniques and mitigation methods to defend against these attacks in generative AI models.
(3) At the end of this review, we emphasize open problems and areas of concern that warrant further investigation, such as the memorization capabilities and architectural considerations of generative AI models.

1.3 Paper structure

The paper structure is shown in Fig. 1.

2 General terminology

In this survey, we utilize a variety of acronyms to denote concepts and models prevalent in the fields of privacy and generative AI model research. Each acronym will be defined at its initial mention; however, for convenience, Table.2 presents a compilation of the most frequently encountered and significant terms.

Numerous large-scale datasets are compiled by scraping vast quantities of text from the internet, while others are collected from domain-specific sources, such as the MNIST (LeCun et al. 1998) for handwritten digits classification, CIFAR-10 (Krizhevsky et al. 2009) for image classification tasks, FaceScrub (Schuhmann et al. 2021) and Celeb (Liu et al. 2015) for face recognition, SST-2 (Socher et al. 2013) for sentiment analysis. We endeavor to provide a foundational framework, thereby enabling the academic community to systematically advance the development of benchmarks within the field.

3 Preliminaries

3.1 Large language model

Language models have become much more capable thanks to important advances in pre-training, fine-tuning methods, and prompting strategies. The approach of pre-training and fine-tuning entire models gained significant traction in NLP following the groundbreaking introductions of ELMo (Peters et al. 2018) and ULMFiT (Howard and Ruder 2018). Both of them are built upon the foundation of the Long Short-Term Memory (LSTM) (Hochreiter and Schmidhuber 1997) architecture, which forms the backbone of their powerful capabili-

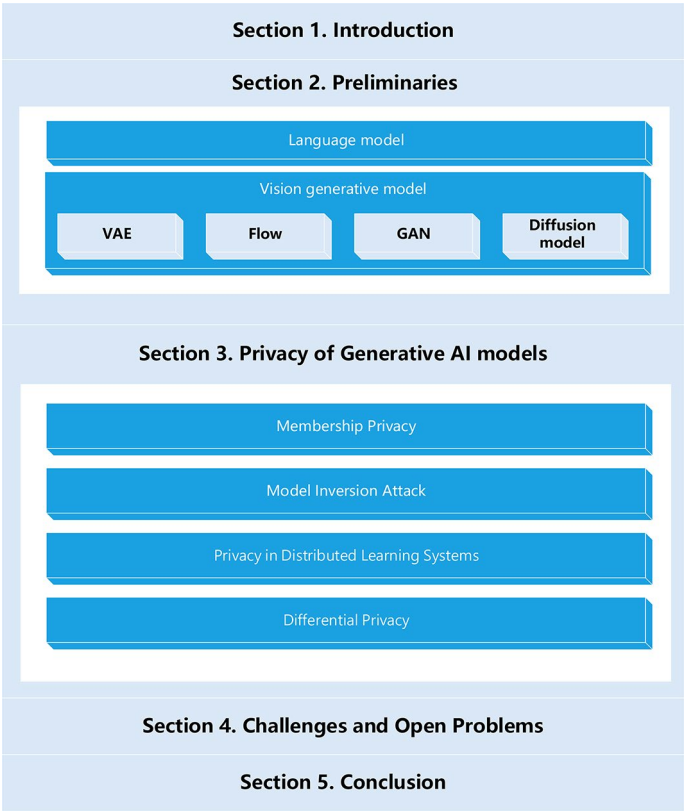

Fig. 1 The structure of this survey

ties. ELMo (Peters et al. 2018) is a deep contextual word representation system based on pre-trained biLMs. Universal Language Model Fine-tuning (ULMFiT) (Howard and Ruder 2018) enables effective transfer learning across various NLP tasks.

The momentum behind this paradigm shift was significantly amplified by Vaswani et al. (2017), which introduced the architecture of Transformer. This innovative design quickly established itself as the go-to architecture for pre-trained language models, leading to a swift and profound transformation in NLP. The Transformer architecture not only initiated a new era of research and innovation but also facilitated the creation of more advanced and resilient natural language understanding and generation systems. As a result, the Transformer has become the foundational element of nearly all leading Pretrained Language Models (PLMs) in NLP, including the GPT series (Brown et al. 2020; Radford et al. 2018, 2019), Gopher (Rae et al. 2021), BERT (Devlin et al. 2019) and its derivatives, XLM-R (Conneau et al. 2020), BART (Lewis et al. 2020), T5 (Raffel et al. 2020), and T0 (Sanh et al. 2022). The widespread adoption of the Transformer underscores its unmatched versatility and effectiveness in advancing NLP, positioning it as the preferred choice for developing cutting-edge Pretrained Language Models (PLMs).

Table 2 List of acronyms commonly used in this survey

Acronym	Description
GAIM	Generative AI model
VGM	Vision generative model
LMM	Large multi-modal model
ML	Machine learning
NLP	Natural language processing
LSTM	Long short-term memory
MLM	Masked language model
PLMs	Pretrained language models
CNNs	Convolutional neural networks
PEFT	Parameter-efficient fine-tuning
LMs	Language models
VAE	Variational auto-encode
GAN	Generative adversarial network
MIA	Model inversion attack
DP	Differential privacy
DP-SGD	Differentially-private stochastic gradient descent
RGP	Reparametrized gradient perturbation
DDPM	Denoising diffusion probabilistic model
CLIP	Contrastive language-image pre-training

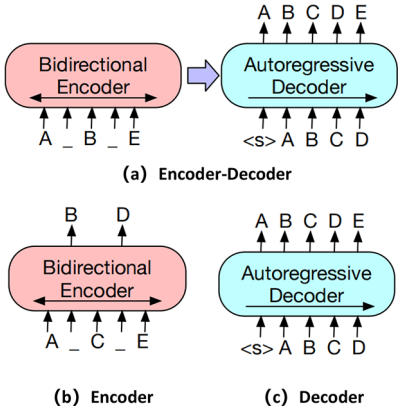

Fig. 2 Different structures of large language models (Lewis et al. 2020)

3.1.1 Large language model architectures

Different structures of Large Language Models are shown in Fig. 2.

- **Encoder-decoder models**

  When considering the structure of an encoder-decoder model, BART (Lewis et al. 2020) and T5 (Raffel et al. 2020) stand out for their innovative architectures. BART (Lewis et al. 2020), with its bidirectional and autoregressive Transformer design, offers a robust solution for processing input sequences and generating coherent output. Similarly, T5 (Raffel et al. 2020) has revolutionized NLP by adapting the Transformer architecture to a text-to-text format, enabling it to tackle diverse tasks such as translation,

summarization, and text classification. These models embody the shift towards versatile and generalizable models that can efficiently handle multiple tasks with a single pre-trained architecture, thus reducing the need for task-specific models and simplifying the integration of machine learning in language processing tasks.

- **Encoder models**

  Encoder models are pivotal in NLP for extracting textual representations. BERT (Devlin et al. 2019) is a key model that employs next-sentence prediction (NSP) and a masked language model (MLM) to comprehend sentence and context relationships. RoBERTa (Liu et al. 2019b) enhanced BERT (Devlin et al. 2019) with hyperparameter tuning and a larger dataset, while ALBERT (Lan et al. 2020) reduced parameters to train large models efficiently, enabling deployment even in resource-constrained settings.

- **Decoder models**

  Modern decoder language models, such as GPT (Radford et al. 2018) and GPT2 (Radford et al. 2019), employ attention mechanisms to weigh input parts for generating coherent output. Subsequently, GPT-4 (Achiam et al. 2023), released in 2023, is a multimodal giant that is trained on vast online data with 175 billion parameters. It can process diverse media, like text and images, into a unified semantic space, enabling tasks such as role-playing and visual question answering with comprehensive context understanding. Moreover, GPT-4 (Achiam et al. 2023) has notably improved in reducing hallucinations or incorrect information compared to previous models.

3.1.2 Language model training

- **Pre-training**

  The concept of pre-training has historical roots in the principles of transfer learning, which draws upon the human ability to reuse knowledge. This idea is realized in practice through pre-training, which has become increasingly popular in CV with the development of deep learning and convolutional neural networks (CNNs) (Zhao et al. 2024). In NLP, pre-training models have emerged to make use of the abundance of unlabeled text data. Pre-trained Language Models (PLMs) are typically structured around a specific architecture and training goal. Lewis et al. (2020) outlined three common configurations: Autoregressive models (like GPT, GPT2/3) predict the next word based on context, Masked models (like BERT, RoBERTa and XLM-R) reconstruct masked sequences, and Encoder models (like BART) reconstruct sequences by filling in missing words. The field of language model development has witnessed a significant shift towards the use of extensive and diverse datasets for pre-training. Initially, models are trained on more limited and curated data sources, such as subsets of Wikipedia, as exemplified by ULMFiT (Howard and Ruder 2018). This has since evolved, with advanced models like XLM-R, GPT-3, and T5 drawing upon vast collections of internet-scraped text, accumulating billions of words across diverse domains. This expanded

data horizon is intended not only to enhance the models' capacity for language production and comprehension in a variety of contexts, as well as the ability to deal with issues like data quality, computational resources, and potential biases.

- **Fine-tuning and prompting**

  A pre-trained model is often fine-tuned by retraining it on a task-specific dataset in order to increase its performance for that task. This is achieved by employing a labeled and smaller dataset to refine the comprehension of the model and align its predictions with the task's needs. Parameter-efficient fine-tuning (PEFT) methods have shown remarkable success in various tasks by updating merely a portion of the model's parameters. Adapters (Houlsby et al. 2019) and Compacter (Karimi Mahabadi et al. 2021) introduced additional trainable components within the T5 model's transformer layers. BitFit (Zaken et al. 2022) focused on bias parameter updates but may not perform as well on larger networks. Prefix-tuning (Li and Liang 2021) refines the model's input through a soft prompt governed by a feed-forward network. Diff pruning (Guo et al. 2021) enabled sparse weight updates but can increase memory consumption. FishMask (Sung et al. 2021) also used sparse updates but is computationally demanding and not optimized for current deep learning infrastructure. LoRA (Hu et al. 2022a; Yang et al. 2024) simplified weight updates with a low-rank matrix approach. $(IA)^3$(Liu et al. 2022) enhanced few-shot learning by adjusting activations through learned vectors. LST (Sung et al. 2022) complemented the pre-trained network with a compact transformer network to minimize training memory.
  Over the past few years, scholars have been exploring prompt-based strategies for enhancing the effectiveness of fine-tuning processes. Such strategies are generally categorized into two streams: prompt-based fine-tuning (FT) and parameter-efficient fine-tuning (PEFT). The first stream, prompt-based FT, involves comprehensive parameter optimization within language models (LMs) to improve performance (Schick and Schütze 2021; Gao et al. 2021; Liu et al. 2023b; Zhang et al. 2022b). Adaprompt (Chen et al. 2022) has notably enhanced the efficacy of prompt-based FT (Schick and Schütze 2021; Gao et al. 2021) on single-sentence tasks by employing standard ongoing pre-training. The second stream, PEFT (Li and Liang 2021; Qin and Eisner 2021; Lester et al. 2021; Su et al. 2021), aims to achieve comparable results with minimal computational resources. PPT (Gu et al. 2022) has attempted to bolster PEFT (Lester et al. 2021) by additional pre-training of the T5 model (Raffel et al. 2020), echoing a concept akin to our own. But this approach depends on a sequence of manually crafted, task-specific modifications for additional pre-training, limiting its flexibility for new, unanticipated downstream tasks (Vu et al. 2022). In contrast, research (Shi and Lipani 2024) presented a consistent design applicable to all tasks, with an emphasis on prompt-based fine-tuning.

3.2 Vision generative models

In this chapter, we briefly introduce different architectures of modern deep generative AI models to help understand the differences in adapting various attack methods to models with different structures. We will provide more detailed information about the specific

victim models under each attack method. Common deep generative AI models are based on architectures such as Transformer, GAN, Diffusion, VAE, Flow, etc. Generally, they can be categorized based on the generation approach into Auto-regressive, Auto-encoding, Explicit Density Estimation, and Implicit Density Estimation. Among them, Transformers are widely used in the implementation of Auto-regressive/Auto-encoding generative language models and multimodal image-to-text models. On the other hand, GAN/Diffusion/VAE/Flow are extensively employed in the construction of visual generative models. We focus on introducing the two popular models, GAN and Diffusion, which have attracted more attention from researchers.

3.2.1 Variational auto-encoders, VAEs

VAE (Kingma and Welling 2013) learns to approximate the data distribution $p_{data}(\boldsymbol{x})$ through variational inference. The approach starts with adding a latent variable $\boldsymbol{z}$. The prior distribution $q(\boldsymbol{z})$ is considered to be a standard normal, while the posterior distribution $q(\boldsymbol{z} \mid \boldsymbol{x})$ is set to be a multivariate Gaussian. This setup leads to an optimization objective that aims to reduce the Kullback–Leibler (KL) divergence between the estimated posterior $q(\boldsymbol{z} \mid \boldsymbol{x})$ and the real posterior $p(\boldsymbol{z} \mid \boldsymbol{x})$. To achieve this, the VAE introduces an encoder $q_\phi(\boldsymbol{z} \mid \boldsymbol{x})$ and a decoder $p_\theta(\boldsymbol{x} \mid \boldsymbol{z})$. The overall optimization objective can then be expressed as:

$$\mathbb{E}_{q_\phi(\boldsymbol{z}|\boldsymbol{x})}[\log p_\theta(\boldsymbol{x} \mid \boldsymbol{z})] + \beta \cdot \mathrm{KL}(q_\phi(\boldsymbol{z} \mid \boldsymbol{x}) \| p(\boldsymbol{z})) \tag{1}$$

During image generation, $\boldsymbol{z}$ is sampled from the normal distribution and mapped to the real distribution $p_{data}(\boldsymbol{x})$ through p_θ. Additionally, VAE employs the reparameterization trick during training to address the non-differentiability issue in the sampling process. While VAE has a simple structure, it suffers from the problem of generating blurry images. Subsequent models such as NVAE (Vahdat and Kautz 2020) and VQ-VAE (Van Den Oord et al. 2017) have improved upon the VAE architecture, significantly enhancing the generation quality.

3.2.2 Flow

Under the framework of maximum likelihood estimation, instead of optimizing the evidence lower bound (ELBO) like VAE, Normalizing Flow is based on the Change of Variables Theorem. Given a distribution $\boldsymbol{z} \sim p(\boldsymbol{z})$ and $\boldsymbol{x} = f(\boldsymbol{z})$, $p_{\boldsymbol{x}}(\boldsymbol{x}) = p_{\boldsymbol{z}}(f^{-1}(\boldsymbol{x}))|\det \frac{\partial f^{-1}(\boldsymbol{x})}{\partial \boldsymbol{x}}|$, a series of deterministic invertible mapping functions are used to gradually transform a simple distribution into an arbitrary complex distribution, ultimately yielding an optimizable negative log-likelihood. Normalizing Flow focuses on constructing the invertible transformation function f. NICE (Dinh et al. 2014) introduced additive coupling layers and enhanced the model's nonlinear expressiveness through partitioning and cross-coupling. RealNVP (Dinh et al. 2016) further improved the expressive power of invertible transformations by using Affine coupling layers and introduced convolutional and multi-scale structures to reduce computational costs. Glow (Kingma and Dhariwal 2018) incorporated reversible 1×1 convolutions for channel mixing to strengthen nonlinear expressiveness. RevNets (Gomez et al. 2017) introduced residual connections into flow models to alleviate gradient vanishing. There is also a class of autoregressive flow models, such as MADE (Khajenezhad et al.

2020) and PixelCNN (Van den Oord et al. 2016), which decompose the joint probability distribution into a product of conditional probabilities and model each conditional probability.

3.2.3 Generative adversarial networks

Generative Adversarial Networks(GANs) (Goodfellow et al. 2014) are a popular type of deep generative model. GANs introduced the ingenious generator-discriminator structure for adversarial training. The discriminator D attempts to distinguish between samples from the training set and fake samples produced by the generator G, assigning high scores to real samples. The generator, in turn, tries to fool the discriminator into giving high confidence scores to its generated samples. The optimization objective can be written as a min-max problem:

$$\min_G \max_D V(D,G) = \mathbb{E}_{\boldsymbol{x}\sim p_{\text{data}}(\boldsymbol{x})}[\log D(\boldsymbol{x})] + \mathbb{E}_{\boldsymbol{z}\sim p_{\boldsymbol{z}}(\boldsymbol{z})}[\log(1 - D(G(\boldsymbol{z})))]. \tag{2}$$

The adversarial training method includes switching updates to the generator and discriminator networks. The process continues until a Nash equilibrium is reached. At this stage, the discriminator can no longer distinguish between the true data distribution and the distribution of the generated samples. This process implicitly minimizes the Jensen-Shannon (JS) divergence between the generated and target distributions. Compared to vanilla VAEs, GANs can generate clear images. However, the adversarial training process in GANs can lead to mode collapse and training instability issues. In the case of mode collapse, the generator may only learn to produce a limited subset of samples that can fool the discriminator, failing to capture the full data distribution. Training instability arises because adversarial training is difficult to converge, especially when the discriminator is too strong or too weak. Subsequent works have focused on improving GANs at different levels.

GANs focus on training stability and controllability: To address the challenges of mode collapse and training instability in GANs, various approaches have been developed. One line of work focuses on the distance metrics, like the Wasserstein distance in WGAN (Arjovsky et al. 2017b), which helps alleviate instability, vanishing gradients, and mode collapse, though it requires abandoning the log loss and certain optimizers. WGAN-GP (Gulrajani et al. 2017) further builds on this by adding a gradient penalty to mitigate the issue of non-uniform weight distributions caused by weight clipping in WGAN. SN-GAN (Miyato et al. 2018) achieves a global 1-Lipschitz constraint through spectral normalization of the weight matrices, which enhances training stability and reduces mode collapse.

In parallel, other GAN variants aim to improve the quality and controllability of generated outputs. DCGAN (Radford et al. 2015) pioneers the use of CNNs for unsupervised learning. cGAN (Mirza and Osindero 2014) and ACGAN (Odena et al. 2017) extend the vanilla GAN by incorporating conditional information like class labels or text embeddings. ProGAN (Karras et al. 2018) employs a progressive growing approach to generate higher resolution images, while SAGAN (Zhang et al. 2019) incorporates self-attention to better capture global and long-range dependencies. The large-scale BigGAN (Brock et al. 2018) further pushes the boundaries of high-resolution (up to 512x512) and high-quality image generation. Finally, the StyleGAN (Karras et al. 2019) and StyleGAN-2 (Karras et al. 2020)

frameworks introduce techniques to disentangle latent features and provide fine-grained style control.

3.2.4 Diffusion model

Diffusion models are currently a popular class of probabilistic generative models in the field of visual generation. Diffusion models view the generation process as a gradual denoising process from a noise distribution, sampling at each denoising step. Compared to the aforementioned generative models, diffusion models have the advantages of simple structure, stable training, and high generation quality, which has led to their widespread application in Artificial Intelligence Generated Content (AIGC). Diffusion models can be studied from perspectives of variational inference such as Denoising Diffusion Probabilistic Models (DDPMs) (Ho et al. 2020). There are also approaches that interpret them from the perspectives of score matching (Song and Ermon 2019) and stochastic differential equations (Song et al. 2020b). From the more intuitive DDPM perspective, instead of introducing a parameterized posterior distribution $q_\phi(\boldsymbol{z} \mid \boldsymbol{x})$, the diffusion model defines the posterior distribution as a Markov process:

$$\begin{aligned} q\left(\boldsymbol{x}_t \mid \boldsymbol{x}_{t-1}\right) &= \mathcal{N}\left(\boldsymbol{x}_t; \sqrt{1-\beta_t}\boldsymbol{x}_{t-1}, \beta_t \mathbf{I}\right) \\ &= \mathcal{N}\left(\sqrt{\frac{\alpha_t}{\alpha_{t-1}}}\boldsymbol{x}_{t-1}, \left(1-\frac{\alpha_t}{\alpha_{t-1}}\right)\mathbf{I}\right), \end{aligned} \tag{3}$$

where $\{\beta_t\}_{t=1}^T$ is the variance sequence and $\alpha_t = \prod_{s=1}^t (1-\beta_s)$, which can be viewed as gradually injecting noise into the data. Subsequently, by optimizing the ELBO:

$$\begin{aligned} &\mathbb{E}_{q(\boldsymbol{x}_1 \mid \boldsymbol{x}_0)}\left[\log p_\theta(\boldsymbol{x}_0 \mid \boldsymbol{x}_1)\right] - D_{\mathrm{KL}}(q(\boldsymbol{x}_T \mid \boldsymbol{x}_0) \| p(\boldsymbol{x}_T)) \\ &- \sum_{t=2}^{T} \mathbb{E}_{q(\boldsymbol{x}_t \mid \boldsymbol{x}_0)}\left[D_{\mathrm{KL}}(q(\boldsymbol{x}_{t-1} \mid \boldsymbol{x}_t, \boldsymbol{x}_0) \| p_\theta(\boldsymbol{x}_{t-1} \mid \boldsymbol{x}_t))\right], \end{aligned} \tag{4}$$

A trained model $p_\theta(\boldsymbol{x}_{t-1} \mid \boldsymbol{x}_t)$ is obtained, and a denoising process $\prod_1^T p_\theta(\boldsymbol{x}_{t-1} \mid \boldsymbol{x}_t)$ is performed to remove the gradually added noise.

Conditional generation and multimodal diffusion models: The method of using a classifier to guide the image generation process, known as classifier guidance, was first proposed by Dhariwal and Nichol (2021). In this method, at each step of denoising, the target classification of the classifier trained on the corresponding noisy data at that moment is injected into the gradient of the input noisy image, avoiding the need for retraining the diffusion model. Similarly, some methods (Liu et al. 2023a; Avrahami et al. 2022; Kim et al. 2022) use text prompts from Contrastive Language-Image Pre-training(CLIP)(Radford et al. 2021) for conditional guidance. However, these methods are prone to sampling failures, and the robustness of the classifier cannot be guaranteed. Additionally, training a classifier under a noise distribution introduces extra overhead.

Ho and Salimans (2021) proposed another guidance method called classifier-free guidance. They directly modify the training process by introducing the target condition c into

the denoising network for joint training. Given the condition c, the optimization process becomes:

$$\min_{\theta} \mathbb{E}_{t,\epsilon} \left[D_{\mathrm{KL}}(q(\boldsymbol{x}_{t-1} \mid \boldsymbol{x}_t, \boldsymbol{x}_0) \| p_\theta(\boldsymbol{x}_{t-1} \mid \boldsymbol{x}_t, c)) \right]. \tag{5}$$

During the inference stage, a linear combination of unconditional and conditional sampling results under specific weights is used as the sampling result at each step. This approach has been inherited by multimodal models such as GLIDE (Nichol et al. 2022), DALL· E 2 (Ramesh et al. 2022), Imagen (Saharia et al. 2022), Stable Diffusion (Rombach et al. 2022), etc.

Some methods focus on how to add new attributes to a pre-trained conditional diffusion model. For example, Textual Inversion (Gal et al. 2022) learns to describe the features of new training images by optimizing the method to search for the minimal training loss conditional text embedding and binding a low-frequency special text to that embedding. Methods like Dreambooth (Ruiz et al. 2023) and LoRA (Hu et al. 2021) choose to fine-tune pre-trained models while mitigating the catastrophic forgetting problem. In addition, ControlNet (Zhang et al. 2023) introduces an auxiliary network to increase spatial control, using zero convolution layers to ensure that no detrimental noise might impact the fine-tuning.

4 Privacy of generative AI models

This section will review how privacy can be compromised in generative AI models and how to protect their privacy. For example, in Sect. 4.1, we primarily explore how to use membership inference attacks on generative AI models and provide a detailed explanation of how to defend against such attacks. The goal of a membership inference attack is to determine whether a particular data is part of the training data of the target model (Shokri et al. 2017). In Sect. 4.2, we discuss model inversion attacks, which aim to reconstruct or infer sensitive attributes from the training samples using the model's predictions (Fredrikson et al. 2014). Given the widespread application of model inversion attacks on classification models, this section will also explore how generative AI models can facilitate model inversion attacks. Both of these forms of privacy can be unlawfully inferred in a fully integrated centralized ML system. To mitigate this problem, distributed learning systems have emerged as effective solutions, enabling geographically distributed data to be processed locally by various participants without the need for data sharing (Hu et al. 2024). However, even in distributed learning systems, privacy issues still exist for generative AI models, which we will explore in Sect. 4.3. Based on the analysis in the previous subsections, differential privacy has emerged as a common defense for generative AI models. Therefore, in Sect. 4.4, we will focus on how to use differential privacy to protect generative AI models.

4.1 Membership privacy

Membership inference attack is a variety of privacy breach that targets machine learning models, particularly those that have been trained on sensitive data (Shokri et al. 2017). For instance, if a generative AI model is developed using medical data from individuals with a

specific illness, an attacker could potentially deduce the victim's health condition by determining if the victim's medical information was utilized in the model's training process. This could reveal health information that the patient would prefer to keep private, as it pertains to their personal medical confidentiality. In a black-box scenario, an attacker can determine whether a particular dataset has been utilized in training the GAIM model simply by analyzing its output. See Fig. 3, which could potentially expose sensitive information about individuals or entities within that dataset. Officially, taking into account a trained machine learning model M, a data sample x, and external adversary knowledge denoted by K. The definition of membership inference attack A is as follows:

$$A : x, M, K \rightarrow \{0, 1\}. \tag{6}$$

In this case, 0 indicates that In this case, 0 indicates that the training dataset M does not contain data sample x, while 1 indicates that it does. Table.3 shows previous work for membership inference attack on generative AI models.

4.1.1 Membership privacy in language models

Language models like GPT and BERT are trained on large datasets of text to predict or generate text sequences. These models can potentially memorize specific details from the training data, which could include sensitive information, such as health records, biometric data, sexual orientation, or any other information shared during the course of interactions with the language models. Since text-generation models tend to memorize specific sequences of words from their training data, attackers can easily exploit this characteristic to perform membership inference attacks. In 2019, Song and Shmatikov (2019) presented a black-box auditing technique that leverages this phenomenon by building a binary classifier to distinguish whether the model has encountered specific user data. They used the same training technique as the target model to train several "shadow models", but with a different auxiliary dataset. As shown in Fig. 4, these shadow models imitate the target model's behavior, to help the auditor understand how it responds to various inputs. With this, they are able to extract features from the outputs of the shadow models according to the model's ranking of target words. After that, a binary membership classifier is trained using these features to determine if the model has seen a certain input sequence during training.

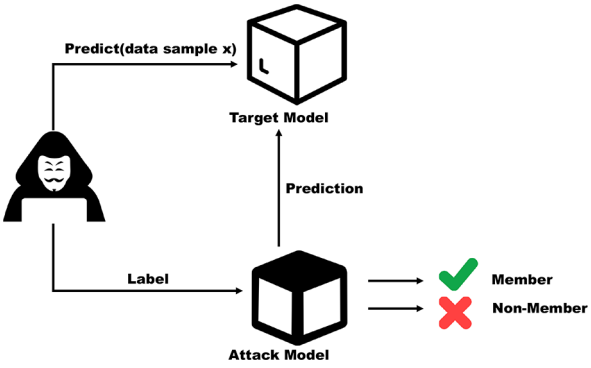

Fig. 3 Membership inference attack in the black-box setting (Shokri et al. 2017)

Table 3 Previous work for membership inference attack on generative AI models

GAIM	Target model	References	Knowl.*	Dataset
LM	Seq2Seq	**SIGKDD [2019]** (Song and Shmatikov 2019)	■	Reddit, SATED and Dialogs
	LSTM	**CCS [2020]** (Song and Raghunathan 2020)	■,□	Wikipedia and BookCorpus
	Seq2Seq	**TACL [2020]** (Hisamoto et al. 2020)	■	WMT18
	BERT, GPT-2	**ARXIV [2021]** (Jagannatha et al. 2021)	■,□	MIMIC-III, UMM, and VHA
	GPT-2	**USENIX Security [2021]** (Carlini et al. 2021)	■	Public Internet
	BERT	**EMNLP [2022]** (Mireshghallah et al. 2022b)	□	MIMIC-III and i2b2
	GPT-2	**ACL [2023]** (Mattern et al. 2023)	■	AG News, Sentiment140 and Wikitext-103
	GPT-2	**NeurIPS [2024]** (Jagielski et al. 2024)	□	CIFAR-10, WikiText103, Purchase100 and Texas100
VGM	DCGAN, BEGAN, VAE	**PoPETs [2017]** (Hayes et al. 2019)	■,□	LFW, CIFAR-10, DR
	WGAN, VAE	**ICDM [2019]** (Liu et al. 2019a)	□	MNIST, CelebA, ChestX-ray8
	GAN, VAE	**PoPETs [2019]** (Hilprecht et al. 2019)	■,□	MNIST, Fashion-MNIST, and CIFAR-10
	GANs, VAE	**CCS [2020]** (Chen et al. 2020b)	■,□	CelebA, MIMIC-III, Instagram New-York
	GANs	**NDSS [2021]** (Zhou et al. 2022)	■	CelebA, AFAD, MNIST, Census Income
	cGAN	**ICCV [2021]** (Shafran et al. 2021)	■	Facades, Maps2sat, Cityscapes, ADE20K, Covid19, Polyp
	Diffusion Model	**ICML [2023]** (Duan et al. 2023)	■	CelebA, CIFAR-10/100, STL10-U, Tiny-IN
	Diffusion Model	**SPW [2023]** (Matsumoto et al. 2023)	■,□	CelebA and CIFAR-10
	Diffusion Model	**USENIX Security [2023]** (Carlini et al. 2023a)	■,□	CIFAR-10
	Diffusion Model	**ICLR [2024]** (Kong et al. 2024)	■	CIFAR-10, CIFAR-100, TinyImageNet
LMM	CNN+LSTM	**NeurIPS [2022]** Hu et al. (2022c)	■	MSCOCO, FLICKR8k, and IAPR TC-12
	LDM, DALL-E mini	**ARXIV [2022]** Wu et al. (2022)	■	LAION-400 M, CC3M, CC12M, YFCC100M
	CLIP	**ICCV [2023]** Ko et al. (2023)	■	LAION-400 M, CC3M, CC12M

* This column is the adversarial knowledge of different attacks. □: white-box. ■: black-box. ■: gray-box

Nevertheless, it is noteworthy that there are several elements that influence the likelihood of a successful attack, including the model's design and training techniques, the model's complexity, and the quantity and variety of training data. One year later, Song and Raghunathan (2020) extended the reach to encompass a greater variety of embedding models, such as word embeddings and sentence embeddings. They presuppose the opponent has access to vocabulary V for word embedding and a target context of words $[w_1, ..., w_n]$, or

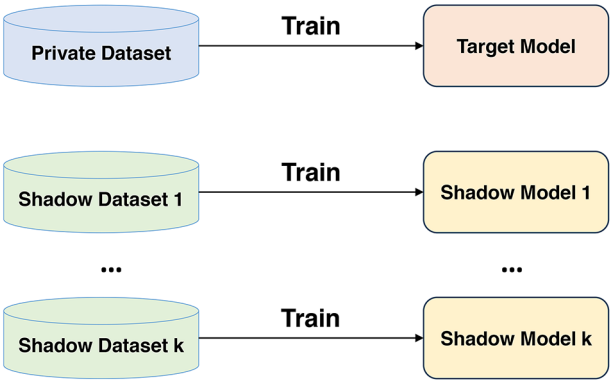

Fig. 4 Training shadow models to simulate the target model's behavior (Shokri et al. 2017)

a context of target sentences (s_a, s_b) and the model M for sentence embedding. In contrast, Hisamoto et al. (2020) focus on sequence to sequence model, employing the construction of shadow models to simulate the target models' behavior, and using these shadow models to train a classifier. The classifier $g(\boldsymbol{x}, e, \hat{e})$ is designed to distinguish $(\boldsymbol{x}, e)$ that are part of the training set and those that are not, where $\boldsymbol{x}$ is input, e and $\hat{e}$ are denoted as the output of target model and shadow model, respectively. The attack accuracy is defined as follows: $accuracy(g, P) = \frac{1}{|P|}\sum^{P}[g(\boldsymbol{x}, e, \hat{e}) = l]$, where P is considered as a probe set that includes $(\boldsymbol{x}, e, \hat{e}, l)$and l are noted as the labels to distinguish in or out. Additionally, Jagannatha et al. (2021) discuss the issue of group-level privacy leaks in clinical language models based on BERT and GPT-2 architectures and evaluate their privacy preservation capabilities through membership inference attacks. These attacks treat collections of patient or admission records as single data samples and estimate privacy leakage by evaluating the mean error of all samples within those groups. In the meanwhile, Carlini et al. (2021) discover that large language models like GPT-2 have a tendency to remember and reveal specific training samples. To tackle this problem, they provide a black-box methodology-based query strategy, which requires removing samples with low likelihood and insufficient accuracy because of flaws in the language models. To more properly quantify the privacy issues associated with memorization in the Masked Language Models(MLMs), Mireshghallah et al. (2022b) suggest an enhanced membership inference attack utilizing likelihood ratio hypothesis testing, which incorporates an extra reference MLM. The likelihood ratio test is distinguished by the subsequent statistic: $L(\boldsymbol{x}) = \log\left(\frac{p(\boldsymbol{x};\,\theta_R)}{p(\boldsymbol{x};\,\theta)}\right)$, where θ and θ_R are denoted as parameters of target model and shadow model, respectively. Furthermore, Mattern et al. (2023) propose a novel attack method known as neighborhood attacks, which contrast the model scores of a target sample with those of artificially generated similar n neighbors $\{\tilde{\boldsymbol{x}}_1, \ldots, \tilde{\boldsymbol{x}}_n\}$, eliminating the requirement to obtain the training data distribution. The decision rule can be interpreted as follows:

$$A_{M_\theta}(\boldsymbol{x}) = \mathbb{1}\left[\left(\mathcal{L}(M_\theta, \boldsymbol{x}) - \sum_{i=1}^{n}\frac{\mathcal{L}(M_\theta, \tilde{\boldsymbol{x}}_i)}{n}\right) < \gamma\right], \tag{7}$$

where γ is denoted as threshold value. The only possible explanation for the target sample's loss value being much lower than the neighbors' losses is overfitting. Due to the rising popularity of model distillation, Jagielski et al. (2024) target "student" models that are distilled from "teacher" models, through experiments, they find that even during the distillation process, privacy information can be indirectly transmitted to the student model via the teacher model's predictions.

4.1.2 Defense techniques for membership privacy in language models

It's evident that membership inference attacks pose a significant threat to language models. To address this challenge, studies (Mireshghallah et al. 2022b; Mattern et al. 2023; Song and Shmatikov 2019; Song and Raghunathan 2020; Hisamoto et al. 2020; Jagannatha et al. 2021; Carlini et al. 2021) have widely adopted the utilization of DP, which introduces carefully calibrated noise into the training process, ensuring that the existence or lack of any individual data point minimally impacts the model's outputs. However, someone failed against the DP model. For example, the auditing algorithm suggested in Song and Shmatikov (2019) performs poorly, with an accuracy that is indistinguishable from random guessing. A similar situation occurs in the work of Song and Raghunathan (2020), due to the use of word embeddings with over 10 million parameters, making the training process with DP and the fine-tuning of hyperparameters nearly impossible to achieve. Nevertheless, the attack methods devised by Jagannatha et al. (2021); Mattern et al. (2023) continue to demonstrate significant effectiveness following the implementation of DP. Additionally, they discovered that while DP can serve as a regularization technique to enhance a model's generalization ability, it may negatively impact the model's accuracy in certain scenarios, particularly when the amount of data is limited. Except for using DP, Carlini et al. (2021) propose several other protection methods. The first way is curating the training data, automatically identifying and filtering out data that contain personally identifiable information (PII) or other sensitive content. Using more sophisticated deduplication techniques to reduce the occurrence of sensitive information within individual documents. Selectively collecting training data can avoid sources known to host private content. Employing this method ensures that some private data will inevitably be disclosed. Second, limiting the memory's influence on downstream applications, and fine-tuning the model on specific tasks may overwrite or reduce the memory of the original training data, but this could also introduce new memorization issues. Moreover, in downstream applications, attempts can be made to filter out generated text that includes memorized content, provided that such content can be reliably detected. Third, auditing ML models for memorization. Conducting empirical analysis to assess the model's privacy protection level. Using membership inference attacks and other existing attack methodologies to test the model and determine its privacy risks in practical applications. However, the auditing process can be resource-intensive and can not completely expose all of the model's privacy vulnerabilities.

4.1.3 Membership privacy in vision generative models

Membership privacy is a concern not only for language models but also for vision generative models such as GANs and diffusion models. These models, while capable of generating realistic images, might inadvertently reveal sensitive information from the training data. In

the context of GANs and diffusion models, membership inference attacks might attempt to identify if a specific image was included in the model's training data. If the model has overfitted to certain data samples, it could inadvertently reproduce or hint at the characteristics of these samples in the generated media, thereby exposing membership information from the training set. Hayes et al. (2019) first introduce a membership inference attack specifically targeted at generative models. The attacks utilize GANs to identify inputs that were included in the training datasets and to detect overfitting via making use of the discriminator's capacity to discover statistical variations in distributions. Liu et al. (2019a) propose co-membership attacks which is a novel approach that considers not only whether a single sample is in the training set but also whether a group of *n* instances collectively belongs to the training set. The reconstruction loss for each of the n instances is averaged to determine the new attack loss:

$$\min_{\theta} \frac{1}{n} \sum_{i}^{n} \Delta(\boldsymbol{x}_i, G(A_\theta(\boldsymbol{x}_i))). \tag{8}$$

L2 distance is taken as distance function $\Delta(.,.)$. The attacker is a neural network A, parameterized by θ, against the generator G. Hilprecht et al. (2019) present two membership inference attacks-the Monte Carlo Attack and the Reconstruction Attack. The Monte Carlo Attack is applicable to any generative model that allows sampling, while the Reconstruction Attack is specifically designed for VAEs. Chen et al. (2020b) introduce a new attack calibration technique that improves the performance of attacks in all considered attack scenarios, which is performed by comparing the reconstruction errors of samples on both the victim model and the reference model. Formally,

$$\mathcal{A}(\boldsymbol{x}, M(\theta)) = \mathbb{1}\left[\log \frac{P(\boldsymbol{x} \in D_{\text{train}} | \boldsymbol{x}, \theta)}{P(\boldsymbol{x} \notin D_{\text{train}} | \boldsymbol{x}, \theta)} \geq 0\right], \tag{9}$$

where P is the probability of predicting whether $\boldsymbol{x}$ is included in the training dataset. Zhou et al. (2022) propose a novel attack aims at inferring macro-level properties of the training datasets used by GANs, which also shows how property inference attacks (Ganju et al. 2018) can be utilized to enhance membership inference attacks. Mathematically, this property inference attack can be denoted as follows: $\phi\left(\{f_p(G_{target}(\boldsymbol{z}_i))\}_{i=1}^{|\boldsymbol{x}|}\right)$. These obtained samples $\boldsymbol{x}$ are consumed by f_p and subsequently, prediction of property classifier function ϕ to obtain the attack result p. These samples were created from a random latent coding set $\boldsymbol{z}$. Based on previous work, Shafran et al. (2021) propose a hybrid approach to membership inference attack by combining reconstruction errors with image predictability errors: $L_{mem}(\boldsymbol{x}, y) = L_{rec}(\boldsymbol{x}, y) - \alpha \cdot L_{pred}(\boldsymbol{x}, y)$, where $\boldsymbol{x}$ presents image in input domain, y presents ground truth in output domain. The computation of L_{mem} involves deducting the reconstruction error L_{rec}, weighted by α, from the predictability error L_{pred}. This dual measure provides a more accurate assessment of whether the model has memorized the training data.

Unlike GANs, most existing membership inference attack methods are not applicable to diffusion models due to their distinct generative processes and characteristics. Duan et al. (2023) present a method called Step-wise Error Comparing Membership Inference (SecMI), which is tailored to the characteristics of diffusion models. It infers the membership of sam-

ples via evaluating the posterior estimate of the forward process matching at each timestep. Instead of focusing on proposing a new attack method, Matsumoto et al. (2023) evaluate if diffusion models are susceptible to membership inference attack by comparing them with other types of generative models, specifically GANs. This comparative approach offers a broader understanding of how various model architectures, including the unique aspects of diffusion models, affect privacy protection. Moreover, Carlini et al. (2023a) propose a hybrid approach that combines image generation with a filtering process to conduct the attack. This not only generates a large number of image samples but also identifies and extracts samples that closely resemble images in the training set. Nevertheless, the method proposed by Carlini et al. (2023a) needs more queries to the model, resulting in longer attack times and higher computational consumption. Conversely, the Proximal Initialization Attack (PIA) proposed by Kong et al. (2024) requires only two queries. This method uses the model's intermediate outputs during the diffusion process to ascertain if a given sample is included in the training set. The attack starts with the model's output at time t = 0, which is treated as the noise ϵ, and then compares the ground truth trajectory with the predicted points. PIA measures the discrepancy between these points to assess the likelihood that the sample originated from the training data.

4.1.4 Defense techniques for membership privacy in vision generative models

Similar to language models, vision generative models also exhibit vulnerability to membership inference attacks. To protect vision generative models against membership inference attacks, DP is also widely applied (Hayes et al. 2019; Chen et al. 2020b; Shafran et al. 2021; Wu et al. 2022; Duan et al. 2023). Adding noise throughout the training phase, DP prevents attackers from inferring whether a given image is part of the training dataset according to the vision generative model's output. The addition of noise ensures that small changes in the vision generative model's output do not accurately reflect the presence or absence of individual samples, thereby enhancing the vision generative model's privacy protection capability. However, Shafran et al. (2021), Duan et al. (2023) found that applying DP during the training process of a diffusion model can make it difficult for the DDPM to converge, such as causing the model to output meaningless information. On the contrary, applying DP during the training of a GAN has been found to perform well. For instance, when using the methods of Hayes et al. (2019); Chen et al. (2020b) to attack a GAN that has been trained with DP (with ϵ >10), the success rate is quite high. While it continues to diminish the impact of membership inference attacks. In addition to applying DP, in the work of Hayes et al. (2019), they propose two different defense methods, the former is Weight Normalization, which reparameterizes weight vectors that decouple the weights' length from their direction, incorporated into each layer in the target GAN model's generator and discriminator. The latter is Dropout, which is a technique to eliminate overfitting by arbitrarily removing connections between neurons during training. In this study (Hayes et al. 2019), Dropout with a probability of 0.5 was utilized on each discriminator layer. However, using Dropout can significantly prolong the training process, necessitating additional epochs to produce qualitatively satisfactory samples. Moreover, Weight Normalization can often result in fluctuations during training.

4.1.5 Membership privacy in multi-modal models

Multi-modal models, which process images, speech, and text, are similarly vulnerable to membership inference attacks. Attackers can use correlations between modalities to infer if a data sample is included in the training set. For instance, in image captioning, analyzing the relationship between descriptions and images can reveal training membership. In image-to-text translation, observing model outputs can similarly expose the presence of specific images in the training data.

Hu et al. (2022c) first introduce two membership inference attack methods tailored for multimodal models: Metric-based Membership Inference (MBM^4I) and Feature-based Membership Inference (FBM^4I). The MBM^4I method relies on comparing the model's output to the training dataset to infer membership. For image captioning tasks, this involves evaluating the similarity between the model-generated captions and known reference captions using text similarity metrics like ROUGE or BLEU. By training a shadow model and using its outputs as a benchmark, attackers can establish a threshold to differentiate between member and non-member data. FBM^4I, on the other hand, focuses on analyzing the feature representations learned by the multimodal model. This method involves training a feature extractor that processes paired image and text data to learn the intrinsic connections between them. Attackers use this extractor to obtain the feature representations of both the input image and the model's output text, and the differences in these features are used to infer membership status. This method does not rely on the availability of reference texts, making it effective even when reference texts are unavailable or the model's architecture is unknown. Experimental results suggest that FBM^4I generally outperforms MBM^4I in attacking multimodal models. To address the risk of membership leakage, Hu et al. (2022c) apply data augmentation and L2 regularization. However, the effectiveness of data augmentation as a defense is reduced in specific scenarios, for instance, when the target model is trained on the IAPR dataset (Makadia et al. 2008). Additionally, Hu et al. (2022c) have attempted to incorporate DP during the model training process. While this approach does weaken membership inference attacks, it unfortunately leads to poor model performance in terms of output quality.

Wu et al. (2022) propose four membership inference attack methodologies targeting text-to-image generation models. These attacks are based on three insights: (i) training set pairs should have higher image-generation quality than test set pairs; (ii) the reconstruction error between the generated image and the original image from the training set should be lower than from the test set; (iii) the generated image should more accurately reflect the semantics of the training set textual caption than that of the test set. The methodologies include Attack I-P and Attack I-S, which are based on the first insight, focusing on the quality of generated images. Attack I-P uses pixel-level differences, while Attack I-S leverages semantic-level embeddings from a pre-trained vision-language model. Attack II-P and Attack II-S, grounded in the second insight, measure reconstruction errors. Attack II-P examines pixel-level discrepancies, and Attack II-S uses semantic embeddings to determine errors. Attack III, focusing on the third insight, assesses the faithfulness of the generated image to the semantic content of the text caption using semantic embeddings. Finally, Attack IV integrates all three insights, utilizing semantic-level discrepancies to create a comprehensive attack feature set that feeds into the attack model to predict membership status. Empirical results show that all proposed attacks are significantly effective, underscoring the severe

privacy risks membership inference poses to text-to-image generation models. In an effort to mitigate the impact of membership inference attacks on the model, Wu et al. (2022) have restricted the number of data samples available from the target training dataset. However, they have observed that by diminishing the dataset to just 5% of its original volume, the proposed attacks experience only a marginal decline in effectiveness and remain significantly potent. Therefore, it is concluded that curtailing the adversary's access to member samples does not substantially hinder the attack performance.

Ko et al. (2023) present three practical membership inference attack strategies against large-scale multi-modal models, such as CLIP (Radford et al. 2021), which are trained on extensive datasets. The first strategy, Cosine Similarity Attack (CSA), utilizes the model's tendency to maximize cosine similarity on training data, predicting membership based on the thresholded cosine similarity between text and image features. The second approach, Augmentation-Enhanced Attack (AEA), enhances the baseline by applying various transformations to target samples and aggregating the resulting cosine similarity changes, leveraging the observation that member samples exhibit a more significant drop in similarity post-transformation than non-members. Lastly, the Weakly Supervised Attack (WSA) utilizes one-sided non-member information, such as data published after the model's release, to create a pseudo-member set. Using this set, WSA trains an attack model that predicts membership, achieving improved accuracy and demonstrating particular effectiveness at low false-positive rates. These attacks draw attention to multi-modal models' privacy flaws without requiring access to the model's training process or architecture, presenting a significant step toward understanding and mitigating privacy risks in large-scale AI systems. Ko et al. (2023) have also attempted to use established defensive techniques to counter membership inference attacks. Experiments revealed that neither L2 regularization nor data augmentation were able to reduce the accuracy of membership inference attack CLIP (Radford et al. 2021). Additionally, they have considered a straightforward defense strategy of injecting noise into the output features of a pre-trained CLIP (Radford et al. 2021). They found that noise with a standard deviation of at least $\sigma = 0.5$ is required to weaken the attacks, but this simultaneously leads to a significant degradation in the performance of the CLIP (Radford et al. 2021).

4.1.6 Jailbreaking privacy attacks on large language models

While membership inference attacks typically focus on evaluating a model's predictions to ascertain if a certain data item is included in the training dataset (Shokri et al. 2017), jailbreaking privacy attacks are uniquely directed at generating privacy content through the strategic construction of inputs. Researchers have developed several techniques to leverage these vulnerabilities. For example, Huang et al. (2022) investigate whether pre-trained language models (PLMs) disclose email addresses when the owner's name appears in prompts or circumstances pertaining to the email address. The study reveals that PLMs do indeed disclose private data, which is attributed to their ability to memorize data. Moreover, Li et al. (2023) propose a novel method that involves a multi-step jailbreaking prompt to bypass the ethical and safety mechanisms of LLMs. This technique involves integrating jailbreaking prompts into a three-part dialogue between the user and ChatGPT: inputting jailbreak prompts, confirming the activation of jailbreak mode, and then submitting queries on the user's behalf. This tricks the model into a 'Developer Mode', where it is more likely to gen-

erate personal information. In addition, Deng et al. (2024) present an automated jailbreaking privacy attack method, which leverages carefully crafted prompts to bypass or "jailbreak" the chatbots' security safeguards. This approach begins by identifying potential vulnerabilities in LLMs through empirical research, then uses timing analysis techniques to reverse engineer and understand the internal defense mechanisms of chatbots. Subsequently, Deng et al. (2024) develop automated tools to generate prompts that can trick LLMs into producing privacy content that violates policies. A three-step workflow-dataset construction and enhancement, continuous pre-training and task adaptation, and reward ranking fine-tuning is employed to further enhance the ability to generate effective jailbreak prompts. Ultimately, this method not only reveals the vulnerabilities of LLMs in terms of privacy protection but also promotes the development of more robust defenses for these intelligent systems through responsible disclosure. Even more astonishingly, Nasr et al. (2023) discover that by prompting the model with certain phrases, such as asking it to repeat a word many times, the model would eventually diverge from its normal behavior and start to emit verbatim examples from its pre-training data. Once the model diverges, it can start to generate outputs that are copied directly from the training data. They can then collect these outputs to extract the training data.

4.2 Model inversion attack

Data-driven machine learning has been widely adopted due to its exceptional predictive capabilities. Today, cloud-based Machine Learning as a Service (MLaaS) platforms are extensively deployed, allowing users to upload their private data for model training and granting them query privileges. Through public HTTP(S) interfaces and ML APIs, users can obtain model predictions. These services are often applied in privacy-sensitive domains such as lifestyle choices, identity recognition, medical diagnosis, and pharmacogenetics. However, malicious attackers can potentially infer users' private information, such as facial features, sexual habits, or genetic markers, by merely accessing these interfaces (Fredrikson et al. 2014, 2015). This scenario is typically a black-box setting, where attackers have limited access to model outputs.

In another context, with the rise of artificial intelligence, open-source online model publishing platforms like Hugging Face, TensorFlow Hub, and ModelDepot have emerged. On these platforms, models potentially containing private or confidential information can be freely published and downloaded (Chen et al. 2021). Victims' private data, such as personal images, might be used without consent to train these models, which are then disseminated through model releases. In this white-box setting, malicious attackers who obtain the model have full access to its parameters, enabling them to infer private and confidential information that others may not wish to disclose.

Model Inversion Attack (MIA) is a method of revealing training data. In the white-box scenario, an attacker can recover partial attributes or directly reconstruct training samples from model parameters and the target loss function. In the more difficult black-box scenario, an attacker can also abuse access to a model's API on the internet by collecting a large number of soft/hard labels to steal user privacy from the private dataset, where soft labels expose the confidence score of a class, an attack by querying hard labels only yields labels in the form of one-hot coding that discards more information, testing the construction of the attack methodology.

Unlike membership inference attacks, which determine whether a sample belongs to a dataset and can be viewed as a binary classification problem, model inversion attacks focus on recovering partial or complete sensitive attributes of the training samples. For instance, attackers can reconstruct facial features of a specific ID in the training set by rebuilding images with the aid of auxiliary information (Fredrikson et al. 2015; Zhang et al. 2020; Aïvodji et al. 2019) (e.g., blurred or partially occluded facial images), or optimize a randomly selected initial generated sample via generative AI model from a public dataset to approximate training samples. Compared to membership inference attacks, model inversion attacks reveal sensitive information in a more comprehensive manner. Fredrikson et al. (2014) first propose model inversion attacks and a model-agnostic model inversion attacks method that recovers sensitive attributes on linear models by maximizing the posterior probability.

Subsequently, Fredrikson et al. (2015) extend the victim models, attacking decision trees, shallow neural networks, and other models. Based on the correlation between sensitive features and model outputs, they optimize the input under a given model to maximize the likelihood of the corresponding classification. Specifically, they provide two attack methods. The first starts from an initial vector and gradually optimizes the input vector through gradient descent to maximize the confidence of the target label, with denoising post-processing after each optimization step, known as the reconstruction attack. Given the classification model $\tilde{f}$ and target label, the cost function $c(\cdot)$ and image iteration process are defined as follows:

$$c(\boldsymbol{x}) \stackrel{\text{def}}{=} 1 - \tilde{f}_{label}(\boldsymbol{x}) + \text{AUXTERM}(\boldsymbol{x}) \tag{10}$$

$$\boldsymbol{x}_i \leftarrow \text{PROCESS}(\boldsymbol{x}_{i-1} - \lambda \cdot \nabla c(\boldsymbol{x}_{i-1})) \tag{11}$$

where the AUXTERM uses any available auxiliary side information of target label to inform the cost function. Following each gradient descent step, the feature vector that results is sent to a post-processing function PROCESS. This function can carry out different image alterations such as sharpening and denoising.

The second attack method uses a blurred, unrecognizable face image as the initial input and prior knowledge to facilitate the optimization process. After the attack, human judgment is introduced to quantify whether the reconstructed specific identity face image approximates the training sample. Under the Softmax regression model, their MIA yielded an identification rate as high as 87% and an overall accuracy of 75%. However, for the three models, they attacked (Softmax Regression, Multilayer Perceptron, Stacked Denoising Autoencoder), despite applying denoising post-processing functions, the reconstructions often only produced blurred and low-quality face images that failed to be consistent with the training distribution. Similar to gradient-based adversarial examples (Szegedy et al. 2013), gradient descent optimization of high-dimensional input vectors can obtain some target features but often optimizes to adversarial examples or noise images. Therefore, a natural idea is to start the optimization from a low-dimensional manifold (Nguyen et al. 2017; Zhao et al. 2018; Song et al. 2018; Lang et al. 2021; Jacob et al. 2022) that encodes features and map this latent code back to the data sample space through a generator after executing MIA (Zhang et al. 2020; Chen et al. 2021; Wang et al. 2021; Struppek et al. 2022). This becomes the benchmark scheme for a series of subsequent generative model-based MIAs. Next, we

will introduce several different approaches that incorporate generative models into MIA. Table.4 shows previous work for vision model inversion attack.

4.2.1 Applying generative models to MIA based on latent space search optimization

Optimizing in the latent code of generative models can alleviate the issue of optimization results lacking semantics. Models like VAEs, GANs, Flows, and Diffusions can all serve as accomplices for adversaries to steal private data. Among them, GANs are the most widely applied generative models in model inversion attacks. Figure 5 presents the general method of white-box attack based on GAN.

4.2.2 Generative models for MIA in the white-box scenario

Zhang et al. (2020) point out the issues with the aforementioned optimization process. Due to the non-convexity of neural networks, optimization can easily get stuck in local optima, and optimizing high-dimensional data tends to generate unrealistic features lacking semantic information. Their proposed GMI has two stages: the public knowledge distillation stage introduces a public dataset with broad knowledge, and trains a generative model whose distribution is close to that of the private dataset. The loss term is defined as the joint loss of the canonical Wasserstein GAN training loss:

$$\min_{G} \max_{D} L_{\text{wgan}}(G, D) = E_{\boldsymbol{x}}[D(\boldsymbol{x})] - E_{\boldsymbol{z}}[D(G(\boldsymbol{z}))] \tag{12}$$

and a diversity loss:

$$\max_{G} L_{\text{div}}(G) = E_{\boldsymbol{z}_1, \boldsymbol{z}_2}\left[\frac{\|F(G(\boldsymbol{z}_1)) - F(G(\boldsymbol{z}_2))\|}{\|\boldsymbol{z}_1 - \boldsymbol{z}_2\|}\right] \tag{13}$$

then the optimization objective is $\min_G \max_D L_{\text{wgan}}(G, D) - \lambda_d L_{\text{div}}(G)$.

During the secret revelation stage, the latent variable $\boldsymbol{z}$ is optimized via the joint loss of L_{prior} and L_{id}:

$$\hat{\boldsymbol{z}} = \arg\min_{\boldsymbol{z}} L_{\text{prior}}(\boldsymbol{z}) + \lambda L_{\text{id}}(\boldsymbol{z}) \tag{14}$$

where L_{prior} is the discriminator's adversarial loss $-D(G)$, and L_{id} is the negative log-likelihood of the corresponding identity id. Finally, the generator G projects $\boldsymbol{z}$ to the high-dimensional space to obtain the revealed face image. Additionally, they incorporate auxiliary information such as blurred or missing facial regions as conditional inputs to the generator to assist optimization.

This method does not fully leverage the public data to distill knowledge from the target model during training. Furthermore, the inversion process can only obtain a simple mapping from the latent space to the pixel space, leading to the attack only retrieving one sample for each target label, while in reality a target classification may have multiple training samples, and the mapping from samples to target labels should be many-to-one.

To address the first issue, Chen et al. (2021) propose KEDMI, which annotates the public dataset P_{pub} with the target model during the training stage and uses the soft labels to super-

Table 4 Previous work for vision model inversion attacks

References	Approach	Generator	Knowl.*	Public and private datasets
CVPR [2020] (Zhang et al. 2020)	GMI	WGAN	□	MNIST, Chest X-ray, CelebA
ICCV [2021] (Chen et al. 2021)	KEDMI	Inversion-Specific GAN	□	CIFAR-10, MNIST, Chest X-ray, CelebA, FFHQ, FaceScrub
NeurIPS [2021] (Wang et al. 2021)	VMI	DCGAN, StyleGAN, Flow	□	MNIST, EMNIST, Chest X-ray, CelebA
ICML [2022] (Struppek et al. 2022)	PPA	StyleGAN2, BigGAN	□	CelebA, FaceScrub, FFHQ, Stanford Dogs
AAAI [2023] (Yuan et al. 2023)	PLGMI	cGAN/SN-GAN	□	CelebA, FFHQ, FaceScrub
CVPR [2023] (Nguyen et al. 2023)	LOMMA	GAN	□	MNIST, CelebA, FFHQ, CIFAR-10
MM [2023] (Qi et al. 2023)	DMMIA	StyleGAN2	□	MNIST, CelebA, FaceScrub, Stanford Dogs, FFHQ, AFHQ
ARXIV [2019] (Yang et al. 2019)	AMI	Auto-Encoder(Decoder-Only)	■ △	MNIST, CelebA, FaceScrub, CIFAR-10
ECCV [2022] (Yuan et al. 2022)	SecretGen	WGAN	□■	CelebA, FaceScrub
NDSS [2022] (An et al. 2022)	MIRROR	StyleGAN	□■	VGGFace, VGGFace2, CASIA-WebFace
TDSC [2023] (Ye et al. 2023)	C2FMI	StyleGAN2	■	CelebA, CASIA-WebFace, FaceScrub
TDSC [2023] (Tian et al. 2023)	SMI	cGAN	■	MNIST, Fashion-MNIST, CIFAR-10, CelebA
CVPR [2023] (Han et al. 2023)	RLBMI	WGAN	■	CelebA, FFHQ, FaceScrub, PubFig83
CVPR [2022] (Kahla et al. 2022)	BREPMI	GAN	■ △	CelebA, FFHQ, Facescrub, Pubfig83
NeurIPS [2024] (Nguyen et al. 2024)	LOKT	ACGAN	■ △	CelebA, FFHQ, Facescrub, Pubfig83
TIFS [2024] (Liu et al. 2024b)	DMI	Diffusion with Classifier-Free Guidance	■ △	MNIST, CelebA, FaceScrub

* This column is the adversarial knowledge of different attacks. □: white-box. ■: black-box. △: label-only

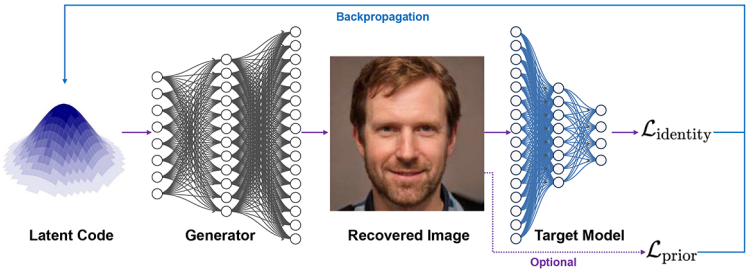

Fig. 5 Typical white-box MI attack pipeline

vise the discriminator to output a distribution consistent with the target model. They also introduce $L_{entropy}$ to reduce the uncertainty of the generator's output. During the revelation stage, instead of optimizing a Dirac distribution that can only produce a one-to-one mapping like GMI, KEDMI directly optimizes the learnable parameters μ and σ of a Gaussian distribution $\mathcal{N}(\mu, \sigma)$ in the latent space, and samples multiple images belonging to a certain class from this distribution, ensuring diversity in the attack results.

From a variational inference perspective, Wang et al. (2021) provide the optimization objective for the latent variable by replacing the prior term with the KL divergence between $p_{Aux(z)}$ and the variational distribution $q(\boldsymbol{z})$, and searching for a joint distribution $q(\boldsymbol{z}_1, \boldsymbol{z}_2, ..., \boldsymbol{z}_l)$ in the extended $\boldsymbol{z}$ space of StyleGAN2. They also validate the method on a flow model. This method discards the discriminator, effectively avoiding mode collapse to point estimates.

The above methods require introducing a prior loss term to constrain the model to generate natural samples close to the public dataset. Struppek et al. (2022) argue that due to the distribution shift between public and private data, forcibly aligning the private data distribution with the public data distribution would introduce irrelevant features and degrade generation quality. Therefore, they remove the prior loss term. Without changing the generator parameters, the optimized latent code can be decoded into natural images when searched in the proper direction. They verify the generality of this idea on BigGAN and StyleGAN2. Furthermore, they address the neural network robustness issue by introducing a Poincaré loss and data augmentation to avoid optimization falling into local optima and generating adversarial examples.

Some other methods have also been proposed to tackle issues in model inversion attacks. Yuan et al. (2023) introduce high-confidence pseudo-labels as conditional embeddings into cGANs during training, arguing that high-confidence images from the public dataset intersect with the private dataset and can thus leak its information, using pseudo-labels can better incorporate information from the target model, and they also use a max-margin loss to mitigate vanishing gradients. Nguyen et al. (2023) focus on the identity overfitting issue during inversion.

While Qi et al. (2023) focus on the issues of catastrophic forgetting and decreased generation diversity brought by optimizing generators, they introduce two additional constraint terms, L_{imr} and L_{idr}, defined by the Intra-class Multicentric Representation (IMR) and Inter-class Discriminative Representation (IDR) modules, in addition to the cross-entropy loss L_{ce}. The IMR consists of a learnable parameter matrix that represents multiple concepts for the target classification, and the IDR utilizes a memory bank to store features of

historical training samples. These two modules aim to enhance sample diversity and generate discriminative features.

4.2.3 Generative models for MIA in the black-box scenario

Researchers' attention has recently turned to more practical model inversion attacks in the black-box setting. Yang et al. (2019) propose an autoencoder-like structure, cleverly treating the target model F_w as the encoder and the *m*-truncated prediction vector trunc $(F_w(\mathbf{a}))$ which *m*-largest truncated vector is preserved. They train a decoder G_θ which minimizes the objective:

$$\mathcal{C}(G_\theta) = \mathbb{E}_{\mathbf{a}\sim p_a}[\mathcal{R}(G_\theta(\text{trunc}_m(F_w(\mathbf{a}))), \mathbf{a})] \tag{15}$$

where $\mathcal{R}$ is reconstruction loss as the inversion model on the public dataset p_a to output the reconstructed data $\hat{\boldsymbol{a}}$. However, due to the nature of autoencoders, this can only produce blurred generation results. Ye et al. (2023) propose a similar method in the first stage of their approach. They first train a feature extractor $\mathcal{E}$ and an inverse network $\mathcal{M}$ using public data to map the image $\boldsymbol{x}$ and label y_c to the coordinates $\xi \in \mathcal{F}$ in the same low-dimensional manifold space. Subsequently, they compute the gradient $\nabla_{\mathbf{w}}\mathcal{L}\left(\xi, \mathcal{E}(G(\mathbf{w}))\right)$, where $\xi = \mathcal{M}(y_c)$, to optimize the variable $\mathbf{w}$ of StyleGAN, so that the features of the generated images were aligned with the low-dimensional vectors obtained by the feature extractor.

For GANs, similar to GMI, Yuan et al. (2022) also train a generator G on the public dataset. In the black-box setting where the target model is inaccessible, they use a feature extractor trained on the public dataset to provide the diversity loss. With corrupted images as prior knowledge, they sample a large number of latent variables from the latent space to generate a batch of recovered images. These images are then transformed and used to generate pseudo-labels, from which the most robust samples matching the corresponding labels are selected after transformation. Finally, the corresponding latent variables are optimized by discrimination loss $\mathcal{L}_{dist}$ and $\mathcal{L}_{id}$ (only for white-box setting with backward propagation). In the black-box setting, they utilize a memory bank denoted as z_{bank_y} to store each sampled latent code *z*. From this bank, they select the latent code *z* with the highest confidence for label *y* as a candidate sample to optimize $\mathcal{L}_{dist}$, this approach aims to improve the identification performance by selecting the most representative latent codes for each label.

There are also some methods that do not require optimization-based search. A genetic algorithm is suggested by An et al. (2022) to explore the latent space using scores derived from a black-box target model. In the second stage of the method proposed by Ye et al. (2023), they use Differential Evolution combined with the confidence of the target classification to continuously optimize the intermediate latent variable $\mathbf{w}$ of StyleGAN.

Recently, there have been methods combining reinforcement learning with generative models. Han et al. (2023) introduced Soft Actor-Critic and confident soft labels for latent space search, where an agent generates a guidance vector as input to a GAN trained on public data to generate recovered samples.

Some methods focus on the more challenging label-only attack scenario. Similar to zeroth-order gradient optimization, Kahla et al. (2022) start from an initial sample point and estimate a gradient $\widehat{M_{c^*}}(z, R)$ by:

$$\widehat{M_{c^*}}(z, R) = \frac{1}{N}\sum_{n=1}^{N} \Phi_{c^*}(z + Ru_n)u_n, \tag{16}$$

where u_n represents a uniformly randomly sampled point over a sphere with a radius of R, and N denotes the total number of points sampled on that sphere. $\Phi_{c^*}(\cdot)$ assigns a value of 0 to inputs that hit the target label and -1 otherwise. The latent variable $\boldsymbol{z}$ is updated along the estimated gradient direction, and as R increases, the generated samples move further away from the classifier's decision boundary and closer to the centroid of the target label.

In LOKT (Nguyen et al. 2024), a label-only attack method based on knowledge transfer was proposed. Similar to SMI, which uses the class labels predicted from public data to supervise the training of cGAN for inverting target privacy attributes, they employ ACGAN as the training framework, where the combination of the discriminator and the classifier serves as a surrogate model. The generator generates fake samples under conditional guidance, and the target model is used as an oracle to obtain pseudo-labels. In addition, they only used oracle to annotate fake images, while public data was only used to reduce the identification loss, which effectively alleviated the problem of class imbalance on public data.

Different from the above approaches, recently, there have been some diffusion model-based MI methods. Kansy et al. (2023) propose using the outputs of a face recognition model as conditions to train a conditional diffusion model for model inversion, though they do not focus on attacks. Similarly, Liu et al. (2024b) use the hard labels of public data as conditions to train a conditional DDPM. Since conditional generation does not involve gradient estimation, they need to sample a large number of samples, apply transformations, query the target model, and select robust samples as the attack results.

4.2.4 The necessity of GAIM in model inversion attack

In the field of AI security, gradient optimization is a method used in white-box settings to obtain natural unrestricted adversarial examples (Zhao et al. 2018; Song et al. 2018) and perform counterfactual explanations (CE) (Wachter et al. 2017). The former seeks a natural sample on the image manifold that causes the classifier to produce an erroneous output. While the latter attempts to minimize a semantic or feature-based $\boldsymbol{\delta}$ for a given sample point $\boldsymbol{x}$ and a class c not belonging to $\boldsymbol{x}$, such that $\boldsymbol{x} + \boldsymbol{\delta}$ is classified as c without being deemed an adversarial sample (Wachter et al. 2017; Dhurandhar et al. 2018). In both CE and model inversion attack, the optimized sample's class changes. However, CE requires minimal semantic alterations, while MIA does not have this constraint and aims to obtain a sample with high confidence score for the target classifier. This is reflected in MIA-generated samples being further from the target model's decision boundary compared to CE-generated samples (Kuppa and Le-Khac 2021; Kahla et al. 2022). Despite these methods having different optimization objectives and initial conditions, they all require that the generated images do not deviate from the image manifold. A common approach is to conduct the optimization process in the latent space of vision generative models (Song et al. 2018; Lang et al. 2021; Zhang et al. 2020). However, improper optimization may still cause the optimized latent variables to deviate from their assumed distribution, leading to optimization in an adversarial direction.

Moreover, the variables being optimized are widely sampled from the latent space of public data. Therefore, when the distribution discrepancy between public and private data

is significant or non-overlapping, the attack becomes challenging. This issue is particularly prevalent in GANs, despite their ability to generate high-fidelity samples. Due to mode collapse in GANs (Goodfellow et al. 2014; Arora et al. 2017), the generator's output diversity is limited and may not cover the entire distribution of public data. Compared to GANs with adversarial training, diffusion models offer better training while ensuring generation diversity and quality (Dhariwal and Nichol 2021). In the field of white-box visual counterfactual explanations, some papers have applied diffusion models to generate high-fidelity counterfactual explanations (Jeanneret et al. 2022; Augustin et al. 2022; Jeanneret et al. 2023, 2024). However, the potential of diffusion models in MIA has not been fully explored.

4.3 Privacy in distributed learning systems

4.3.1 Privacy concerns in language models for distributed learning

Distributed learning systems, particularly those used for language models, strive to safeguard user data and privacy while maintaining high performance. These systems are engineered to train models using data from multiple clients, such as devices or servers, without centralizing the data, thereby mitigating privacy risks (Hu et al. 2024; Saha et al. 2024). The model in federated learning is trained on several decentralized devices with local data samples. Model updates, such as gradients or parameters, are exchanged among the devices rather than the data itself, minimizing the requirement for data to leave the device and thus enhancing privacy. For instance, multiple hospitals can collaboratively train a model without sharing their patients' medical data (Jochems et al. 2017). However, in a federated learning scenario, language models can still be subjected to various threats. According to Zhu et al. (2019), participant-shared gradients during training might allow for the leakage of private training data (As shown in Fig. 6). This is contrary to the common belief that sharing gradients doesn't compromise user privacy. They introduce an attack called Deep Leakage from Gradients (DLG), which is capable of extracting training inputs x and labels y from the gradients ∇W. This is achieved by optimizing dummy inputs $\mathbf{x}'$ and labels $\mathbf{y}'$ to

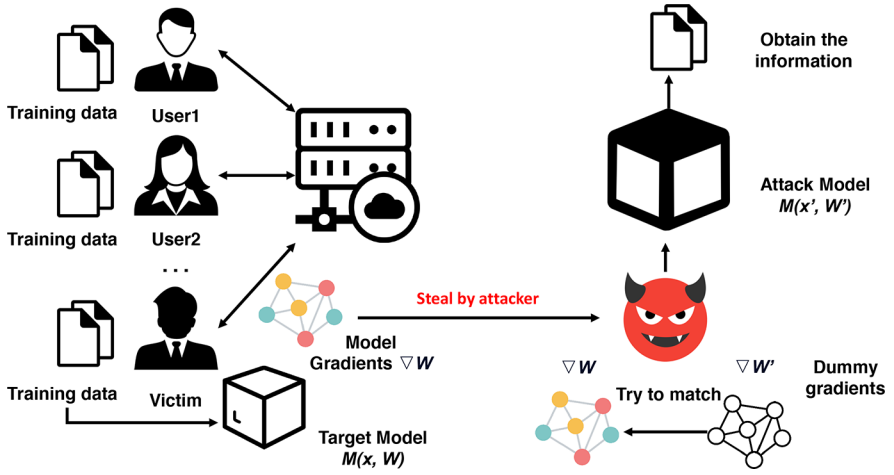

Fig. 6 Gradient attack in federated learning setting. (Zhu et al. 2019)

minimize the distance $||\nabla W' - \nabla W||^2$ between the dummy gradients $\nabla W'$ and the real gradients ∇W shared during federated learning. They achieved the training data by aiming to minimize the subsequent goal:

$$\begin{aligned} \mathbf{x}'^*, y'^* &= argmin_{\mathbf{x}',y'} ||\nabla W' - \nabla W||^2 \\ &= argmin_{\mathbf{x}',y'} ||\frac{\partial \ell(M(\mathbf{x}', W), y')}{\partial W} - \nabla W||^2. \end{aligned} \tag{17}$$

Although the DLG method is effective, it struggles with convergence and consistently discovering true labels. Zhao et al. (2020) introduce an improved method (iDLG) that can extract true labels from shared gradients. Unlike DLG, iDLG ensures the extraction of true labels, thereby enabling more effective data extraction. Moreover, the use of the Euclidean distance in DLG may lead to suboptimal recovery of the ground truth data, especially during the initial stages of the attack, because it tends to focus on large gradients while ignoring the majority of gradients that are near zero. This can be problematic in scenarios where the model weights are initialized normally, as a significant portion of the gradients will be clustered around zero, potentially leading to important information being overlooked. To deal with this matter, Deng et al. (2021) present a novel distance function: $\mathcal{D}(\nabla \mathbf{W}', \nabla \mathbf{W}) = ||\nabla \mathbf{W}' - \nabla \mathbf{W}||_2 + \alpha(\nabla \mathbf{W})||\nabla \mathbf{W}' - \nabla \mathbf{W}||$ that combines the L2 norm (Euclidean distance) and L1 norm (Manhattan distance) which helps to ensure that even smaller gradients contribute to the recovery process. Shortly after, Balunovic et al. (2022) present a method named LAMP, which employs an auxiliary language model to guide the reconstruction process toward more natural and linguistically plausible text, which helps generate text that is more coherent and readable. In addition, LAMP alternates between continuous optimization (using methods like gradient descent to optimize embedding vectors) and discrete optimization (applying text transformation operations to adjust word order) to enhance the accuracy of reconstructed text. This alternating optimization strategy helps avoid local minima and improves the sequence and structure of the reconstructed text. Furthermore, Gupta et al. (2022) first demonstrate the feasibility of text recovery from massive batches up to 128 phrases in length with their approach FILM. Unlike DLG and TAG, which have been optimized to align gradients directly, FILM takes a different strategy by first determining a group of words from gradients and then reconstructing sentences using a prior-based reordering technique and beam search. This method is more suited for discrete text inputs and is less sensitive to initialization. Additionally, most of these works assume that the server implements the federated learning protocol With faithfulness. In 2023, Fowl et al. (2022) provided a brand-new attack that uses malicious parameter vectors to expose private user content. This attack is successful even with mini-batches, several users, and lengthy sequences. In contrast to other attacks, it takes advantage of features of the token embedding and the Transformer architecture, extracting the tokens and positional embeddings independently in order to recover high-fidelity text. Table.5 shows previous work for privacy issues arising from AI generative models in distributed learning systems.

Table 5 Previous work for privacy issues arising from AI generative models in distributed learning systems

GAIM	Target Model	References	Knowl.*	Dataset
LM	BERT	**NeurIPS [2019]** Zhu et al. (2019)	□	MNIST, CIFAR-100, SVHN, LFW
	BERTs	**EMNLP [2021]** Deng et al. (2021)	□	CoLA, SST-2, RTE
	BERTs	**NeurIPS [2022]** Balunovic et al. (2022)	□	CoLA, SST-2, RottenTomatoes
	GPT-2	**NeurIPS [2022]** Gupta et al. (2022)	□	WikiText-103, Enron Email
	BERT-base, GPT-2	**ICLR [2022]** Fowl et al. (2022)	□	wikitext, Shakespeare and stackoverflow
VGM	GAN	**CCS [2017]** Hitaj et al. (2017)	□	MNIST, AT&T
	GAN	**JSAC [2020]** Song et al. (2020a)	□	MNIST, AT&T
	GAN	**TIST [2022]** Ren et al. (2022)	□	MNIST, CIFAR-100, LFW, VGG-Face
	Vision Transformers	**CVPR [2022]** Lu et al. (2022)	□	MNIST, CIFAR-10, ImageNet

* This column is the adversarial knowledge of different attacks. □: white-box. ■: black-box. ■ : gray-box

4.3.2 Defense techniques for language models in distributed learning setting

To defend language models in distributed learning settings from privacy threats, the most common protection measure is adding noise to gradients prior to sharing and applying differential privacy (Zhu et al. 2019; Balunovic et al. 2022; Gupta et al. 2022; Fowl et al. 2022). For instance, when the variance of noise exceeds 0.01, noise begins to impact the accuracy of DLG (Zhu et al. 2019), TAG (Deng et al. 2021), and LAMP (Balunovic et al. 2022), LAMP (Balunovic et al. 2022) demonstrates the best performance among them. However, even with the addition of noise, attackers can still potentially recover information from the gradients. Therefore, Zhu et al. (2019) have proposed gradient clipping as a more effective defense strategy. It involves setting small gradients to zero to reduce information leakage. By truncating or capping the gradients at a certain threshold, the model's sensitivity to individual training samples is reduced, thereby enhancing privacy protection. However, the maximum tolerable level of sparsity is approximately 20%. Beyond this threshold, the pruned images become visually unrecognizable. Moreover, Gupta et al. (2022) introduce another method called Freezing Word Embeddings to prevent attackers from extracting word information from the gradients of word embeddings. Before training on private data, the word embedding matrix's parameters are set as non-trainable, ensuring that during the training process, the weights within the matrix will not be updated through backpropagation. While the word embedding matrix remains frozen, the hidden layers of a Transformer

model will continue to be trained and updated based on the private data. Consequently, the gradients associated with these embeddings will not be computed or transmitted to the server, thwarting any attempt by an attacker to recover useful word information from the gradient information. Nevertheless, it's crucial to acknowledge that freezing word embeddings may restrict the model's capacity to learn vocabulary patterns specific to private data.

4.3.3 Privacy concerns in vision generative models for distributed learning systems

Moreover, GANs have been explored to target deep learning models in distributed learning systems. GANs have the ability to produce harmful data samples and tamper with model updates, resulting in the model's performance declining or compromising the privacy of the model. In 2017, Hitaj et al. (2017) proposed a novel method by using GANs to produce prototype samples specific to the target training set during the real-time learning process and these samples ought to come from the training data's similar distribution. Song et al. (2020a) combine GANs and multi-task discriminators to simultaneously classify the category, authenticity, and client identity. This novel client identity discrimination task permits the generator to retrieve the private information of users. Moreover, Ren et al. (2022) build a generative model that is optimized by minimizing the distance between the gradients produced by the two branches and the real gradients. This optimization process involves training the model using metrics such as Mean Squared Error $MSE(g, \hat{g})$, Wasserstein Distance $WD(g, \hat{g})$, and Total Variation Loss $TVLoss(\hat{\boldsymbol{x}})$, to evaluate the difference between the generated gradients and the real gradients. The loss function for this attack can be described as follows: $\hat{\mathcal{L}}(g, \hat{g}, \hat{\boldsymbol{x}}) = MSE(g, \hat{g}) + WD(g, \hat{g}) + \alpha \cdot TVLoss(\hat{\boldsymbol{x}})$, where $\boldsymbol{x}$ is the fake image generated by GAN, g and $\hat{g}$ are true gradient and fake gradient, respectively. Additionally, the smoothness regularization weighting parameter is denoted by α. More importantly, Lu et al. (2022) reveal that learnable positional embeddings could be a potential vulnerability for privacy leakage in visual transformers (Khan et al. 2023) such as the Vision Transformer by using the APRIL attack. This aspect has not received sufficient attention in previous research, but the APRIL attack demonstrates that positional embeddings can effectively recover input data, posing a risk of privacy leakage.

4.3.4 Defense techniques for vision generative models in distributed learning setting

Hitaj et al. (2017); Ren et al. (2022) are still attempting to apply DP and noise addition to safeguard privacy. In the work of Hitaj et al. (2017), the effectiveness of DP is demonstrated in its ability to provide a certain level of privacy protection for data labels that are actually used during the training phase, preventing the recovery of specific elements associated with these labels. However, its limitation lies in its inability to effectively defend against active attacks using GANs, which can bypass the privacy protection offered by record-level DP and leak sensitive information from the training data. Moreover, Ren et al. (2022) use noise addition in their method GRNN, which is capable of successfully recovering image data and achieving satisfactory results when the noise level is reduced to 0.01. However, as the noise level increases, it fails to recover the image. In addition to these two widely used privacy protection techniques, Song et al. (2020a) discuss the use of encryption techniques and isolated environments to protect models from privacy attacks in the context of distributed learning scenarios, specifically in federated learning. First, Secure Aggregation (SA):

clients encrypt their model updates prior to submitting them to the server. The server can compute this encrypted data without having to decrypt the data, thus safeguarding privacy. Second, Homomorphic Encryption (HE): this encryption method enables the server to process encrypted data and output specific calculations. Upon decryption, they correspond with the identical procedures carried out on the original data. This indicates that even while the training procedure keeps the data encrypted, the server can still perform useful computations. Third, Trusted Execution Environment (TEE): a TEE provides an isolated execution environment that ensures the confidentiality and integrity of the code and data loaded into it. Within the framework of federated learning, training may take place within a TEE, preventing even a compromised server from accessing intermediate values such as model parameter updates. These methods can effectively prevent a malicious server from inferring private data information from the model updates. However, Song et al. (2020a) also present challenges, such as increased computational costs and difficulties in detecting malicious updates.

Besides, in Vision Transformer models, position embeddings are used to provide positional information to the image patches since the Transformer architecture does not inherently capture spatial relationships. These embeddings are typically learned during the model training process. Consequently, learnable position embeddings can be a vulnerability to privacy breaches. To enhance the privacy protection of visual Transformer models in distributed learning environments, Lu et al. (2022) propose a defense strategy named learnable position embedding which can make position embedding fixed. This strategy initializes the position embeddings before training and then does not optimize them, for example, not calculating their gradients during the training process. As a defense measure, fixed position embeddings are practical because they do not add extra computational burden and do not significantly alter the model's architecture. Moreover, this method can be easily integrated into existing federated learning frameworks. However, applying this may impact the model's performance since these embeddings cannot be adapted to the specific data distribution of the task at hand.

4.4 Differential privacy

Differential Privacy (Dwork 2006) is a framework for measuring and managing the privacy risks associated with disclosing personal data about specific persons within a dataset. It makes sure that a person's data is either included in or excluded from a dataset and doesn't materially alter the probability of any outcome when that dataset is used to compute statistics or machine learning models. It does this by incorporating a predetermined level of noise into the data or the data analysis's output. The noise is calibrated to preserve the overall value of the data while maintaining individual privacy. It is extremely useful when handling big datasets, such as those used to train large language models, which can inadvertently expose sensitive information about the individuals who contributed to the data. For example, by using DP, Apple collects anonymized user data to enhance features such as emoji suggestions, QuickType, and other services while ensuring that individual privacy is maintained.

When discussing differential privacy, a mechanism M is a function that takes as input a dataset d and outputs some information or a summary of that dataset. If the technique satisfies the following requirements for every nearby dataset d and d' and for every potential

set of outputs Y, it is said to fulfill differential privacy (Dwork 2006). Formally described as follows:

$$\frac{Pr(M(d) \in Y)}{Pr(M(d') \in Y)} \leq e^{\epsilon}. \tag{18}$$

The ϵ parameter in differential privacy is often referred to as the privacy budget or privacy parameter. It acts as a tuning knob for a differentially private mechanism's degree of privacy protection.

The Laplace mechanism is one of the fundamental mechanisms in DP. It is used to privatize real-valued queries, such as sums, averages, or counts, over a dataset. The Laplace mechanism is particularly useful because it provides a straightforward way to add noise that fulfills differential privacy. The Laplace mechanism is a method for adding noise to a real-valued function $m(d)$ in a way that ensures ϵ-differential privacy. Here's how the Laplace mechanism defines a differentially private mechanism $M(d)$ for a function $m(d)$ that returns a number:

$$M(d) = m(d) + Lap\left(\frac{s}{\epsilon}\right). \tag{19}$$

The sensitivity s of m, which is the maximum absolute difference in $m(d)$ over any two adjacent datasets d and d'. $Lap(\frac{s}{\epsilon})$ represents the output of an arbitrary variable drawn from the Laplace distribution with location parameter (center) 0 and scale parameter $\frac{s}{\epsilon}$.

Approximate differential privacy is a variant of differential privacy that introduces an additional parameter δ, allowing for slightly less stringent privacy protection with a certain probability δ. Specifically, a randomized algorithm F fulfills (ϵ, δ)-differential privacy if for any two nearby databases d and d', and any possible subset of outputs Y, the following holds:

$$Pr(M(d) \in Y) \leq e^{\epsilon} Pr(M(d') \in Y) + \delta. \tag{20}$$

Here, ϵ represents the strength of privacy protection, and δ represents the probability with which the algorithm can violate differential privacy protection. When $\delta = 0$, we say that the algorithm satisfies ϵ-differential privacy, which is a stricter standard of privacy protection. Table.6 shows previous work for differential privacy protects generative AI models.

4.4.1 Differential privacy for language models

As mentioned in Carlini et al. (2019), during the training process, neural networks may encounter the issue of unintended memorization, where the model unintentionally memorizes occasional or unique sequences that appear in the training data. This can become a privacy concern when the model is trained on sensitive data such as private message texts. They implement the Differentially-Private Stochastic Gradient Descent (DP-SGD) (Abadi et al. 2016) algorithm by scaling down individual training example gradients to a predefined maximum norm and adding Gaussian noise to test its efficacy in preventing unintentional memorization by neural networks. Based on the framework of DP-SGD (Abadi et al. 2016),

Table 6 Previous work for differential privacy protects generative AI models

GAIM	Target Model	Reference	Optimization	Dataset
LM	Smart Compose	**USENIX Security [2019]** (Carlini et al. 2019)	DP-SGD+RMSProp	PTB, WikiText-103
	BERT	**EMNLP [2021]** (Hoory et al. 2021)	DP-SGD+RMSProp	MIMIC-III, Wikipedia, BookCorpus
	BERT-Base	**ICML [2021]** (Yu et al. 2021)	RGP	MNLI, SST-2, QQP, QNLI
	BERT-Large	**EMNLP [2021]** (Anil et al. 2022)	DP-SGD+Adam	Wikipedia, BookCorpus
	BERT, RoBERTa, GPT-2	**ICLR [2021]** (Li et al. 2022)	DP-SGD+Adam	MNLI, SST-2, QQP, E2E, DART, Persona-Chat
	RoBERTa-Base/Large, GPT-2	**ICLR [2021]** (Yu et al. 2022)	DP-SGD+AdamW	MNLI, SST-2, QQP, QNLI
	BERT, 1/2BERT, DistilBERT	**NeurIPS [2022]** (Mireshghallah et al. 2022a)	DP-SGD	MNLI, SST-2, QQP, QNLI
	XLM-R	**ICML [2023]** (Rust and Søgaard 2023)	DP-SGD+AdamW	XNLI
VGM	RBM, VAE	**TKDE [2018]** (Acs et al. 2018)	DP-SGD+Adam	MNIST, CDR, TRANSIT
	GAN	**ARXIV [2018]** (Zhang et al. 2018)	DP-SGD+Adam	MNIST, LSUN-U, LSUN-L, and CelebA
	WGAN	**ARXIV [2018]** (Xie et al. 2018)	DP-SGD+RMSProp	MNIST and MIMIC-III
	GAN	**ICLR [2018]** (Jordon et al. 2018)	DP-SGD+Adam	Credit card fraud detection dataset
	CGAN	**CVPRW [2019]** (Torkzadehmahani et al. 2019)	DP-SGD+Adam	MNIST
	WGAN	**ICPADS [2019]** (Liu et al. 2019c)	DP-SGD	MNIST and MIMIC-III
	GAN	**ICLR [2019]** (Augenstein et al. 2020)	FedAvg	EMNIST
	GAN	**TIFS [2019]** (Xu et al. 2019)	DP-SGD	MNIST, LSUN, CelebA
	GAN	**IJCAIW [2019]** (Triastcyn and Faltings 2019)	FedAvg	MNIST, CelebA
	WGAN	**NeurIPS [2020]** (Chen et al. 2020a)	DP-SGD	MNIST, Fashion-MNIST

Hoory et al. (2021) successfully train the first differentially private BERT model, which provides a robust privacy guarantee while preserving a high standard of performance in downstream tasks. Additionally, Yu et al. (2021) introduce a novel approach called Reparametrized Gradient Perturbation (RGP) to train BERT on many downstream tasks. After the completion of the backward propagation phase, RGP first clips the gradients associated with the matrices L and R. This clipping action limits the gradient's magnitude. Subsequently, RGP introduces noise into these already-clipped gradients and constructs an update for the original weight matrix. To focus on the high accuracy baseline for DP BERT pertaining, Anil et al. (2022); Li et al. (2022); Rust and Søgaard (2023) used DP-SGD (Abadi et al. 2016) with Adam optimizer (Kingma and Ba 2015), which performs hyper-parameter tun-

ing for Adam, specifically at one batch size and subsequently applies the optimized hyperparameters to all other batch sizes. Also, highlighting the baseline of fine-tuning, Yu et al. (2022) use the additive fine-tuning scheme LoRA (Hu et al. 2022a), which can be seen as a condensed form of RGP (Yu et al. 2021). In LoRA (Hu et al. 2022a), the weight matrix W_{FT} is reparametrized as the sum of the pre-trained weight matrix W_{PT} and a learnable reparametrization LR. Notably, during the training process, the pre-trained weight matrix W_{PT} remains frozen.

4.4.2 Differential privacy for vision generative models

Differential privacy is a versatile framework that extends beyond the protection of language models to various domains where privacy is a concern, including vision generative models. For example, GANs are used for creating images and are typically trained on large datasets of images. DP can be applied during the training process to make sure the model doesn't retain or divulge details about specific training samples. More importantly, DP techniques can be employed to add noise to the gradients or outputs. In scenarios where vision generative models are trained across multiple decentralized devices, DP can be crucial in safeguarding user data throughout the model update process.

To deal with this matter, Acs et al. (2018) first propose a new method for privately publishing high-dimensional datasets and generative models. The differentially private k-means algorithm is used to divide the whole training dataset into k-disjoint sub-datasets in the suggested strategy. Next, every sub-dataset is utilized to train a different set of generative models independently. In the work of Zhang et al. (2018), they present a dp-GAN framework. Rather than simply cleaning up and distributing data, the data curator unveils a profound generative model, meticulously trained in a differentially private way using the initial data. With the help of this intricate generative model, the analyst gains the ability to generate an infinite array of fake data, tailored for diverse analysis tasks. Different from Zhang et al. (2018), and Xie et al. (2018), Liu et al. (2019c) propose another dp-GAN framework based on WGAN (Arjovsky et al. 2017a), the key idea of their work is incorporating noise into gradients throughout the process of learning. WGAN (Arjovsky et al. 2017a) offers clear advantages over traditional GAN in terms of convergence, sample quality, and gradient stability by introducing the Wasserstein distance as the training objective. In the stage of training, they add designed noise to the gradient of the Wasserstein distance, and when updating the parameters of models, gradient clipping is a method in these works to prevent gradients from becoming too large and ensure that the weight updates are within the preset range. In order to maintain the discriminator with stronger differential privacy, Jordon et al. (2018) substitute the discriminator in the GAN framework with a Private Aggregation of Teacher Ensembles (PATE) mechanism. This entails the incorporation of k teacher-discriminators alongside a student discriminator, through its ability to tightly limit the influence of individual samples on the model's output. This implies that less noise can be added per sample while still meeting differential privacy constraints, thus improving the quality of synthetic data. Similarly, Xu et al. (2019) also focus on adding noise into the gradient of the discriminator. When training the discriminator, gradients are first computed, which indicates how the model parameters should be updated to minimize the loss function. The computed gradients are then subjected to a pruning operation that caps the gradient magnitude within a preset range. This range is defined by a hyperparameter to ensure that the sensitivity of the

gradients is bounded. With the combination of Gradient Pruning and Noise Injection, this framework is capable of providing privacy protection when generating synthetic data, while the generated data still maintains sufficient quality and utility for various analytical tasks.

The conventional approach employed by these recent investigations to ensure DP involves initially constraining the $L2$ norm of the gradients of the combined loss of the discriminator on synthetic and real data, followed by the introduction of Gaussian noise to the clipped gradients. One of the constraints of these recent endeavors is their exclusive emphasis on producing fake data, such as images without corresponding labels. However, Torkzadehmahani et al. (2019) proposed a DP-CGAN framework. For each set of actual data and fake data, DP-CGAN clips the discriminator's loss gradients independently. This makes it possible to precisely adjust how sensitive the model is to private and genuine data. More importantly, it is capable of producing both the relevant labels and synthetic data. As mentioned in Sect. 4.3, in the realm of distributed GANs, it is imperative to protect the privacy of generators in a federated setting, as the mere segregation of data at a physical level falls short of guaranteeing adequate protection. Augenstein et al. (2020) and Triastcyn and Faltings (2019) primarily focus on federated generative privacy. They employ DP-FedAvg, an algorithm that combines DP with federated learning. This algorithm uses clipping and adds Gaussian noise to achieve user-level privacy protection. Furthermore, in the work of Chen et al. (2020a), they introduce the GS-WGAN model, which focuses on publicly releasing only the generator's parameters, discarding the discriminator's parameters post-training, to minimize privacy risks. GS-WGAN achieves differential privacy by precisely distorting gradient information in the training process, enabling more significant gradient updates and ensuring the training of deeper models to generate richer samples.

5 Challenges and open problems

5.1 Memorization of generative AI models

Model memorization, where machine learning models retain specific details from their training data, poses a significant threat to data privacy. This retention can lead to the unintended disclosure of sensitive information, undermining trust and security, particularly in environments handling private or proprietary data. Carlini et al. (2019) have shown that neural networks may inadvertently memorize specific sequences from the training data, which can include sensitive information such as personal identifiers or private messages. This memorization can lead to the model generating or revealing sensitive information when prompted with certain inputs. Additionally, as the model encounters a growing number of classification categories, it would extract increasingly more features from the data to ensure high classification accuracy. Consequently, models with a larger number of output classes are required to retain more detailed information from their training datasets, which in turn can result in greater information leakage. In this way, attackers are more likely to perform membership inference attacks (Shokri et al. 2017).

5.1.1 Memorization of language models

Moreover, models may also memorize patterns associated with user attributes, leading to attribute inference attacks, as mentioned by Thomas et al. (2020). They have compared memorization issues in GloVe (Pennington et al. 2014), ELMo (Peters et al. 2018), and BERT (Devlin et al. 2019). It was found that GloVe (Pennington et al. 2014) is more prone to memorizing sensitive information and reaching maximum exposure levels earlier in training compared to other models. ELMo (Peters et al. 2018) and BERT (Devlin et al. 2019) embeddings also exhibit memorization, with higher-dimensional embeddings being more susceptible to retaining sensitive data. Furthermore, the presence of multiple instances of sensitive information in the training data seems to reduce memorization, potentially confusing the model. To delve deeper into the issue of memorization of LLM, Carlini et al. (2023b) conduct experiments from three perspectives: model scale, data duplication, and context. First, within the same model family, larger models tend to memorize 2-5 times more information than their smaller counterparts. Second, data examples that are repeated more frequently have a higher likelihood of being extractable. Third, it is significantly easier to extract sequences when a longer context is provided.

To mitigate these risks, several methods are employed. For example, DP, which we mentioned several times before. As noted by Carlini et al. (2023b), reducing the duplication in the dataset can lower the extent to which a model memorizes the training data.

5.1.2 Memorization of vision generative models

To clarify the impact of model memorization in generative models, van den Burg and Williams (2021) propose a method for quantifying memorization, which is used to assess whether the VGM has remembered specific samples from the training dataset. The mechanism behind memorization primarily arises from the VGM's overfitting to certain training samples during the training process, especially in sparse data regions or when the VGM assigns excessively high weights to certain samples. Additionally, memorization may also be related to the size of the VGM, the repetition of training data, and the VGM's ability to fit specific regions of the input space. Moreover, memorization can occur at an early stage of the training process and is associated with the VGM's local probability density estimation.

To mitigate the impact of model memorization, van den Burg and Williams (2021) explore various strategies, First, implement DP to introduce randomness and reduce overmemorization of individual training samples. Second, adjust model architectures to handle outliers. Third, preprocess data to increase diversity and reduce redundancy. Additionally, modify training strategies such as learning rates, early stopping, and regularization to decrease reliance on specific samples.

5.2 Generative AI model architectures impact its privacy

Zhang et al. (2024) conduct a comprehensive study on the impact of deep learning model architectures, specifically comparing CNNs and Transformers, on their vulnerability to privacy attacks. CNNs rely on local convolution operations, using a sliding window to extract features from input data. While this localized receptive field effectively captures patterns, it also makes CNNs susceptible to privacy leakage. Certain micro-design elements, such as

activation layers and batch normalization, can enable attackers to recover sensitive information. When CNNs overfit, attackers can exploit membership or attribute inference attacks to extract private data from the training set.

Transformers, on the other hand, use multi-head self-attention mechanisms, which provide a much broader receptive field than CNNs. Instead of focusing on local features, Transformers process the entire input sequence, allowing them to capture more detailed and sensitive information. The attention modules enable the model to learn global patterns, increasing the risk of exposing private data. Additionally, design elements like layer normalization and stem layers further heighten privacy risks. Research (Zhang et al. 2024) shows that, even with similar levels of overfitting, Transformer-based models are more vulnerable to privacy attacks than CNN-based models.

5.3 Possible attack generative AI models

Research into privacy attacks on language models like BERT and GPT-2 (Jagannatha et al. 2021; Carlini et al. 2021; Mireshghallah et al. 2022b; Mattern et al. 2023; Jagielski et al. 2024) is well-documented, as is the study of privacy attacks on visual models such as GANs and Diffusion models (Hayes et al. 2019; Liu et al. 2019a; Hilprecht et al. 2019; Chen et al. 2020b; Zhou et al. 2022; Shafran et al. 2021; Duan et al. 2023; Matsumoto et al. 2023; Carlini et al. 2023a; Kong et al. 2024). However, there is a relative lack of research specifically focused on multi-modal models that process both textual and visual data (Hu et al. 2022c; Wu et al. 2022; Ko et al. 2023). This field is gradually attracting attention as researchers recognize the significance of understanding privacy vulnerabilities and potential attack vectors in these complex systems.

Multi-modal models, which integrate NLP and CV, are being widely deployed across various domains (Hu et al. 2022c; Wu et al. 2022; Ko et al. 2023). The unique challenge with privacy attacks on multi-modal models is the intricate interplay between different modalities, which adversaries can exploit to infer sensitive information or launch sophisticated attacks. For example, an attacker might use the relationship between text captions and images to deduce private details or infer sensitive attributes not explicitly disclosed.

5.4 Jailbreaking privacy attacks on language models

Currently, there is a growing body of research that targets jailbreaking attacks on models transitioning from large language models to multi-modal models (Huang et al. 2022; Li et al. 2023; Deng et al. 2024; Nasr et al. 2023). However, the majority of these studies concentrate on security issues such as adversarial examples (Liu et al. 2024a), and backdoor attacks (Li et al. 2021), which aim to compromise the model's decision-making process or extract sensitive information from the model itself. In comparison, privacy-focused attacks that specifically address the protection of user data and the prevention of unauthorized use of personal information are less common in the literature. This discrepancy suggests a need for more research that emphasizes privacy protection in the context of generative AI models, where the risks of data misuse and privacy intrusions can be particularly high due to the models' ability to process and generate diverse types of sensitive content.

5.5 Enhanced membership privacy

To enhance membership privacy, Wen et al. (2024) introduce a backdoor into a pre-trained model to augment membership inference attacks. In this approach, an adversary poisons the model by altering its weights to create a backdoor. When a victim fine-tunes this compromised model using their private dataset, it leaks the fine-tuning data at a much higher rate than a regular model. This poisoning generates a differential loss pattern, making membership inference attacks more effective and enabling the adversary to identify specific data points used in the fine-tuning process. Additionally, Bertran et al. (2023) improve membership inference attacks by using quantile regression, which has a computational advantage over traditional shadow model-based methods. It requires training only one model and operates without knowledge of the target model's architecture, enabling a true "black-box" attack.

5.6 Enhanced model inversion attack

In Sect. 4.2.4, we delved into the necessity of the generator in model inversion attack. However, MIA still confronts challenges that are GAIM-agnostic. Currently, the focus in this domain has shifted towards Label-only MIA (Kahla et al. 2022; Nguyen et al. 2024; Liu et al. 2024b). A promising attack paradigm involves training a joint distribution $p_\theta(\boldsymbol{x}_{pub}, c)$ on public data through knowledge transfer and vision generative models, utilizing pseudo-labels obtained from extensive interactions with the model. Sampling is then conducted through $p_\theta(\boldsymbol{x} \mid c)p(c)$ (Nguyen et al. 2024; Tian et al. 2023; Liu et al. 2024b). This approach circumvents the iterative optimization of samples, thereby mitigating the risk of generating adversarial examples. Nevertheless, these methods typically require a large number of queries to characterize the decision boundary of the target model. The query process can be optimized by incorporating techniques such as active learning, reinforcement learning, and evolutionary algorithms (Oliynyk et al. 2023).

More pragmatic approaches, such as BREPMI (Kahla et al. 2022), leverage the geometric properties of decision boundaries. These methodologies progressively shift the initial sample towards the centroid of the target classification distribution through zeroth-order gradient like optimization within the latent space, which requires fewer queries for specific classifications. To the best of our knowledge, there is currently a scarcity of attack methods that incorporate the geometric boundary properties of neural networks or substitute gradient optimization in this context. Conversely, numerous attack methods that combine above properties have been devised for the generation of adversarial examples in the label-only black-box scenario (Brendel et al. 2018; Chen et al. 2020c; Maho et al. 2021; Fu et al. 2024; Cheng et al. 2018).

5.7 Advanced differential privacy mechanisms

To enhance DP, it is crucial for existing algorithms to consider scalability and computation (Jia et al. 2023; Wang et al. 2022, 2019). This requires developing new mathematical methods for noise injection that protect privacy while maintaining accuracy. Cummings et al. (2024) have reviewed and proposed improvements to privacy infrastructure, trade-offs, and practical auditing. They emphasize the need for clear communication of DP guarantees and

integration with broader privacy practices. Future DP research should focus on eliminating hyperparameters, setting benchmarks, incorporating user feedback, improving usability, and advancing theory for DP's practical use in various contexts.

6 Conclusion

This work presents an in-depth systematic review of the privacy concerns surrounding generative AI models, addressing a wide range of privacy vulnerabilities, including membership privacy, model inversion attacks, privacy in distributed learning systems, and differential privacy. For the different purposes of attack and defense, these approaches formulate problems based on various generative AI models, language models, vision generative models, and multi-modal models. Subsequent to an extensive analysis, the remaining challenges and open problems are presented for further discussion, focusing on GAIM's memorization issues, architecture, possible attack GAIM, as well as other advanced attacks and defense techniques. Our goal is to create a targeted resource that encourages further research in this critical area.

Author Contributions Yihao Liu was responsible for drafting the main text and creating the figures of the manuscript. Jinhe Huang contributed to the manuscript by writing two subsections of the text and generating some of the figures. Yanjie Li undertook the role of reviewing and revising the sections pertaining to privacy within the manuscript. Dong Wang focused on the AI components of the manuscript, providing critical reviews and revisions. Bin Xiao conceptualized the overall framework of the paper and was involved in the comprehensive review and revising of the entire manuscript.

Funding This work was supported in part by HK RGC GRF under Grants PolyU 15201323 and PolyU 15226224.

Data Availibility No datasets were generated or analysed during the current study.

Declarations

Conflict of interest The authors have no relevant financial or non-financial interests to disclose.

References

Abadi M, Chu A, Goodfellow I, et al (2016) Deep learning with differential privacy. In: Proceedings of CCS, pp 308–318

Achiam J, Adler S, Agarwal S, et al (2023) Gpt-4 technical report. arXiv preprint arXiv:2303.08774

Acs G, Melis L, Castelluccia C et al (2018) Differentially private mixture of generative neural networks. IEEE Trans Knowl Data Eng 31(6):1109–1121

Aïvodji U, Gambs S, Ther T (2019) Gamin: An adversarial approach to black-box model inversion. arXiv preprint arXiv:1909.11835
Anil R, Ghazi B, Gupta V, et al (2022) Large-scale differentially private bert. In: Proceedings of EMNLP, pp 6481–6491
An S, Tao G, Xu Q, et al (2022) Mirror: Model inversion for deep learning network with high fidelity. In: Proc. of NDSS
Arjovsky M, Chintala S, Bottou L (2017a) Wasserstein gan. In: Proc. of ICML
Arjovsky M, Chintala S, Bottou L (2017b) Wasserstein generative adversarial networks. In: Proc. of ICML, PMLR, pp 214–223
Arora S, Ge R, Liang Y, et al (2017) Generalization and equilibrium in generative adversarial nets (gans). In: International conference on machine learning, PMLR, pp 224–232
Augenstein S, McMahan HB, Ramage D, et al (2020) Generative models for effective ml on private, decentralized datasets. In: Proc. of ICLR
Augustin M, Boreiko V, Croce F et al (2022) Diffusion visual counterfactual explanations. Adv Neural Inf Process Syst 35:364–377
Avrahami O, Lischinski D, Fried O (2022) Blended diffusion for text-driven editing of natural images. In: Proc. of CVPR, pp 18,208–18,218
Bae H, Jang J, Jung D, et al (2018) Security and privacy issues in deep learning. arXiv preprint arXiv:1807.11655
Balunovic M, Dimitrov D, Jovanović N, et al (2022) Lamp: Extracting text from gradients with language model priors. In: Proc. of NeurIPS, pp 7641–7654
Bertran M, Tang S, Roth A, et al (2023) Scalable membership inference attacks via quantile regression. In: Proc. of NeurIPS, pp 314–330
Boulemtafes A, Derhab A, Challal Y (2020) A review of privacy-preserving techniques for deep learning. Neurocomputing 384:21–45
Brendel W, Rauber J, Bethge M (2018) Decision-based adversarial attacks: reliable attacks against black-box machine learning models. In: International conference on learning representations
Brock A, Donahue J, Simonyan K (2018) Large scale gan training for high fidelity natural image synthesis. In: Proc. of ICLR
Brown H, Lee K, Mireshghallah F, et al (2022) What does it mean for a language model to preserve privacy? In: FAccT '22, pp 2280–2292
Brown T, Mann B, Ryder N, et al (2020) Language models are few-shot learners. In: Proc. of NeurIPS, pp 1877–1901
Cai Z, Xiong Z, Xu H et al (2021) Generative adversarial networks: a survey toward private and secure applications. ACM Comput Surv 54(6):1–38
Carlini N, Liu C, Erlingsson Ú et al (2019) The secret sharer: evaluating and testing unintended memorization in neural networks. USENIX Secur 19:267–284
Carlini N, Tramer F, Wallace E et al (2021) Extracting training data from large language models. USENIX Secur 21:2633–2650
Carlini N, Hayes J, Nasr M et al (2023) Extracting training data from diffusion models. USENIX Secur 23:5253–5270
Carlini N, Ippolito D, Jagielski M, et al (2023b) Quantifying memorization across neural language models. In: Proc. of ICLR
Cheng M, Le T, Chen PY, et al (2018) Query-efficient hard-label black-box attack: an optimization-based approach. arXiv preprint arXiv:1807.04457
Chen J, Jordan MI, Wainwright MJ (2020c) Hopskipjumpattack: A query-efficient decision-based attack. In: 2020 ieee symposium on security and privacy (sp), IEEE, pp 1277–1294
Chen S, Kahla M, Jia R, et al (2021) Knowledge-enriched distributional model inversion attacks. In: Proc. of ICCV, pp 16,178–16,187
Chen Y, Liu Y, Dong L, et al (2022) Adaprompt: Adaptive model training for prompt-based nlp. In: Proc. of EMNLP, pp 6057–6068
Chen D, Orekondy T, Fritz M (2020a) Gs-wgan: A gradient-sanitized approach for learning differentially private generators. In: Proc. of NeurIPS, pp 12,673–12,684
Chen D, Yu N, Zhang Y, et al (2020b) Gan-leaks: A taxonomy of membership inference attacks against generative models. In: Proc. of CCS, pp 343–362
Conneau A, Khandelwal K, Goyal N, et al (2020) Unsupervised cross-lingual representation learning at scale. In: Proc. of ACL, pp 8440–8451
Cristofaro ED (2020) An overview of privacy in machine learning
Cummings R, Desfontaines D, Evans D, et al (2024) Advancing differential privacy: where we are now and future directions for real-world deployment. Harvard data science review

Deng G, Liu Y, Li Y, et al (2024) Masterkey: Automated jailbreaking of large language model chatbots. In: Proc. of NDSS, NDSS 2024
Deng J, Wang Y, Li J, et al (2021) Tag: Gradient attack on transformer-based language models. In: Proc. of EMNLP, pp 3600–3610
Devlin J, Chang MW, Lee K, et al (2019) Bert: Pre-training of deep bidirectional transformers for language understanding. In: Proc. of NAACL
Dhariwal P, Nichol A (2021) Diffusion models beat gans on image synthesis. In: Proc. of NeurIPS, pp 8780–8794
Dhurandhar A, Chen PY, Luss R, et al (2018) Explanations based on the missing: Towards contrastive explanations with pertinent negatives. Advances in neural information processing systems 31
Dinh L, Krueger D, Bengio Y (2014) Nice: Non-linear independent components estimation. arXiv preprint arXiv:1410.8516
Dinh L, Sohl-Dickstein J, Bengio S (2016) Density estimation using real nvp. arXiv preprint arXiv:1605.08803
Duan J, Kong F, Wang S, et al (2023) Are diffusion models vulnerable to membership inference attacks? In: Proc. of ICML
Dwork C (2006) Differential privacy. In: Proc. of ICALP, pp 1–12
Fowl L, Geiping J, Reich S, et al (2022) Decepticons: Corrupted transformers breach privacy in federated learning for language models. In: Proc. of ICLR
Fredrikson M, Lantz E, Jha S et al (2014) Privacy in pharmacogenetics: an { End-to-End } case study of personalized warfarin dosing. USENIX Secur 14:17–32
Fredrikson M, Jha S, Ristenpart T (2015) Model inversion attacks that exploit confidence information and basic countermeasures. In: Proc. of CCS, pp 1322–1333
Fu J, Ling X, Qian Y, et al (2024) Towards query-efficient decision-based adversarial attacks through frequency domain. In: 2024 IEEE International Conference on Multimedia and Expo (ICME), IEEE, pp 1–6
Gal R, Alaluf Y, Atzmon Y, et al (2022) An image is worth one word: personalizing text-to-image generation using textual inversion. In: Proc. of ICLR
Ganju K, Wang Q, Yang W, et al (2018) Property inference attacks on fully connected neural networks using permutation invariant representations. In: Proc. of CCS, pp 619–633
Gao T, Fisch A, Chen D (2021) Making pre-trained language models better few-shot learners. In: Proc. of ACL/IJCNLP, pp 3816–3830
Golda A, Mekonen K, Pandey A, et al (2024) Privacy and security concerns in generative ai: a comprehensive survey. IEEE Access
Gomez AN, Ren M, Urtasun R, et al (2017) The reversible residual network: backpropagation without storing activations. In: Proc. of NeurIPS
Goodfellow I, Pouget-Abadie J, Mirza M, et al (2014) Generative adversarial nets. In: Proc. of NeurIPS
Gu Y, Han X, Liu Z, et al (2022) Ppt: Pre-trained prompt tuning for few-shot learning. In: Proc. of ACL, pp 8410–8423
Gulrajani I, Ahmed F, Arjovsky M, et al (2017) Improved training of wasserstein gans. In: Proc. of NeurIPS
Guo D, Rush AM, Kim Y (2021) Parameter-efficient transfer learning with diff pruning. In: Proc. of ACL/IJCNLP, pp 4884–4896
Gupta S, Huang Y, Zhong Z, et al (2022) Recovering private text in federated learning of language models
Han G, Choi J, Lee H, et al (2023) Reinforcement learning-based black-box model inversion attacks. In: Proc. of CVPR, pp 20,504–20,513
Hayes J, Melis L, Danezis G et al (2019) Logan: Membership inference attacks against generative models. PoPETs 1:133–152
Hilprecht B, Härterich M, Bernau D (2019) Monte Carlo and reconstruction membership inference attacks against generative models. PoPETs 4:232–249
Hisamoto S, Post M, Duh K (2020) Membership inference attacks on sequence-to-sequence models: Is my data in your machine translation system? Trans Assoc Comput Linguist 8:49–63
Hitaj B, Ateniese G, Perez-Cruz F (2017) Deep models under the gan: Information leakage from collaborative deep learning. In: Proc. of CCS, p 603-618
Hochreiter S, Schmidhuber J (1997) Long short-term memory. Neural Comput 9(8):1735–1780
Ho J, Jain A, Abbeel P (2020) Denoising diffusion probabilistic models. In: Proc. of NeurIPS, pp 6840–6851
Hoory S, Feder A, Tendler A, et al (2021) Learning and evaluating a differentially private pre-trained language model. In: Proc. of EMNLP, pp 1178–1189
Ho J, Salimans T (2021) Classifier-free diffusion guidance. In: NeurIPS 2021 workshop on deep generative models and downstream applications
Houlsby N, Giurgiu A, Jastrzebski S, et al (2019) Parameter-efficient transfer learning for nlp. In: Proc. of ICML, pp 2790–2799

Howard J, Ruder S (2018) Universal language model fine-tuning for text classification. In: Proc. of ACL, pp 328–339
Hu EJ, Shen Y, Wallis P, et al (2022a) Lora: Low-rank adaptation of large language models. In: Proc. of ICLR
Hu EJ, Wallis P, Allen-Zhu Z, et al (2021) Lora: Low-rank adaptation of large language models. In: Proc. of ICLR
Hu H, Salcic Z, Sun L et al (2022) Membership inference attacks on machine learning: a survey. ACM Comput Surv 54(11s):1–37
Hu K, Gong S, Zhang Q et al (2024) An overview of implementing security and privacy in federated learning. Artif Intell Rev 57(8):1–66
Huang X, Ruan W, Huang W et al (2024) A survey of safety and trustworthiness of large language models through the lens of verification and validation. Artif Intell Rev 57(7):175
Huang J, Shao H, Chang KCC (2022) Are large pre-trained language models leaking your personal information? In: Proc. of EMNLP, pp 2038–2047
Hu P, Wang Z, Sun R, et al (2022c) M^4i: Multi-modal models membership inference. In: Proc. of NeurIPS
Jacob P, Zablocki É, Ben-Younes H, et al (2022) Steex: steering counterfactual explanations with semantics. In: European Conference on Computer Vision, Springer, pp 387–403
Jagannatha A, Rawat BPS, Yu H (2021) Membership inference attack susceptibility of clinical language models. arXiv preprint arXiv:2104.08305
Jagielski M, Nasr M, Lee K, et al (2024) Students parrot their teachers: membership inference on model distillation. In: Proc. of NeurIPS
Jeanneret G, Simon L, Jurie F (2022) Diffusion models for counterfactual explanations. In: Proceedings of the Asian Conference on Computer Vision, pp 858–876
Jeanneret G, Simon L, Jurie F (2023) Adversarial counterfactual visual explanations. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 16,425–16,435
Jeanneret G, Simon L, Jurie F (2024) Text-to-image models for counterfactual explanations: a black-box approach. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, pp 4757–4767
Jia H, Rao H, Wen C et al (2023) Crayfish optimization algorithm. Artif Intell Rev 56(Suppl 2):1919–1979
Jochems A, Deist TM, El Naqa I et al (2017) Developing and validating a survival prediction model for nsclc patients through distributed learning across 3 countries. Int J Radiat Oncol Biol Phys 99(2):344–352
Jordon J, Yoon J, Van Der Schaar M (2018) Pate-gan: Generating synthetic data with differential privacy guarantees. In: Proc. of ICLR
Kahla M, Chen S, Just HA, et al (2022) Label-only model inversion attacks via boundary repulsion. In: Proc. of CVPR, pp 15,045–15,053
Kansy M, Raël A, Mignone G, et al (2023) Controllable inversion of black-box face recognition models via diffusion. In: Proc. of ICCV, pp 3167–3177
Karimi Mahabadi R, Henderson J, Ruder S (2021) Compacter: Efficient low-rank hypercomplex adapter layers. In: Proc. of NeurIPS, pp 1022–1035
Karras T, Aila T, Laine S, et al (2018) Progressive growing of gans for improved quality, stability, and variation. In: Proc. of ICLR
Karras T, Laine S, Aila T (2019) A style-based generator architecture for generative adversarial networks. In: Proc. of CVPR, pp 4401–4410
Karras T, Laine S, Aittala M, et al (2020) Analyzing and improving the image quality of stylegan. In: Proc. of CVPR, pp 8110–8119
Khajenezhad A, Madani H, Beigy H (2020) Masked autoencoder for distribution estimation on small structured data sets. IEEE Trans Neural Netw Learn Syst 32(11):4997–5007
Khan A, Rauf Z, Sohail A et al (2023) A survey of the vision transformers and their cnn-transformer based variants. Artif Intell Rev 56(Suppl 3):2917–2970
Kim G, Kwon T, Ye JC (2022) Diffusionclip: Text-guided diffusion models for robust image manipulation. In: Proc. of CVPR, pp 2426–2435
Kingma DP, Ba J (2015) Adam: A method for stochastic optimization. In: Proc. of ICLR
Kingma DP, Dhariwal P (2018) Glow: Generative flow with invertible 1x1 convolutions. In: Proc. of NeurIPS
Kingma DP, Welling M (2013) Auto-encoding variational bayes. arXiv preprint arXiv:1312.6114
Ko M, Jin M, Wang C, et al (2023) Practical membership inference attacks against large-scale multi-modal models: a pilot study. In: Proc of ICCV pp 4848–4858
Kong F, Duan J, Ma R, et al (2024) An efficient membership inference attack for the diffusion model by proximal initialization. In: Proc. of ICLR
Krizhevsky A, Hinton G, et al (2009) Learning multiple layers of features from tiny images
Kuppa A, Le-Khac NA (2021) Adversarial xai methods in cybersecurity. IEEE Trans Inf Forensics Secur 16:4924–4938

Lan Z, Chen M, Goodman S, et al (2020) Albert: A lite bert for self-supervised learning of language representations. In: Proc. of ICLR
Lang O, Gandelsman Y, Yarom M, et al (2021) Explaining in style: training a gan to explain a classifier in stylespace. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp 693–702
LeCun Y, Bottou L, Bengio Y et al (1998) Gradient-based learning applied to document recognition. Proc IEEE 86(11):2278–2324
Lester B, Al-Rfou R, Constant N (2021) The power of scale for parameter-efficient prompt tuning. In: Proc. of EMNLP, pp 3045–3059
Lewis M, Liu Y, Goyal N, et al (2020) Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In: Proc. of ACL
Li XL, Liang P (2021) Prefix-tuning: optimizing continuous prompts for generation. In: Proc. of ACL/IJCNLP, pp 4582–4597
Li H, Guo D, Fan W, et al (2023) Multi-step jailbreaking privacy attacks on chatgpt. In: Proc. of EMNLP
Li S, Liu H, Dong T, et al (2021) Hidden backdoors in human-centric language models. In: Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security, pp 3123–3140
Li X, Tramer F, Liang P, et al (2022) Large language models can be strong differentially private learners. In: Proc. of ICLR
Liu KS, Xiao C, Li B, et al (2019a) Performing co-membership attacks against deep generative models. In: IEEE ICDM, pp 459–467
Liu X, Xie L, Wang Y et al (2020) Privacy and security issues in deep learning: a survey. IEEE Access 9:4566–4593
Liu B, Ding M, Shaham S et al (2021) When machine learning meets privacy: a survey and outlook. ACM Comput Surv 54(2):1–36
Liu J, Li Y, Guo Y et al (2024) Generation and countermeasures of adversarial examples on vision: a survey. Artif Intell Rev 57(8):199
Liu Z, Luo P, Wang X, et al (2015) Deep learning face attributes in the wild. In: Proceedings of the IEEE international conference on computer vision, pp 3730–3738
Liu Y, Ott M, Goyal N, et al (2019b) Roberta: A robustly optimized bert pretraining approach. arXiv preprint arXiv:1907.11692
Liu X, Park DH, Azadi S, et al (2023a) More control for free! image synthesis with semantic diffusion guidance. In: Proc. of WACV, pp 289–299
Liu Y, Peng J, James J, et al (2019c) Ppgan: Privacy-preserving generative adversarial network. In: IEEE (ICPADS), pp 985–989
Liu H, Tam D, Muqeeth M, et al (2022) Few-shot parameter-efficient fine-tuning is better and cheaper than in-context learning. In: Proc. of NeurIPS, pp 1950–1965
Liu R, Wang D, Ren Y, et al (2024b) Unstoppable attack: label-only model inversion via conditional diffusion model. In: IEEE TransInf Forensics Security
Liu X, Zheng Y, Du Z, et al (2023b) Gpt understands, too. AI Open
Lu J, Zhang XS, Zhao T, et al (2022) April: Finding the achilles' heel on privacy for vision transformers. In: Proc. of CVPR, pp 10,051–10,060
Maho T, Furon T, Le Merrer E (2021) Surfree: a fast surrogate-free black-box attack. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 10,430–10,439
Makadia A, Pavlovic V, Kumar S (2008) A new baseline for image annotation. In: Computer vision–ECCV 2008: 10th European conference on computer vision, Marseille, France, October 12-18, 2008, Proceedings, Part III 10, Springer, pp 316–329
Matsumoto T, Miura T, Yanai N (2023) Membership inference attacks against diffusion models. In: IEEE(SPW), pp 77–83
Mattern J, Mireshghallah F, Jin Z, et al (2023) Membership inference attacks against language models via neighbourhood comparison. In: Proc. of ACL, pp 11,330–11,343
Mireshghallah F, Backurs A, Inan HA, et al (2022a) Differentially private model compression. In: Proc. of NeurIPS, pp 29,468–29,483
Mireshghallah F, Goyal K, Uniyal A, et al (2022b) Quantifying privacy risks of masked language models using membership inference attacks. In: Proc. of EMNLP, pp 8332–8347
Mireshghallah F, Taram M, Vepakomma P, et al (2020) Privacy in deep learning: a survey. arXiv preprint arXiv:2004.12254
Mirza M, Osindero S (2014) Conditional generative adversarial nets. arXiv preprint arXiv:1411.1784
Miyato T, Kataoka T, Koyama M, et al (2018) Spectral normalization for generative adversarial networks. In: Proc. of ICLR
Nasr M, Carlini N, Hayase J, et al (2023) Scalable extraction of training data from (production) language models

Nguyen BN, Chandrasegaran K, Abdollahzadeh M, et al (2024) Label-only model inversion attacks via knowledge transfer. In: Proc. of NeurIPS
Nguyen NB, Chandrasegaran K, Abdollahzadeh M, et al (2023) Re-thinking model inversion attacks against deep neural networks. In: Proc. of CVPR, pp 384–393
Nguyen A, Clune J, Bengio Y, et al (2017) Plug & play generative networks: conditional iterative generation of images in latent space. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 4467–4477
Nichol AQ, Dhariwal P, Ramesh A, et al (2022) Glide: Towards photorealistic image generation and editing with text-guided diffusion models. In: Proc. of ICML, PMLR, pp 16,784–16,804
Odena A, Olah C, Shlens J (2017) Conditional image synthesis with auxiliary classifier gans. In: Proc. of ICML, PMLR, pp 2642–2651
Oliynyk D, Mayer R, Rauber A (2023) I know what you trained last summer: a survey on stealing machine learning models and defences. ACM Comput Surv 55(14s):1–41
Pennington J, Socher R, Manning C (2014) GloVe: Global vectors for word representation. In: Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP), pp 1532–1543
Peters ME, Neumann M, Zettlemoyer L, et al (2018) Dissecting contextual word embeddings: architecture and representation. In: Proc. of EMNLP, pp 1499–1509
Qi G, Chen Y, Mao X, et al (2023) Model inversion attack via dynamic memory learning. In: Proc. of MM '23, pp 5614–5622
Qin G, Eisner J (2021) Learning how to ask: querying lms with mixtures of soft prompts. In: Proc. of NAACL, pp 5203–5212
Radford A, Wu J, Child R et al (2019) Language models are unsupervised multitask learners. OpenAI blog 1(8):9
Radford A, Kim JW, Hallacy C, et al (2021) Learning transferable visual models from natural language supervision. In: Proc. of ICML, pp 8748–8763
Radford A, Metz L, Chintala S (2015) Unsupervised representation learning with deep convolutional generative adversarial networks. arXiv preprint arXiv:1511.06434
Radford A, Narasimhan K, Salimans T, et al (2018) Improving language understanding by generative pre-training. arXiv:2001.08361
Rae JW, Borgeaud S, Cai T, et al (2021) Scaling language models: methods, analysis & insights from training gopher. arXiv preprint arXiv:2112.11446
Raffel C, Shazeer N, Roberts A et al (2020) Exploring the limits of transfer learning with a unified text-to-text transformer. J Mach Learn Res 21(1):5485–5551
Ramesh A, Dhariwal P, Nichol A, et al (2022) Hierarchical text-conditional image generation with clip latents. arXiv preprint arXiv:2204.06125
Ren H, Deng J, Xie X (2022) Grnn: Generative regression neural network-a data leakage attack for federated learning. Acm T Intel Syst Tec 13(4):1–24
Rigaki M, Garcia S (2023) A survey of privacy attacks in machine learning. ACM Comput Surv 56(4):1–34
Rombach R, Blattmann A, Lorenz D, et al (2022) High-resolution image synthesis with latent diffusion models. In: Proc. of CVPR, pp 10,684–10,695
Ruiz N, Li Y, Jampani V, et al (2023) Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. In: Proc. of CVPR, pp 22,500–22,510
Rust P, Søgaard A (2023) Differential privacy, linguistic fairness, and training data influence: impossibility and possibility theorems for multilingual language models. In: Proc. of ICML, pp 29,354–29,387
Saha S, Hota A, Chattopadhyay AK et al (2024) A multifaceted survey on privacy preservation of federated learning: progress, challenges, and opportunities. Artif Intell Rev 57(7):184
Saharia C, Chan W, Saxena S, et al (2022) Photorealistic text-to-image diffusion models with deep language understanding. In: Proc. of NeurIPS, pp 36,479–36,494
Sanh V, Webson A, Raffel C, et al (2022) Multitask prompted training enables zero-shot task generalization. In: Proc. of ICLR
Schick T, Schütze H (2021) Exploiting cloze questions for few shot text classification and natural language inference. In: Proc. of EACL, pp 255–269
Schuhmann C, Vencu R, Beaumont R, et al (2021) Laion-400m: Open dataset of clip-filtered 400 million image-text pairs. arXiv preprint arXiv:2111.02114
Shafran A, Peleg S, Hoshen Y (2021) Membership inference attacks are easier on difficult problems. In: Proc. of ICCV, pp 14,820–14,829
Shi Z, Lipani A (2024) Don't stop pretraining? make prompt-based fine-tuning powerful learner. In: Proc. of NeurIPS
Shokri R, Stronati M, Song C, et al (2017) Membership inference attacks against machine learning models. In: IEEE (SP), pp 3–18

Socher R, Perelygin A, Wu J, et al (2013) Recursive deep models for semantic compositionality over a sentiment treebank. In: Proceedings of the 2013 conference on empirical methods in natural language processing, pp 1631–1642
Song M, Wang Z, Zhang Z et al (2020) Analyzing user-level privacy attack against federated learning. IEEE J Sel Areas Commun 38(10):2430–2444
Song Y, Ermon S (2019) Generative modeling by estimating gradients of the data distribution. In: Proc. of NeurIPS
Song C, Raghunathan A (2020) Information leakage in embedding models. In: Proc. of CCS, pp 377–390
Song C, Shmatikov V (2019) Auditing data provenance in text-generation models. In: Proc. of SIGKDD, pp 196–206
Song Y, Shu R, Kushman N, et al (2018) Constructing unrestricted adversarial examples with generative models. Adv Neural Inform Process Syst 31
Song Y, Sohl-Dickstein J, Kingma DP, et al (2020b) Score-based generative modeling through stochastic differential equations. In: Proc. of ICLR
Sousa S, Kern R (2023) How to keep text private? A systematic review of deep learning methods for privacy-preserving natural language processing. Artif Intell Rev 56(2):1427–1492
Struppek L, Hintersdorf D, Correira ADA, et al (2022) Plug & play attacks: towards robust and flexible model inversion attacks. In: Proc. of ICML, pp 20,522–20,545
Sung YL, Cho J, Bansal M (2022) Lst: Ladder side-tuning for parameter and memory efficient transfer learning. In: Proc. of NeurIPS, pp 12,991–13,005
Sung YL, Nair V, Raffel CA (2021) Training neural networks with fixed sparse masks. In: Proc. of NeurIPS, pp 24,193–24,205
Su Y, Wang X, Qin Y, et al (2021) On transferability of prompt tuning for natural language processing. In: Proc. of NAACL, pp 3949–3969
Szegedy C, Zaremba W, Sutskever I, et al (2013) Intriguing properties of neural networks. arXiv preprint arXiv:1312.6199
Thomas A, Adelani DI, Davody A, et al (2020) Investigating the impact of pre-trained word embeddings on memorization in neural networks. In: Proc. of TSD, pp 273–281
Tian Z, Cui L, Zhang C, et al (2023) The role of class information in model inversion attacks against image deep learning classifiers. IEEE Trans Dependable Secure Comput
Torkzadehmahani R, Kairouz P, Paten B (2019) Dp-cgan: Differentially private synthetic data and label generation. In: CVPR workshop
Triastcyn A, Faltings B (2019) Federated generative privacy. In: Proc. of IJCAIW
Vahdat A, Kautz J (2020) Nvae: A deep hierarchical variational autoencoder. In: Proc. of NeurIPS, pp 19,667–19,679
Van Den Oord A, Vinyals O, et al (2017) Neural discrete representation learning. In: Proc. of NeurIPS
van den Burg G, Williams C (2021) On memorization in probabilistic deep generative models. Adv Neural Inform Process Syst 34:27916–27928
Van den Oord A, Kalchbrenner N, Espeholt L, et al (2016) Conditional image generation with pixelcnn decoders. In: Proc. of NeurIPS
Vaswani A, Shazeer N, Parmar N, et al (2017) Attention is all you need. In: Proc. of NeurIPS
Vu T, Lester B, Constant N, et al (2022) SPoT: Better frozen model adaptation through soft prompt transfer. In: Proc. of ACL, pp 5039–5059
Wachter S, Mittelstadt B, Russell C (2017) Counterfactual explanations without opening the black box: automated decisions and the gdpr. Harv JL Tech 31:841
Wang KC, Fu Y, Li K, et al (2021) Variational model inversion attacks. In: Proc. of NeurIPS, pp 9706–9719
Wang D, Liu Y, Tang W, et al (2019) signadam++: Learning confidences for deep neural networks. In: Proc. of ICDMW, pp 186–195
Wang D, Xu T, Zhang H, et al (2022) Pwprop: A progressive weighted adaptive method for training deep neural networks. In: 2022 IEEE 34th International Conference on Tools with Artificial Intelligence (ICTAI), pp 508–515
Wen Y, Marchyok L, Hong S, et al (2024) Privacy backdoors: Enhancing membership inference through poisoning pre-trained models. arXiv preprint arXiv:2404.01231
Wu Y, Yu N, Li Z, et al (2022) Membership inference attacks against text-to-image generation models. arXiv preprint arXiv:2210.00968
Xie L, Lin K, Wang S, et al (2018) Differentially private generative adversarial network. arXiv preprint arXiv:1802.06739
Xu C, Ren J, Zhang D et al (2019) Ganobfuscator: Mitigating information leakage under gan via differential privacy. IEEE Trans Inf Forensics Secur 14(9):2358–2371
Yang AX, Robeyns M, Wang X, et al (2024) Bayesian low-rank adaptation for large language models. In: Proc. of ICLR

Yang Z, Chang EC, Liang Z (2019) Adversarial neural network inversion via auxiliary knowledge alignment. arXiv preprint arXiv:1902.08552
Ye Z, Luo W, Naseem ML, et al (2023) C2fmi: Corse-to-fine black-box model inversion attack. IEEE Trans Dependable Secure Comput
Yuan X, Chen K, Zhang J, et al (2023) Pseudo label-guided model inversion attack via conditional generative adversarial network. In: Proc. of AAAI, pp 3349–3357
Yuan Z, Wu F, Long Y, et al (2022) Secretgen: Privacy recovery on pre-trained models via distribution discrimination. In: Proc. of ECCV, pp 139–155
Yu D, Naik S, Backurs A, et al (2022) Differentially private fine-tuning of language models. In: Proc. of ICLR
Yu D, Zhang H, Chen W, et al (2021) Large scale private learning via low-rank reparametrization. In: Proc. of ICML, pp 12,208–12,218
Zaken EB, Ravfogel S, Goldberg Y (2022) Bitfit: Simple parameter-efficient fine-tuning for transformer-based masked language-models. In: Proc. of ACL, pp 1–9
Zhang G, Liu B, Zhu T et al (2022) Visual privacy attacks and defenses in deep learning: a survey. Artif Intell Rev 55(6):4347–4401
Zhang H, Goodfellow I, Metaxas D, et al (2019) Self-attention generative adversarial networks. In: Proc. of ICML, PMLR, pp 7354–7363
Zhang Y, Jia R, Pei H, et al (2020) The secret revealer: Generative model-inversion attacks against deep neural networks. In: Proc. of CVPR, pp 253–261
Zhang X, Ji S, Wang T (2018) Differentially private releasing via deep generative model (technical report). arXiv preprint arXiv:1801.01594
Zhang N, Li L, Chen X, et al (2022b) Differentiable prompt makes pre-trained language models better few-shot learners. In: Proc. of ICLR
Zhang G, Liu B, Tian H, et al (2024) How does a deep learning model architecture impact its privacy? a comprehensive study of privacy attacks on cnns and transformers. In: USENIX Security 24
Zhang L, Rao A, Agrawala M (2023) Adding conditional control to text-to-image diffusion models. In: Proc. of CVPR, pp 3836–3847
Zhao X, Wang L, Zhang Y et al (2024) A review of convolutional neural networks in computer vision. Artif Intell Rev 57(4):99
Zhao Z, Dua D, Singh S (2018) Generating natural adversarial examples. In: International conference on learning representations
Zhao B, Mopuri KR, Bilen H (2020) idlg: Improved deep leakage from gradients. arXiv preprint arXiv:2001.02610
Zhou J, Chen Y, Shen C, et al (2022) Property inference attacks against gans. In: Proc. of NDSS
Zhu L, Liu Z, Han S (2019) Deep leakage from gradients. In: Proc. of NeurIPS