arXiv:2402.04013v3 [cs.CV] 18 Sep 2026

# Privacy Leakage on DNNs: A Survey of Model Inversion Attacks and Defenses

Hao Fang [1†], Yixiang Qiu [1†], Hongyao Yu [1], Wenbo Yu [1], Jiawei Kong [1], Baoli Chong [2], Bin Chen [2*], Xuan Wang [2], Shu-Tao Xia [1,3], Ke Xu [4]

[1]Tsinghua Shenzhen International Graduate School, Tsinghua University, Shenzhen, Guangdong 518055, China.
[2]School of Computer Science and Technology, Harbin Institute of Technology, Shenzhen, Guangdong 518055, China.
[3]Peng Cheng Laboratory, Shenzhen, Guangdong 518055, China.
[4]Department of Computer Science and Technology, Tsinghua University, Beijing 100084, China.

*Corresponding author(s). E-mail(s): chenbin2021@hit.edu.cn;
Contributing authors: ffhibnese@gmail.com; qiu-yx24@mails.tsinghua.edu.cn; chrisqcwx@gmail.com; wenbo.research@gmail.com; kjw25@mails.tsinghua.edu.cn; chongbaoli@stu.hit.edu.cn; wangxuan@cs.hitsz.edu.cn; xiast@sz.tsinghua.edu.cn; xuke@tsinghua.edu.cn;
†These authors contributed equally to this work.

**Abstract**

Deep Neural Networks (DNNs) have revolutionized various domains with their exceptional performance across numerous applications. However, Model Inversion (MI) attacks, which disclose private information about the training dataset by abusing access to the trained models, have emerged as a formidable privacy threat. Given a trained network, these attacks enable adversaries to reconstruct high-fidelity data that closely aligns with the private training samples, posing significant privacy concerns. Despite the rapid advances in the field, we lack a comprehensive and systematic overview of existing MI attacks and defenses. To fill this gap, this paper thoroughly investigates this realm and presents a holistic survey. Firstly, our work briefly reviews early MI studies on traditional machine learning scenarios. We then elaborately analyze and compare numerous recent attacks and defenses on **D**eep **N**eural **N**etworks (DNNs) across multiple modalities and learning tasks. By meticulously analyzing their distinctive features, we summarize and classify these methods into different categories and provide a novel taxonomy. Finally, this paper discusses promising research directions and presents potential solutions to open issues. To facilitate further study on MI attacks and defenses, we have implemented an open-source model inversion toolbox on GitHub.

**Keywords:** Data Leakage, Model Inversion Attack, Model Inversion Defense, Privacy Security

# 1 Introduction

With the unprecedented development of Deep Learning (DL), Deep Neural Networks (DNNs) have been widely applied in a variety of fields, including medical research [1, 2], financial analysis [3, 4], and personalized recommendations [5]. While many practical applications of DNNs require training on privacy-sensitive datasets, people may mistakenly expect that these private training samples are securely encoded in the weights of the trained network. However, numerous studies [6–8] have demonstrated that a malicious attacker could utilize the pre-trained models to reveal critical privacy information about the underlying training data. Among different types of attacks, Model Inversion (MI), as presented in Fig. 1, stands out as a potent threat that has raised increasing privacy concerns. Unlike the *membership inference attacks* or the *property inference attacks* which only reveal narrow aspects of privacy associated with training data [9], MI allows an adversary to fully reconstruct the training samples by effectively exploiting the guidance from the given pre-trained models [10].

Model Inversion attack was first introduced by [6] on tabular data in the context of genomic privacy. They demonstrate that when partial input attributes are available to the attacker, the sensitive genomic markers can be accurately recovered by maximizing the posterior probability corresponding to a specific output of the given linear regression model. A following study [11] extends MI attacks to decision trees and shadow face recognition networks and reconstructs certain sensitive personal information based on gradient descent optimization. Subsequent studies [12, 13] further broaden the scope of MI attacks to encompass various Machine Learning (ML) models and more forms of data using diverse optimizing strategies. Additionally, significant advancements have also been made in enhancing the attack efficacy while relaxing the assumptions about the attacker's capabilities and knowledge. Although these methods have exhibited outstanding performance in traditional ML scenarios for shallow networks and simple data, the effectiveness is significantly reduced when attacking deeper and wider DNNs trained on more complex and higher-dimensional data such as RGB images [14].

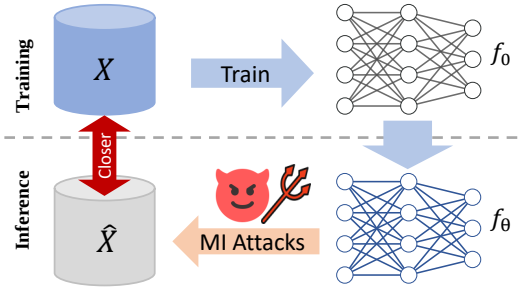



**Fig. 1**: An illustration of MI attacks. The service provider trains the model $f_0(\cdot)$ with private dataset $X$ and obtains $f_\theta(\cdot)$ that is subsequently released for public usage. During the inference stage, the adversary inverts the available $f_\theta(\cdot)$ to acquire recovered data $\hat{X}$, which highly resembles the inaccessible $X$.

To address this challenge, recent studies [14, 15] have made great efforts and achieved remarkable performance improvements. [14] first achieved RGB image recovery from deeper CNNs by exploiting Generative Adversarial Networks (GANs) [16] as image priors. Specifically, they propose to train a GAN model with publicly available data to generate high-fidelity images for target classes. Benefiting from the rich prior knowledge encoded in the learned GAN model, the quality and fidelity of reconstructed images are significantly improved. Building upon this generative recovery paradigm, a variety of subsequent studies [17–19] further propose diverse novel techniques and successfully obtain impressive reconstruction results. Moreover, recent explorations in model inversion on text data [20] and graph data [21] have shown that language models and graph neural networks are also vulnerable to MI threats. Particularly, the serious privacy breach of Large Language Models (e.g., ChatGPT [22]) has drawn growing attention. Users can generate text-based queries and interact with ChatGPT, raising concerns regarding the inadvertent exposure of sensitive information through the model's responses [23]. This threat is greatly amplified in this explosion era of AIGC.

To protect the private training data from powerful MI attacks, a series of defense strategies [24–26] have been proposed to ensure the security of the publicly released models, which can be uniformly divided into *Model Output Processing* and *Robust Model Learning*. In light that MI attacks generally rely on the rich information within the

model output, *Model Output Processing* aims to mislead the attacker by diminishing the valid information in model output [27]. Meanwhile, *Robust Model learning* prefers to incorporate well-designed mechanisms into the model training or fine-tuning process to essentially improve the victim's MI robustness [25].

Despite the privacy threat to training data raised by MI attacks, a comprehensive overview summarizing their advances is currently not available. Recent related research [28] predominantly focuses on the diverse taxonomies and challenges of MI attacks solely based on some early studies only for the tabular and image data, [29] analyzes the Python code implementation of several attacks, or [9] briefly introduces several representative MI attacks in a subsection among various types of privacy inference attacks. In contrast, we present a systematic and elaborate review of the advanced MI attacks and defenses over multiple data modalities and different learning tasks, thoroughly exploring the landscape.

The rest of the paper is organized as follows. Sec. 2 formally defines model inversion attacks and presents the overall profile of MI attacks with diverse taxonomies. Sec. 3 briefly reviews early studies on traditional ML scenarios while Sec. 4 details attacks on DNNs along with a general framework of methods on classification tasks, based on which we systematically introduce the differences and relationships of these methods. Sec. 5 includes MI attacks on more modalities. Sec. 6 then introduces and categorizes numerous defense strategies against MI attacks. Sec. 7 discusses the social impact. Finally, Sec. 8 concludes this work and summarizes some remaining challenges and possible future directions.

**Survey Methodology.** We prioritize papers published in top-tier venues related to deep learning and model inversion, including {NeurIPS, ICML, ICLR, CVPR, ICCV, ECCV, ACL, EMNLP, NAACL, S&P, USENIX security, CCS, NDSS, ACM KDD, AAAI, ACM MM, IJCAI, TPAMI, IJCV, TIFS, TDSC}. For preprints, we consider only those that introduce novel technical designs and demonstrate performance improvements over the state-of-the-art model inversion attacks or defenses available at the time, or extend model inversion to novel tasks or data modalities. Citation count is also taken into consideration, with priority given to preprints that have received more than 20 citations. To ensure comprehensive coverage, we collect candidate papers from multiple sources, *i.e.*, IEEE Xplore, ACM Digital Library, SpringerLink, ACL Anthology, OpenReview, Google Scholar, arXiv, and the official conference proceedings of the venues listed above. To retrieve relevant literature, we adopt a set of related keywords, including {model inversion, inversion attack, inversion defense, privacy leakage, privacy protection, data reconstruction, data extraction, data leakage, generative inversion attack, federated inversion, GAN model inversion, diffusion model inversion, text inversion, LLM inversion, language model inversion, graph inversion}. We independently screen the titles and abstracts to exclude papers unrelated to model inversion. The key criterion is whether the primary research topic satisfies the definition of model inversion [6], as introduced in Section 2.1. Full texts were further examined when necessary to determine whether the paper proposes or analyzes model inversion attacks or defenses. This survey covers works starting from 2014, when model inversion was first explicitly formulated, through April 30, 2026.

Overall, we provide a comprehensive survey of the field, covering more than 130 papers related to model inversion attacks and defenses.

# 2 Model Inversion Overview

In this section, we first formally formulate the definition of Model Inversion attacks. Next, we present an overview of these methods from different perspectives based on the taxonomies as summarized in Figure 2.

## 2.1 Definition of Model Inversion

The process of transforming training data into machine-learned systems is not one-way, but two [30]: the learned models can also be inverted to obtain the training data. Canonical model inversion attacks [10, 14, 21, 31] can be defined as follows: given a pre-trained model $f_\theta(\cdot)$ and certain prior knowledge $\mathcal{D}$ (which may be none), the attacker aims to develop an effective algorithm $\mathcal{T}$ that inverts the network $f_\theta$ to recover as much as possible of the private information about the training dataset $X$. The inversion algorithm $\mathcal{T}$ generally capitalizes on the confidences

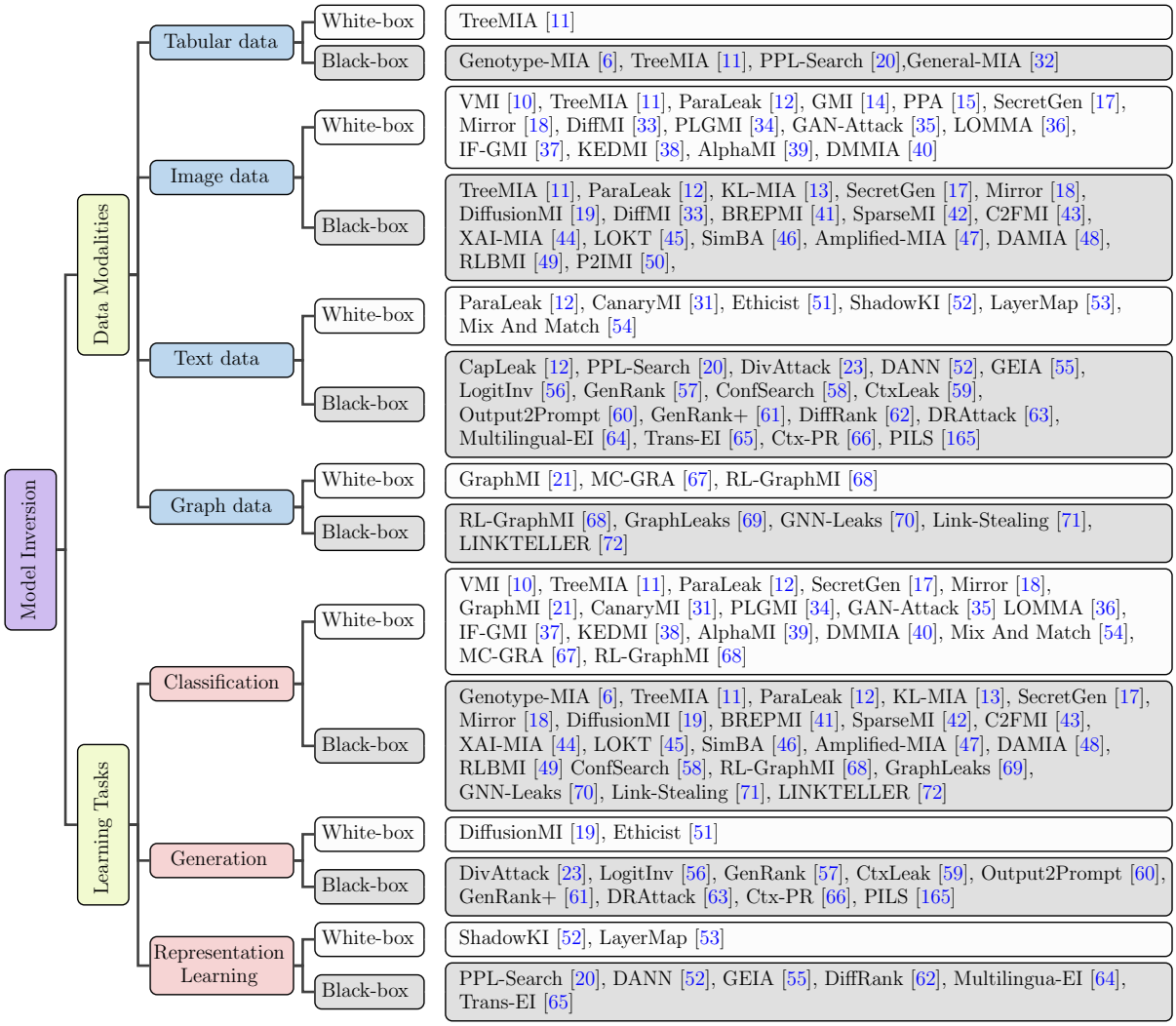



**Fig. 2**: Taxonomy of model inversion attacks across various data modalities and learning tasks.

or feature representations output by the model to achieve satisfactory reconstruction results. A successful model inversion attack should generate realistic reconstructed results $\hat{X} = \mathcal{T}(f_\theta, \mathcal{D})$ that closely resemble the target dataset $X$ without having access to it.

## 2.2 Attacker's Capability

Based on the attacker's capability and knowledge, existing MI methods can be uniformly categorized into ***white-box*** and ***black-box*** attacks. Specifically, a white-box scenario implies that the attacker has full access to the weights and outputs of the target model. Benefiting from the available gradient information of the victim model, white-box attacks [14, 15] have demonstrated excellent recovery efficacy through advanced gradient-based optimization strategies. Conversely, black-box settings only allow access to the predicted confidence probabilities or hard labels, i.e., an attacker can solely control the input query and obtain the model output. To handle this more challenging situation, researchers employ gradient estimation [41, 42] or heuristic algorithms [18, 43] to navigate the data recovery.

## 2.3 Reconstructed Data Modalities

Given that the attack paradigms are largely contingent upon the target data modality, we present

**Table 1**: Comparison of our survey with existing surveys on model inversion.

| Survey | Attack & Defense Taxonomy | Data Modalities | Reproduced Results | Toolkit for MI | Extra Features |
|---|---|---|---|---|---|
| Dibbo et al. [28] | Attack only | Image/Text | – | – | Taxonomy and open challenges |
| Song et al. [29] | Attack only | Image | – | – | Python Implementation analysis |
| Jegorova et al. [9] | – | Image | – | – | – |
| Zhou et al. [78] | ✓ | Image/Text/Graph | – | – | Thorough summary of evaluation metrics and datasets |
| **Ours** | ✓(fine-grained) | Image/Text/Graph | ✓ | ✓ | Rich mathematical formulations and original illustrative figures |

a taxonomy of MI attacks based on data modality to provide a concise overview of these attacks.

*Tabular Data Recovery (Sec. 3).* Pioneer MI studies [6, 11] investigate attacks on tabular data that contains sensitive personal information (e.g., addresses, telephones, and medical records) in traditional ML scenarios. An adversary maliciously accesses an ML model and infers sensitive information by picking feature values for each individual from all possible feature combinations to minimize the expected error rates calculated by the individual's label and the model output.

*Image Data Recovery (Sec. 3 and 4).* Sensitive image data like human faces or medical diagnostic images are also in the scope of MI attacks. Compared with tabular data, an image contains an intractable large set of features to reconstruct, which makes the feature-picking strategy impractical. To address this, [11] proposes an approach against face classifiers that formulates the model inversion as an optimization problem. Subsequent studies [10, 14, 34] generally follow this basic paradigm and further employ numerous techniques (e.g., generative models) to promote the reconstruction of higher-dimensional and higher-resolution images. Another roadmap [13, 50] is to construct a set of input-output pairs by repeatedly querying the victim model and train another inversion network to map the output predictions back to the input images. Besides, [33, 73] consider directly inducing the target diffusion models to generate their private training data.

*Text Data Recovery (Sec. 5.1).* Examples of text data leakage are analogous to tabular ones. One difference is that in certain scenarios, the user input [55] or prompts preceding the input sentences [56] are also pivotal privacy components targeted by MI attacks. Moreover, the distinctive nature of text data and language models motivates rich forms of attack algorithms, which can be broadly categorized into *Embedding Optimization* [31, 51], *Inverse Mapping* [52, 53], *Token search* [57, 58], and *Malicious Prompt Design* [23, 59].

*Graph Data Recovery (Sec. 5.2).* Graphs containing rich personal information and relational data (e.g., friendship or intimacy) between users are also vulnerable to MI attacks. Researchers delve into discrete structures of graph data and propose powerful techniques to achieve successful reconstruction. We further categorize them into: *Adjacency Optimization* [21, 67], *Inverse Mapping* [69, 70] and *Relationship Reasoning* [71, 72].

## 2.4 Learning Tasks

Given that these attacks are also task-specific, we undertake a concise review of these methods from the perspective of learning tasks. Figure 2 demonstrates that most MI studies focus on classification tasks, which typically involve minimizing the classification loss specific to the target label to update the data using diverse optimization algorithms. For generation tasks, attackers usually conduct a heuristic search for the private data based on a pre-defined metric [57, 58] or directly induce the generative models to output their training data [33, 73]. Attacks against representation learning [55, 57] tasks generally target text data, and their algorithms are highly determined by the specific pre-training tasks of the victim model.

*Relationship with Data Modality.* The victim models trained on image and graph data are primarily used for classification tasks, whereas MI attacks on text data generally target language models for generation or representation learning. Correspondingly, the varying learning tasks of victim models result in disparate attack strategies.

## 2.5 Comparison with Existing Related Surveys

The vulnerabilities of DNNs have attracted increasing research attention [37, 74–76]. *E.g.*, [77] discusses backdoor attacks and defenses in

face recognition systems. In this work, we focus on model inversion, a fundamental and critical privacy threat in machine learning systems.

Several recent surveys have explored this topic from different perspectives. A comparison between our survey and existing surveys across several key aspects is provided in Table 1. [28] focuses on early MI studies largely restricted to tabular and image data, and emphasizes taxonomies and open challenges. [29] analyzes Python-based implementations of several attack methods. Meanwhile, [9] briefly discusses MI as part of a broader overview of privacy inference attacks. In comparison, this work provides a more comprehensive and in-depth treatment of both advanced MI attack and defense across diverse data modalities and learning tasks.

A concurrent study [78] also presents an MI survey across multiple domains. However, our work makes several distinct contributions. First, we present a more fine-grained taxonomy of attack methods by organizing key methodological dimensions in Table 2, enabling more systematic comparisons across attacks. We also propose a dedicated taxonomy of defense strategies in Figure 9 for understanding defense design principles.

In contrast to a largely enumerative presentation, we provide a more integrated analysis of existing methods with explicit comparisons, better clarifying their relationships and distinctions. Our work is further supported by rich mathematical formulations and well-designed original illustrative figures. Finally, we contribute an open-source toolkit for MI attacks, which has been widely adopted by the community. Using this toolkit, we reproduce representative attacks and present visualized results in Figure 3, providing intuitive comparisons of their effectiveness.

# 3 MI Attacks on ML Scenarios

This section provides a brief review of early MI attacks and defenses on traditional ML models with simple structured data.

The pioneering study by Fredrickson et al. [6] proposes the first model inversion attack against tabular data in a case study of personalized warfarin dosing [79]. They simply infer the target attribute $x_t^*$ (*i.e.*, genotype) from a linear regression model via *maximum a posterior probability* (MAP) estimate as follows:

$$x_t^* = \arg\max_{x_t} \Sigma_{\mathbf{x}\in\hat{\mathbf{X}}:\mathbf{x}_t=x_t} \Pi_{1\le i\le m} p_i(\mathbf{x}_i), \quad (1)$$

where $\mathbf{x}_i$ is the $i$-th attribute of the sample $\mathbf{x}$, $p_i(\cdot)$ denotes the marginal distribution of $\mathbf{x}_i$, and $m$ is the number of attributes. Given the access to the $k$ insensitive attributes $\mathbf{z} = (x_1, \ldots, x_k)$ and target label $y$, the candidates set $\hat{\mathbf{X}}$ includes samples that match these non-private attributes and are predicted into $y$, i.e., $\hat{\mathbf{X}} = \{\hat{\mathbf{x}} : \hat{\mathbf{x}}_{\mathbf{K}} = \mathbf{z}, f(\hat{\mathbf{x}}) = y\}$, where $\mathbf{K}$ denotes the corresponding indices. To mitigate the stated privacy threat, [80] proposes the first MI defense based on differential privacy (DP) [81]. They leverage an adaptive strategy by adding more noise for sensitive attributes and less noise for non-sensitive attributes to improve MI robustness and retain the utility of target models. Subsequently, more in-depth MI attack methods emerge. [11] explores attacks on decision trees and facial recognition models. They develop novel estimators for decision trees trained on tabular data while applying gradient descent directly on pixels of grayscale images when attacking shallow facial recognition models. Besides, [11] presents the first definition of *black-box* and *white-box* MI attacks according to the knowledge about the target model. To build a universal MI framework, [82] summarizes previous methods and proposes a more rigorous formulation of MI attacks. Considering attacks in the Federated Learning (FL) system, [35] introduces a simple GAN model to enhance the reconstruction. Regarding the shared global model in FL as the discriminator, the attacker trains a generator to learn the same distribution as the target category. Once the training finishes, the generator can directly produce samples similar to private training data. Apart from image recovery, MI attacks were first extended to text modality by [12], which explores attacks from the lens of a service provider for programming code. In the white-box scenario, they manipulate the training codes to encode the user's private information into model parameters using a proposed encoding strategy. After the model is released, the privacy can be extracted by carefully analyzing the model parameters. For black-box attacks, model parameters become inaccessible. Hence, they synthesize extra malicious training data whose labels carry the

encoded private information. The attacker queries the released model with these synthetic data and then analyzes the returned labels to infer the hidden private information. To relax the assumption about access to some non-sensitive attributes [6], [32] designs a general black-box MI framework against the linear regression model that does not require any knowledge about target samples. By considering the data poisoning mechanism, they divide the attack into three stages: *Setup*, *Poisoning*, and *Model Inversion*. In the initial *Setup* stage, the attacker estimates the regression coefficients and necessary hyperparameters of the victim model. During the *Poisoning*, the estimated parameters are utilized to generate poisoned training data that aims to reduce the correlation between non-sensitive attributes and model parameters. Finally, the target model becomes highly correlated with the sensitive attributes and leaks considerable private information in the *Model Inversion* stage. Recently, [83] has explored a novel defense strategy against MI attacks on traditional linear models. Specifically, they introduce sparse coding layers into the classifier, which compute the sparse representation of their inputs by Locally Competitive Algorithm (LCA) [84]. With fewer active neurons, the sparse coding layers can jettison unnecessary private information while maintaining the utility of the classifier, leading to more robustness against MI attacks.

*Inversion Model-Based Image Recovery.* This category of MI attack constitutes a series of black-box methods against relatively shallow networks mostly trained on grayscale images. With the assumption that the predicted vectors of private images are known to the attacker, these approaches treat the target model as a fixed encoder and construct an inversion model as the corresponding decoder, which maps these private confidence vectors of the target model back into the reconstructed images. Formally, the inversion model $\mathcal{M}(\cdot)$ is trained by minimizing:

$$\mathcal{M}^* = \arg\min_{\mathcal{M}} \mathbb{E}_{x\sim X_{pub}} \mathcal{L}(\mathcal{M}(f_\theta(x)), x), \tag{2}$$

where $\mathcal{L}(\cdot,\cdot)$ is the distance loss for reconstructed images, $X_{pub}$ represents the auxiliary public dataset and $f_\theta(\cdot)$ denotes the target model. Once the inversion model is trained, the adversary can reconstruct the target private images with the given corresponding prediction vectors. [13] first proposes this inversion paradigm in scenarios where only corrupted predictions of private images are available (i.e., only partial confidence scores are accessible). Specifically, the attacker collects public data as an auxiliary training dataset for the inversion model. To simulate the corruption in the confidence vectors, the adversary truncates the predictions of public images with top-$k$ selection, which is then fed into the inversion model $\mathcal{M}(\cdot)$ to perform optimization via Eq. (2). When the training is completed, the target images are reconstructed directly from the given prediction through the inversion model. [44] explores the privacy risk in Explainable Artificial Intelligence (XAI) and presents a novel inversion model structure to enhance MI attacks. To better leverage the explanations (e.g., Gradients [85] and LRP [86]) from the target model, they design a multi-modal inversion model to receive the processed explanations as a second input. Besides, an additional network (e.g., Flatten and U-Net [87]) is introduced in front of the inversion model to preprocess the explanations. With the rich information contained in the explanations, they first conquer the challenge of reconstructing RGB images in the context of XAI learning scenarios. To improve the inversion model training, [46] generates adversarial samples with the untargeted attack SimBA [88] as data augmentation to expand the auxiliary training set. In addition to the reconstruction loss in Eq. (2), a novel semantic loss is introduced to ensure the semantic information in the reconstructed samples. A surrogate model is first trained under the guidance of the target model using knowledge distillation techniques. The semantic loss is then computed and serves as the classification loss of the target label provided by the surrogate model.

Previous methods usually attach great importance to the confidence of the target class while neglecting the information from other classes' confidences [47] that measure the differences between each predicted vector of the same target class. To utilize these differences to strengthen sample-wise reconstruction, they insert a *nonlinear amplification layer* ahead of the inversion model to enlarge minor entries within the prediction vector, amplifying the differences between predicted vectors with the same class and further promoting

the inversion model to achieve better sample-wise recovery. Recent work [48] considers the domain divergence issue between the auxiliary and the private dataset. Inspired by the principle of generative adversarial learning, they regard a pre-trained auxiliary classifier as the *generator*, while jointly adversarially training a domain *discriminator* to distinguish the generated and the real private confidence vectors. Under the guidance of the domain *discriminator*, the attacker fine-tunes the *generator* to produce prediction vectors whose domain aligns with the private ones, Correspondingly, the inversion model $\mathcal{M}(\cdot)$ is trained simultaneously with the generated auxiliary prediction vectors as input. Moreover, they also adopt the additional *nonlinear amplification layer* [47] and achieve the recovery of the RGB images on shallow networks for classification tasks. In addition, they [48] further explore the scenario where attackers have access to private features extracted by the target model. The attacker then removes the *nonlinear amplification layer* and adopts a specially designed feature alignment model as the *generator* to utilize the extracted features better.

Although these early MI attacks achieve high efficiency and low computational overhead on traditional ML models, they are limited by the shortcomings of insufficient image generation capability and optimization algorithms. As a result, they perform well solely on shallow models and grayscale images, and fail to conduct valid attacks in more practical scenarios for DNN models or high-dimensional RGB images.

# 4 MI Attacks on DNNs

In this section, we provide detailed introductions to the core features and differences of current mainstream MI attacks on DNNs.

## 4.1 Attacks for Image Classification

Attacks on image classifiers, particularly those for human face recognition, have become a significant research focus of model inversion attacks. Figure 3 shows the visual results reconstructed by several representative state-of-the-art (SOTA) attacks. By reconstructing training data for a given target class, an attacker is able to access an individual's private facial features, which poses a serious privacy breach threat.

We begin by elucidating the general threat model of MI on DNNs. Then, to better clarify and compare these MI approaches, we disassemble the optimization process and summarize the major characteristics of current attacks in Table 2, based on which we systematically analyze the attack paradigm across four perspectives: *Generative Models*, *Data Initialization*, *Attack Process* and *Additional Generic Techniques*. Finally, we review multiple commonly used metrics to evaluate the effectiveness of MI attacks.

### 4.1.1 Threat Model

Due to the remarkable improvement facilitated by generative image priors, generative models have been incorporated into the basic paradigm for subsequent research of MI on DNNs [10, 14, 34]. Figure 4 outlines the threat model for a typical GAN-based model inversion attack. Given the target image classifier $f_\theta(\cdot)$ parameterized with weights $\theta$ and the trained generator $G(\cdot)$, attackers attempt to recover private images $\mathbf{x}^*$ from the generator's latent vectors $\mathbf{z}$ that are initialized by the Gaussian distribution:

$$\mathbf{z}^* = \arg\min_{\mathbf{z}} \mathcal{L}_{iden}(f_\theta(G(\mathbf{z})), c) + \lambda \mathcal{L}_{prior}(\mathbf{z}; G), \tag{3}$$

where $c$ is the target class, $\mathcal{L}_{iden}(\cdot,\cdot)$ typically denotes the identity loss (i.e., the classification loss of the target label), $\lambda$ is a hyperparameter, and $\mathcal{L}_{prior}(\cdot)$ is the prior knowledge regularization, e.g., the discriminator's classification loss used to improve the reality of $G(\mathbf{z})$. Intuitively, the first term attempts to drive the generated images $G(\mathbf{z})$ to be classified as the desired category by the victim classifier, which essentially endeavors to reduce the generations' deviation from the private data based on the perspective of the target model $f_\theta(\cdot)$. The latter term serves as image priors to enhance the quality of the reconstructed samples. Before launching an attack, the adversary needs to acquire a generative model $G(\cdot)$ as prior information about the target images. This can be achieved through either manual training of a specialized generator [14, 34] or direct utilization of a pre-trained generator [15]. Subsequently, the generator is leveraged to generate dummy images that are fed into the target classifier. During the recovery, the attacker performs iterative updates of the latent vectors $\mathbf{z}$ by minimizing the loss function in

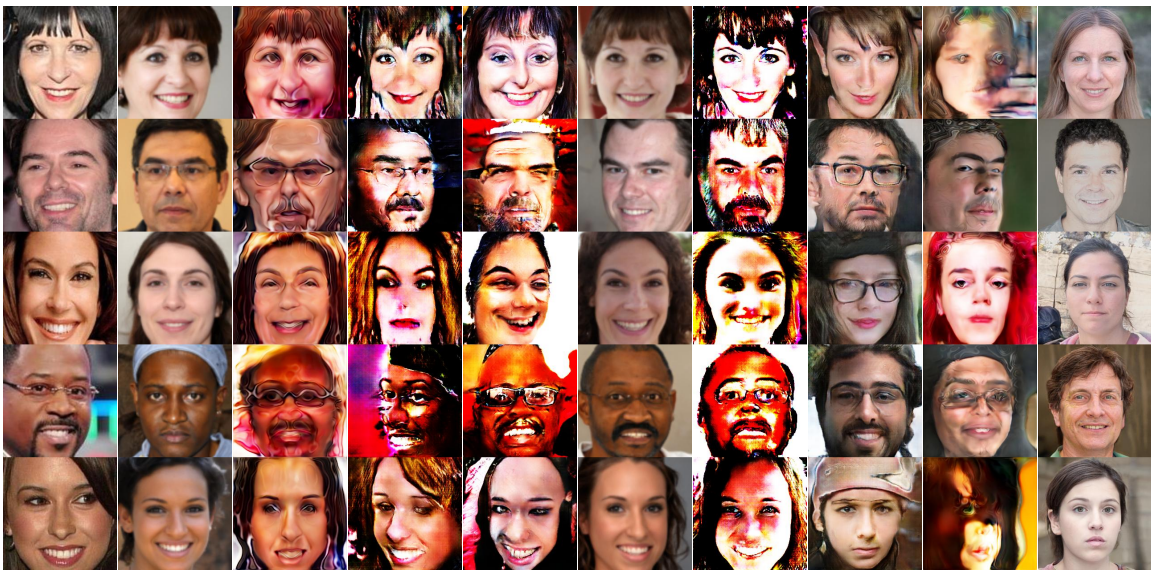



**Fig. 3**: Visual comparison of representative SOTA attacks against image classification. It can be observed that these attacks (e.g., IF-GMI [37]) cause serious privacy leakage about the semantic features in human faces. Note that the last four columns display black-box attacks.

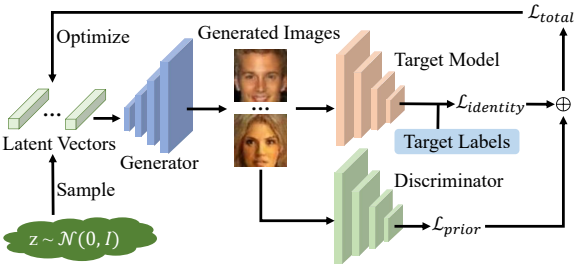



**Fig. 4**: Threat model of generative model inversion attacks against image classification tasks.

Eq. (3). Once obtaining the optimal $\mathbf{z}^*$, the reconstructed images that closely align with the private images can be generated by $\mathbf{x}^* = G(\mathbf{z}^*)$.

### 4.1.2 Generative Models

As introduced in Sec. 3, most previous studies on traditional ML scenarios directly optimize the random noises and neglect the benefits of powerful generative priors, resulting in poor reconstruction outcomes when faced with high-dimensional images [14]. To overcome this difficulty, GMI [14] proposes to employ GAN models to guide image generation and achieve outstanding improvements on deeper CNNs with RGB images. They manually train a GAN model on a large public dataset $X_{pub}$, which contains images with similar structures to the private data. Subsequent studies follow this GAN training paradigm [41, 49] and design special loss functions [14, 17] for effective GAN learning or utilize various GAN architectures tailored to the MI scenario, as summarized in Table 2. Moreover, some methods further exploit information from victim models to strengthen the generator's capability. A widely used technique involves feeding the GAN's training images into the victim classifier to obtain pseudo labels, which then serve as class-condition inputs to the generator [34, 45] and the discriminator [34, 38, 45] in the generative adversarial training.

However, constrained by the limited data volume and poor quality of auxiliary datasets for GAN training, the trained generator is only capable of recovering low-resolution images at 64 × 64. Additionally, performing generative adversarial learning incurs expensive computational overheads and consumes a significant amount of time. One viable resolution is to utilize publicly available pre-trained GANs that have been well-trained on large image datasets [10, 18, 43, 50].

| Method | Generative Models | Data Initialization | | Optimization | | | Extra Features |
|---|---|---|---|---|---|---|---|
| | | Resolution | Preprocess | Optim-algorithm | Loss-fn | Search Space | |
| ***White-Box Attacks*** | | | | | | | |
| GMI [14] | WGAN-GP | $64 \times 64$ | ✗ | Momentum SGD | $\mathcal{L}_{CE} + \lambda\mathcal{L}_{Dis}$ | $\mathcal{Z}$ space | - |
| KEDMI [38] | Inversion-specific GAN† | $64 \times 64$ | ✗ | Adam | $\mathcal{L}_{CE} + \lambda\mathcal{L}_{Dis}$ | $\mathcal{Z}^*$ space | Distributional recovery |
| VMI [10] | DCGAN<br>StyleGAN2-Ada | $64 \times 64$<br>$128 \times 128$ | ✗ | Adam | $\mathcal{L}_{CE} + \lambda\mathcal{L}_{KL}$ | $\mathcal{Z}$ space | Variational inference |
| $\alpha$-GAN-MI [39] | $\alpha$-GAN | $64 \times 64$ | ✔ | - | $\mathcal{L}_{CE}$ | $\mathcal{Z}$ space | - |
| SecretGen [17] | WGAN-GP | $64 \times 64$ | ✔ | Momentum SGD | $\mathcal{L}_{CE} + \lambda\mathcal{L}_{Dis}$ | $\mathcal{Z}$ space | - |
| Mirror [18] | StyleGAN | $224 \times 224$ | ✔ | Adam | $\mathcal{L}_{CE}$ | $\mathcal{P}$ space | $\mathbf{p}$ vector Clipping |
| PPA [15] | BigGAN<br>StyleGAN2-Ada | $224 \times 224$ | ✔ | Adam | $\mathcal{L}_{Poincaré}$ | $\mathcal{W}$ space | Results selection |
| PLGMI [34] | Conditional GAN | $64 \times 64$ | ✗ | Adam | $\mathcal{L}_{MM}$ | $\mathcal{Z}$ space | Pseudo labels guidance |
| LOMMA [36] | ~ | ~ | ~ | ~ | $\mathcal{L}_{Logit} + \lambda\mathcal{L}_{Feature}$ | ~ | Model augmentation |
| DMMIA [40] | StyleGAN2-Ada | $224 \times 224$ | ✗ | Adam | $\mathcal{L}_{CE} + \mathcal{L}_{Memory}$ | - | Dynamic memory bank |
| IF-GMI [37] | StyleGAN2-Ada | $224 \times 224$ | ✔ | Adam | $\mathcal{L}_{Poincaré}$ | $\mathcal{W}$ space | Intermediate feature optimization |
| PPDG-MI [89] | ~ | ~ | ~ | ~ | ~ | ~ | Generator fine-tuning |
| AlignMI [90] | ~ | ~ | ~ | ~ | $\mathcal{L}_{avg}$ | ~ | Training-Free alignment promotion |
| ***Black-Box Attacks*** | | | | | | | |
| SecretGen [17] | WGAN-GP | $64 \times 64$ | ✔ | Momentum SGD | $\mathcal{L}_{Dis}$ | $\mathcal{Z}$ space | - |
| BREPMI [41] | WGAN-GP | $64 \times 64$ | ✔ | Gradient estimation | - | $\mathcal{Z}$ space | - |
| Mirror [18] | StyleGAN | $224 \times 224$ | ✔ | Genetic algorithm | - | $\mathcal{P}$ space | $\mathbf{p}$ vector Clipping |
| BMI-S [42] | StyleGAN2-Ada | $224 \times 224$ | ✗ | Gradient estimation | - | $\mathcal{W}$ space | Style vectors selection |
| C2FMI [43] | StyleGAN2 | $128 \times 128$ | ✔ | Genetic algorithm | - | $\mathcal{W}$ space | Initial coarse optimization |
| RLBMI [49] | WGAN-GP | $64 \times 64$ | ✗ | SAC | - | $\mathcal{Z}$ space | Reinforcement learning |
| LOKT [45] | T-ACGAN | $128 \times 128$ | ✗ | Adam | $\mathcal{L}_{Max-margin}$ | $\mathcal{Z}$ space | Invert surrogate models |
| DiffusionMI [19] | Conditional Diffusion | $64 \times 64$ | ✗ | Optimization-free | - | - | Results selection |
| P2I-MI [50] | StyleGAN2-Ada | $64 \times 64$ | ✗ | Optimization-free | - | - | Prediction alignment |
| SMILE [91] | StyleGAN | $224 \times 224$ | ✔ | MetaModelOnePlusOne [92] | - | $\mathcal{P}$ space | Long-tailed learning |

*The latent vectors are sampled from a learnable Gaussian Distribution.

†This customized GAN structure is first proposed by KEDMI [38].

~ LOMMA [36], PPDG-MI [89] and AlignMI [90] are plug-and-play techniques that can seamlessly adapt to generative attacks.

**Table 2**: Summary of existing MI attacks for visual tasks. Note that SecretGen and Mirror consider both white-box and black-box scenarios.

Compared with GANs trained manually with low-quality auxiliary datasets, these pre-trained generators typically have more intricate structures and hold richer semantic information, enabling higher dimensional reconstruction up to $224 \times 224$ resolution [15, 37, 40, 42].

Furthermore, the recent work DiffusionMI [19] substitutes GANs with the prevalent conditional diffusion model [93]. Specifically, they leverage pseudo-labels classified by the target model as input conditions to train a denoising model with better guidance. This approach demonstrates notable improvements in generation quality over former GAN-based methods, particularly in terms of human perception distance (e.g., LPIPS [94]).

Overall, the evolution trend of generative priors in current MI attacks is characterized by continuously enhancing image generation capability while reducing the deployment overhead for attackers, thereby making the attack algorithms more applicable to real-world scenarios.

### 4.1.3 Data Initialization

We then introduce the features of data initialization in MI. **Resolution.** We summarize the image resolutions employed by different algorithms in Table 2. Given that more image pixels imply more parameters to optimize, the difficulty of MI attacks is highly determined by the resolution of recovered images. Early MI methods resort to reconstructing low-resolution images for improved performance, *e.g.*, GMI [14] and KEDMI [38]. To enable attacks on higher-resolution images,

subsequent studies introduce stronger pre-trained GAN models (e.g., StyleGAN [95]) and manage to recover images with a resolution of $224 \times 224$.

**Preprocess.** While [14, 38] randomly sample initial latent vectors from certain data distributions, [15, 17] highlight that a batch of appropriately initialized latent vectors tends to yield better reconstruction results. A common and effective approach is to employ the classification confidence to assist the selection process [17, 18]. To obtain those latent vectors, the attacker first samples a large number of latent vectors and generates their corresponding images. These images then undergo a series of data augmentation transformations and are fed into the target classifier to obtain their corresponding confidence scores. The latent vectors with the highest confidence scores as the target class are therefore selected for the later optimization process [15, 37]. In addition, C2FMI [43] further proposes a coarse inversion strategy that additionally trains an inverse network $\mathcal{M}(\cdot)$ to improve the initialization process. The attacker first feeds the images $\mathbf{x}$ into the target model $f_\theta(\cdot)$ and a pre-trained feature extractor $\xi(\cdot)$ to acquire predicted confidences and extracted features, respectively. The obtained confidence-feature pairs are leveraged to train the inverse network $\mathcal{M}(\cdot)$ that maps the confidence scores back to the feature space of $\xi(\cdot)$, which then serves as a surrogate model for attackers to perform white-box MI attacks toward the target label. Finally, these optimized latent vectors that can synthesize images close to the target category are subsequently used as the initial latent vectors. [39] conducts this process from a different perspective, which initializes the latent vectors based on real facial features with the image encoder of the $\alpha$-GAN [96]. In practice, the attacker averages several randomly sampled public images and then inputs the averaged image into the $\alpha$-GAN [96]'s image encoder to obtain the initial latent vector.

To obtain extra knowledge gains, GMI [14] and SecretGen [17] additionally consider the scenario where attackers can access the blurred or masked versions of private images and fully leverage them as auxiliary information to enhance the initialization of latent vectors. Specifically, the extracted features of corrupted images are concatenated with randomly initialized noise as input latent vectors to the generator, converting the image generation task into an image restoration task to enhance their underlying semantic information.

#### 4.1.4 Attack Process

**White-Box Attacks.** In the white-box scenario, the attackers have full access to the weights and outputs of the target model, where they primarily conduct the inversion by employing gradient optimization with Momentum SGD or Adam optimizer to minimize the loss function in Eq. (3). This involves computing the identity loss $L_{iden}$ and the prior loss $L_{prior}$. As summarized in Table 2, the majority of white-box methods adopt cross-entropy (CE) loss as the metric to compute the identity loss. However, PPA [15] proves that the cross-entropy loss is susceptible to gradient vanishing issues, which makes it challenging to obtain the optimal latent vectors during the reconstruction. To solve this problem, PPA moves the optimization to hyperbolic spaces with constant negative curvature by employing the Poincaré loss function, which can guarantee sufficient gradient information to the iteration process. Specifically, the $l_1$ normalized output confidences $\hat{y}'$ and the target one-hot vector $y$ are viewed as two points in a Poincaré ball, and the loss is formulated as the hyperbolic distance between two points:

$$\mathcal{L}_{\text{Poincaré}} = \operatorname{arccosh}\left(1 + \frac{2||\hat{y}' - y||_2^2}{(1 - ||\hat{y}'||_2^2)(1 - ||y||_2^2)}\right). \tag{4}$$

PPA adopts an edited version of the typical Poincaré loss by replacing 1 with 0.9999 for enhanced attack performance. Another effective solution proposed by PLGMI [34] is to substitute CE loss with the max-margin (MM) loss function $\mathcal{L}_{MM}$, which maximizes the output logit of the target class while minimizing that of an additional category with the largest logit:

$$\mathcal{L}_{MM} = -l_c(x) + \max_{j \neq c} l_j(x), \tag{5}$$

where $c$ is the target class and $l_i$ represents the output logit of the class $i$. The resultant derivative of max-margin loss concerning the logits contains constant values that can help avoid the gradient vanishing problem. Moreover, PLGMI points out that the max-margin loss facilitates searching for samples that closely resemble the target class while distinctly different from other classes,

which aligns with the basic goal of MI attacks. Nevertheless, LOMMA [36] rethinks the optimization objective and proposes a different viewpoint. They emphasize that the fundamental goal of MI attacks is to reconstruct images that highly resemble those in the target class, rather than deviating from non-target classes. Since CE loss inherently combines both objectives, they suggest bypassing the softmax function in the cross-entropy loss and directly maximizing the logit of the target class. Recent work AlignMI [90] has deciphered the core mechanism of MIAs through the geometric lens of the manifold hypothesis. They reveal that better alignment between loss gradients and the generator manifold leads to more serious vulnerability. Building on this insight, they propose to sample multiple variants within the local neighborhood of synthetic samples during optimization and compute the average of the corresponding loss gradients. This operation suppresses off-manifold noise and enhances gradient-manifold alignment, achieving strong attack in a training-free manner.

As mentioned in Section 3, various regularization terms are introduced as prior knowledge to ensure the quality and fidelity of the generated images. One representative attempt is the use of the discrimination classification loss $\mathcal{L}_{Dis}$, i.e., the realism penalization from GAN's discriminator [14, 17]. However, [15, 18] note that the GAN is trained on a public dataset and thus the $\mathcal{L}_{Dis}$ might impair the inversion performance by matching the generated images with the distributions of the public data rather than the private ones. Consequently, subsequent MI methods no longer employ this term to avoid the distribution shift. Apart from the constraint on the confidence scores, LOMMA [36] analyzes that successful MI attacks are expected to require the feature similarity between the reconstructed and private images given by the target model. Therefore, an $\mathcal{L}_{Feature}$ term is utilized to minimize the distance between the penultimate layer representations of reconstructed samples and public images, further boosting the performance of MI attacks.

Unlike the above methods, DMMIA [40] focuses on the issue in the GAN model itself, i.e., the catastrophic forgetting problem where the generated images from early reconstruction contain more diverse characteristics, but some disappear as the optimization goes further, decreasing the diversity of inverted images. To solve this issue, DMMIA uses the generated samples to build two types of memory banks, which are the learnable intra-class multicentric representation (IMR) and the non-parametric inter-class discriminative representation (IDR), respectively. IMR learns multiple distinctive features of intra-class images to represent the target class while IDR stores the historical knowledge into prototypes for each class. Based on the two dynamically maintained memory banks, they propose a novel regularizer $\mathcal{L}_{Memory} = \lambda_1\mathcal{L}_{imr} + \lambda_2\mathcal{L}_{idr}$, where $\mathcal{L}_{imr}$ leverages IMR to prevent the optimization from overfitting to specific image features and $\mathcal{L}_{idr}$ encourages the reconstruction of images with more class-distinguishable characteristics.

Instead of instance-level data reconstruction, KEDMI [38] aims to directly recover the private data distribution for a given label with a learnable Gaussian distribution $\mathcal{N}(\mu, \sigma^2)$. They adopt the reparameterization trick to make the loss differentiable and optimize the learnable parameters $\mu$ and $\sigma$. Once finished modeling the private data distribution, attackers can directly sample $z$ from the learned distribution to generate sensitive images. VMI [10] also discusses the distribution-level recovery of the training data. To establish theoretical understanding, VMI formulates the MI attack as a variational inference that attempts to estimate the target posterior distribution $p(\mathbf{x}|y)$ with a variational distribution $q(\mathbf{x})$. Based on the theoretical analysis, an extra KL-divergence regularizer is introduced to constrain the distance between a learnable distribution $q(\mathbf{z})$ estimated by deep flow models [97] and the latent distribution $p_{AUX}(\mathbf{z})$ of public auxiliary images in the GAN's latent space. By optimizing the flow model using the proposed loss function, the attacker gradually optimizes the latent distribution $q(\mathbf{z})$ to reach the latent area of target data, further estimating the target data distribution $q(\mathbf{x})$ through the GAN model $G(\cdot)$, i.e., $q(\mathbf{z}) \xrightarrow{G} q(\mathbf{x})$.

While the latest MI attacks have achieved outstanding performance in white-box settings, their reliance on white-box access to the target model restricts their applicability exclusively to offline-deployed or open-source models, making them inapplicable to real-world scenarios where service providers only expose black-box API interfaces.

**Black-Box Attacks.** In contrast to white-box settings, this type of attack has no access

to any information from the target model except for the output content. Consequently, the gradient information is unavailable for performing the back-propagation operation.

SecretGen [17] proposes a straightforward solution to the problem of gradient inaccessibility. The attacker first samples numerous latent vectors from random noise and selects the ones that produce images predicted as the correct labels. These vectors are then optimized exclusively with the discriminator loss to improve the reconstruction quality. RLBMI [49] introduces a reinforcement learning-based model inversion attack that leverages confidence scores to provide rewards and optimizes the agent model with the Soft Actor-Critic (SAC) [98] algorithm. This algorithm employs an actor-critic architecture where both states and actions are represented as the latent vectors of the generator, and the next state is computed as a weighted combination of the current state and the action. In this actor-critic framework, the actor network generates an action based on the current state, while the critic network estimates the corresponding reward. To achieve the goal of MI, RLBMI designs the reward with three components: state reward, action reward, and distinguishability reward. The state and action rewards are defined as the confidence scores of the target label with respect to the next state and the action respectively, and distinguishability reward measures the disparity between the confidence score of the target class and the highest confidence score of other classes. By updating the actor-critic networks, the actor network learns to produce actions that are able to yield higher rewards, which contributes to significantly enhancing the effectiveness of MI attacks.

*Label-only scenarios.* BREPMI [41] proposes a boundary-repelling strategy to tackle label-only scenarios where only the predicted hard labels are available. Their intuition is that the farther image from the class decision boundary tends to be more representative of the class. Consequently, BREPMI utilizes zero-order optimization to urge the latent vector to gradually move away from the decision boundary. In light that images that are not predicted as the target class represent incorrect optimization directions, the attacker estimates the gradients $\hat{g}_{\mathbf{z}}$ of latent vectors $\mathbf{z}$ opposite to these misclassified images by randomly sampling $N$ unit vectors $\mathbf{u}$ and calculating the loss with sampled points on a sphere at radius $r$:

$$\hat{g}_{\mathbf{z}} = \frac{1}{N}\sum_{i=1}^{N}\Phi_c(\mathbf{z}+r\mathbf{u_i})\mathbf{u_i}, \tag{6}$$

where $\Phi_c(\cdot)$ denotes a function that equals zero if the generated image is classified as the target class $c$, otherwise $-1$. The attacker calculates the gradient as an average over those misclassified points and optimizes the latent vector $\mathbf{z}$ using gradient ascent to move them in the direction opposite to this unintended average. During the optimization, BREPMI increases the value of $r$ when all sampled points are classified into the target label to better estimate the gradients. A following study BMI-S [42] adopts a similar pipeline with BREPMI [41] but assumes the soft labels are available. The research replaces the $\Phi_c(\cdot)$ with CE loss in Eq. 6 and fixes the value of $r$ during the optimization. LOKT [45] also considers label-only scenarios and introduces a Target model-assisted ACGAN (T-ACGAN). During the T-ACGAN training, LOKT employs the target model to tag public auxiliary images and synthesized images generated by generator $G$ with pseudo labels as class supervision to improve the training of ACGAN's discriminator $D(\cdot)$ and classification head $C(\cdot)$ (where $D(\cdot)$ is used to penalize the image realism and $C \circ D(\cdot)$ is an auxiliary classifier to provide identity loss for better GAN learning). With the training of ACGAN finished, the attacker could directly use $C \circ D(\cdot)$ as the surrogate model or additionally train multiple surrogate models with sufficient image-label pairs generated by $G(\cdot)$ and labeled by the victim model $f_\theta(\cdot)$. Finally, the attacker executes SOTA white-box MI methods on the target model replaced by the obtained surrogate models, which achieves satisfying attack results.

*Gradient-free optimizer.* Orthogonal to the above methods, Mirror [18] and C2FMI [43] firstly explore gradient-free optimization techniques, which utilize genetic algorithms instead of gradients for optimization. Specifically, the population of $k$ candidates in the genetic algorithm represents the latent vectors of GANs, and the fitness score is defined as the confidence score for the target label. To obtain an adequate initial population, Mirror [18] employs a simple initial selection strategy while C2FMI [43] proposes a more complicated coarse optimization method as detailed in

Section 4.1.3. In each iteration of the genetic algorithm, a subset of candidates is selected as parents and undergoes crossover to generate new candidates. Additionally, some candidates are mutated either by adding random noise (Mirror [18]) or by crossing over with the candidate that has the highest score (C2FMI [43]) to create more high-quality candidates. The top $k$ candidates with the highest scores are retained for the next iteration. This process repeats over multiple iterations and continually refines the candidates toward achieving higher confidence scores for the target label. Finally, results are generated from the candidates with the highest scores via the GAN model.

P2I-MI [50] claims that optimization-based methods suffer from high cost and low efficiency. Thus, P2I-MI proposes an optimization-free training-based strategy to overcome this issue. The attacker trains a prediction alignment encoder (PAE) to map the output prediction to the style vector $\mathbf{w}$ in the $\mathcal{W}$ space of StyleGAN2-Ada. During the attack stage, the research proposes an ensemble scheme that samples a batch of public images and gets their corresponding predicted confidences to be mapped into style vectors with the learned PAE. The final style vector $\mathbf{w}$ is calculated as a weighted average of these style vectors, which serves as the condition input for StyleGAN2-Ada to generate the recovered image. In addition, the prevalent denoising diffusion models are also introduced to MI attacks. DiffusionMI [19] uses an auxiliary dataset labeled with pseudo labels to train a class-conditional diffusion model, which is directly used to iteratively generate plenty of target images without any further optimization during the attack phase.

Recent study SMILE [91] argues that existing black-box MIAs suffer from two critical limitations: prohibitive query overhead and the extremely long-tailed distribution of samples generated by the generator. To address these limitations, SMILE proposes a novel black-box MIA method with a two-stage pipeline. In the first stage, a long-tailed surrogate model training scheme is adopted: the first n-1 layers of the pre-trained model are frozen to serve as the shared backbone, with only 3 ensemble classifiers being trained. The surrogate model is optimized by integrating a distillation loss, a diversity loss, and a top-k reweighting strategy. In the second stage, an automatic black-box optimization algorithm selector NGOpt is leveraged to identify the optimal gradient-free black-box optimization algorithm and the MetaModelOnePlusOne [92] algorithm is selected finally. Meanwhile, a P-space clipping strategy is applied during the optimization process to expand the search space, thereby significantly reducing the overall query overhead.

Overall, the critical challenge limiting the real-world application of current MI attacks in black-box settings lies in the increased complexity of the optimization process resulting from the absence of the gradient information.

**Search Space.** Most previous studies focus on finding the optimal vectors in the GAN's latent space $\mathcal{Z}$. However, [18] shows that as the resolution increases, optimization becomes under-constrained due to the large space sparsity. Feature entanglement in $\mathcal{Z}$ space is another challenge that further impedes the optimization process since images sharing similar features may not correspond to analogous latent vectors. StyleGANs [95] transform the latent space into the well-disentangled $\mathcal{W}$ space through a mapping network $G_{mapping} : \mathcal{Z} \rightarrow \mathcal{W}$. Therefore, searching the $\mathcal{W}$ space alleviates the above issues, which has been adopted by a series of following studies [15, 43].

Previous methods perform the attack within a fixed search space. In contrast, IF-GMI [37] innovatively operates optimization across multiple feature spaces to fully exploit the hidden information in GAN's intermediate layers, as shown in Fig. 5. To perform the intermediate layer optimization, a StyleGAN2-Ada $G(\cdot)$ is divided into $L+1$ blocks as follows:

$$G(\cdot) = G_{L+1} \circ G_L \circ \cdots \circ G_2 \circ G_1(\cdot). \quad (7)$$

During the attack process, the attacker first optimizes the inputs of $G_1(\cdot)$ and then continues the optimization intermediate features layer by layer until reaching $G_L(\cdot)$. Moreover, IF-GMI [37] employs a $l_1$ ball to limit the optimization space of each feature during the intermediate search. The latest advance PPDG [89] has taken a further step by directly fine-tuning the GAN prior during the optimization process. Specifically, pseudo-private data (*i.e.*, surrogate samples containing the characteristic features of private training data) is first generated using the target model and the GAN prior. Next, a top-k strategy

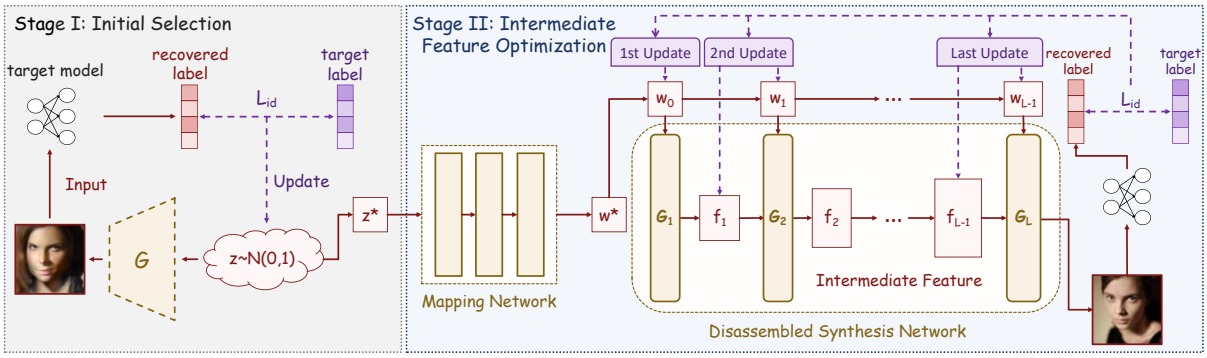



**Fig. 5**: Illustration of the pipeline of IF-GMI attack procedure.

is adopted to select high-quality pseudo-private data based on the model's prediction confidence. Finally, the screened pseudo-private data is used to fine-tune the GAN prior, thereby narrowing the gap between the GAN prior distribution and the private data distribution, thereby fundamentally boosting the performance of MIAs.

Besides, some methods propose to constrain the search space to avoid unrealistic image generation. Mirror [18] and SMILE [91] suggest a $\mathcal{P}$-space clipping strategy to constrain the final $\mathbf{w}$ vectors within the target distribution. The $\mathcal{P}$ space is defined as the feature space located before the final LeakyReLU function in the mapping network. Numerous vectors are first sampled from $\mathcal{Z}$-space to compute the mean $\mu$ and variance $\sigma$ of the activation values in $\mathcal{P}$-space. Given an optimized $\mathbf{w}$ in $\mathcal{W}$-space, attackers first obtain $\mathbf{p}$ by projecting $\mathbf{w}$ to $\mathcal{P}$-space. Then $\mathbf{p}$ is clipped within the range $[\mu - \sigma, \mu + \sigma]$ and projected back to the $\mathcal{W}$-space. This operation preserves the style control capabilities of the $\mathcal{W}$ space while ensuring these controls are within the desired distribution.

### 4.1.5 Additional Generic Techniques

Various innovative techniques have been explored and incorporated into numerous studies as auxiliary. Next, we supplement a detailed review of several generic mechanisms that are leveraged in multiple attack methods.

**Pseudo Label Guidance.** As introduced before, pseudo labels have been utilized to guide the training of the image generator [19, 34, 45], the discriminator [34, 38, 45], and surrogate models [36, 45]. Specifically, the victim image classifier is leveraged to reclassify unlabeled images (*e.g.*, the images generated from the GAN model or public images) with labels from the private dataset. Since the victim model is trained on the private dataset, the provided pseudo labels contribute to exploiting additional privacy information within the private training data, thus enhancing the capabilities of the generator or discriminator trained on these auxiliary labeled images.

**Augmentation.** Many studies have integrated various augmentation techniques into the MI workflow to improve the attack effects. [17] employ sequential cutout as a data augmentation for images to improve the initial latent vector selection. [15, 19] process the reconstructed images with image transformation and select results with higher confidence from the output of the target model. Furthermore, [34] performs random augmentations on the generated images before feeding them to the target model to provide more stable convergence to realistic images during the GAN training. In addition to traditional data augmentations, [36] presents a novel approach called model augmentation. This involves training several auxiliary models from the target model using model distillation techniques. During the MI process, the adversary utilizes an ensemble of the original target model and the trained auxiliary models to calculate the loss function. By augmenting with surrogate models, this strategy mitigates the issue of overfitting to the target model and encourages the reconstructed images to capture more distinctive features of the private data.

**Results Selection.** [15] notes that DNNs often exhibit overconfidence in their predictions, leading to low transferability in attack results. More concretely, while some reconstructed images

are labeled with high confidence by the target model, they receive low scores when evaluated by another model. To overcome this challenge, [15, 19] apply data augmentations to the generated images before classifying them with the target classifier. By selecting results with the highest confidence score after the augmentation, this approach achieves recovery with enhanced attack accuracy and superior transferability.

#### 4.1.6 Evaluation Metrics

Assessing the degree of privacy leakage raised by MI attacks is another pivotal open issue. Generally, researchers evaluate the performance based on the similarity between the reconstructed images and target class images. In addition to the common distance measurement (e.g., PSNR [99], LPIPS [94] or FID [100]), novel metrics tailored for MI evaluation are introduced and widely adopted. We summarize these metrics in detail as follows.

- **Accuracy.** This metric is calculated as the classification accuracy of an auxiliary evaluation classifier (trained on the same private dataset as the victim model but with a different architecture) when applied to these reconstructed samples. It serves as a pivotal criterion for how well the generated samples resemble the target class. The higher accuracy the synthesized samples achieve on the evaluation model, the more privacy in the training dataset is exposed [14].
- **Feature Distance.** The feature distance is defined as the $\mathcal{L}_2$ distance between the feature vector of the reconstructed image and the feature centroid of the private samples specific to the target class. Note that the feature vector is the output from the penultimate layer of the evaluation classifier.
- **$K$-nearest neighbor (KNN) distance.** The KNN distance represents the minimum distance from the reconstructed image to the images within the target class [14]. In practical terms, the $K$ is usually 1, i.e., the nearest image within the target class to the synthesized sample is first identified, and their feature distance is computed as the KNN distance metric.
- **Sample Diversity.** [10] proposes this metric that focuses on the intra-class diversity of the reconstructed images. It first computes the improved Precision-Recall [101] and Density-Coverage [102] scores regarding the classification results of the reconstructed images on a given evaluation model. The average of the two quantities is the *sample diversity*, whose higher values indicate greater intra-class diversity of the synthesized samples.

### 4.2 Attacks against Image Generation

While most existing algorithms are designed for image classification tasks, recent research also explores the potential for inverting the generative model to reconstruct their private training data.

Previous research [103] demonstrates that GAN models [16, 104] are susceptible to model inversion attacks. By repeatedly sampling images from the generator's output distribution, it's highly likely to generate raw samples from the GAN's training set. Compared with GAN models, the denoising diffusion model [105] is able to produce images of higher quality and realism. However, [33] indicates that they are also more susceptible to being inverted to reconstruct private data, which leaks nearly twice as much training data as GAN models.

Specifically, they propose a generation and filtering mechanism and target a range of prevalent text-conditioned diffusion models, such as Stable Diffusion [106] and Imagen [107]. The extraction process consists of two steps: (1) repeatedly query the diffusion model with the selected text prompt to obtain a large number of images, and (2) construct a graph using these generations by establishing an edge between the generated images if they are considered to be sufficiently similar based on a pre-defined distance metric. Ultimately, the largest clique in the graph, which comprises at least 10 nodes, is identified as a memorized image. In essence, this serious data leakage issue is a production of diffusion's significant data *memorization* phenomenon, which allows them to regenerate individual raw training samples under certain guidance.

A subsequent study [108] further examines the influencing factors on data memorization and confirms that the conditioning input of diffusion models greatly increases the data memorization and thus renders them more vulnerable to MI attacks. Building upon this line, SIDE [73] delves into the

more challenging attacks on unconditional diffusion models. To establish a universal theoretical analysis, SIDE first proposes a novel and effective memorization metric to provide theoretical support for enhanced attacks, which also corroborates that the input conditions can enhance the data memorization in diffusion models and further result in more information leakage. To address the difficulty of the missing input conditions in unconditional diffusion models, they propose an effective solution that additionally trains a time-dependent classifier and uses its classification information as surrogate conditions to guide the denoising process for better private data generation.

Existing attacks have been able to effectively induce memory leakage, but their practical applicability is still confined to untargeted data leakage, failing to achieve targeted privacy extraction.

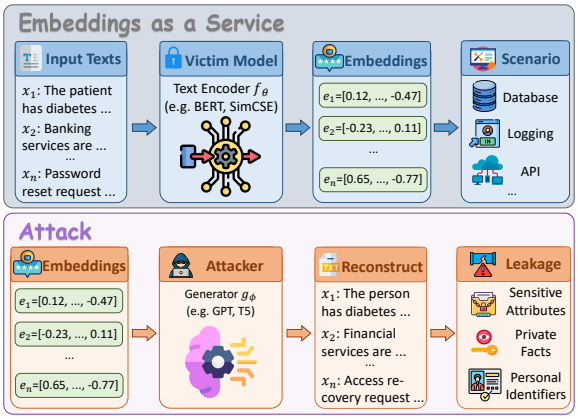



**Fig. 6**: Overview of embedding inversion attacks in Embeddings-as-a-Service. Leaked sentence embeddings can be exploited by an attacker to reconstruct the original texts, leading to exposure of sensitive attributes, private facts, and personal identifiers.

# 5 Data Modalities

While existing MI attacks on DNNs mainly focus on image recovery, recent progress has extended the attacks to text and graph data, which also garners significant attention and concerns.

## 5.1 Text Data

DNN models that process natural-language text also suffer from MI attacks. This has been largely amplified by the rise of LLMs since these models are widely used and trained on massive text corpora that often contain considerable privacy-sensitive information. Formally, we divide these various attacks into *Embedding Optimization*, *Inverse Mapping*, *Token Search*, and *Malicious Prompt Design*.

**Embedding Optimization.** These attacks mainly take advantage of the full access to the victim model by formulating the reconstruction as an optimization process that utilizes the back-propagated gradients. [31] attacks the LSTM model $f_\theta(\cdot)$ by conducting a discrete optimization, where each token $x_i$ of the sentence $s$ is represented as a logit vector $z_i$ to get the embedding $E_s$ of the dummy sentence $s$. Given the target label $y$, the vectors $z_1, \ldots, z_n$ are repeatedly optimized with the gradients computed from the cross-entropy loss $\mathcal{L}(f_\theta(E_s), y)$. Subsequently, [51] adopts the paradigm of prompt learning that freezes the parameter of the victim GPT-Neo and tunes the pre-sentence soft prompt embedding $S$ by forcing the model to output the target suffix $b$ for a given prefix $a$. The learned prompt $S$ elicits the *memorization* of the target language model and induces it to respond with private training data. Additionally, it introduces a smoothing regularizer to make the loss distribution of the suffix sequence smoother. Despite their effectiveness, these methods rely on strong white-box access and costly iterative optimization, which limits their practicality in real-world LLM services where only black-box APIs are usually available.

**Inverse Mapping.** The rapid development of *general-purpose language models* (e.g., BERT [109]) makes them widely used feature extractors to encode sentences into dense embeddings. In the privacy-sensitive learning with service providers, users may expect that it is a safe way to submit text features encoded using the language model. However, research has demonstrated that the submitted sentence embeddings can also incur the leakage of users' private data, as illustrated in Figure 6. To be specific, [52] constructs two types of attacks, namely pattern reconstruction attacks and keywords inference attacks. It collects an external corpus $\mathcal{D}_{ext} = \{s_i\}_{i=1}^N$ to extract sensitive information $\{T(s_i)\}_{i=1}^N$ as labels and query the language model to obtain the sentence embedding $\{f_\theta(s_i)\}_{i=1}^N$. An inversion model consisting of a linear SVM and a 3-layer MLP with 80 hidden units is trained on the $(f_\theta(s_i), T(s_i))$ pairs to invert

the sentence embedding to the sensitive information for attack purpose. However, [53] claims that the direct inversion from output to input with only one step is a highly non-convex optimization issue and limits the recovery accuracy, especially as the victim model grows deeper. Therefore, it divides the inversion into two stages: (1) learn a mapping function $\mathcal{M}(\cdot)$ to map the higher layer embedding $f_\theta(s_i)$ to the shallow-layer features (rather than the input words), and (2) infer the private words by matching their corresponding embedding with the obtained shallow layer representations $\mathcal{M}(f_\theta(s_i))$. Nevertheless, one common drawback of these two attacks is they only reconstruct orderless sets of words that lack meaningful semantic information. To address this problem, [55] reformulates the attack as a generation task, which trains a generative attacker model, *i.e.,* GPT-2, to iteratively map the sentence embedding $f_\theta(s)$ with previous contexts $x_1, \ldots, x_{i-1}$ into the private word $x_i$. Subsequently, [64] further studies embedding inversion beyond monolingual English settings by investigating black-box attacks against multilingual sentence embeddings. They show that generative inversion models can recover text content across multiple languages, even without explicit prior knowledge of the target language. However, these studies typically assume unlimited query access to the victim model. [65] investigates a more realistic query-free black-box setting, where the attacker cannot access or query the victim encoder and only observes a small set of leaked text-embedding pairs. Their method learns a surrogate encoder to mimic the victim model, and then trains an embedding-to-text attacker on large-scale surrogate-generated embeddings. The resulting attacker exhibits strong transferability, which can generalize to embeddings produced by unseen victim encoders. From a practical perspective, inverse-mapping attacks highlight a fundamental trade-off between attack realism and reconstruction quality. Query-based methods can obtain stronger reconstruction performance by repeatedly accessing the victim encoder, but this assumption may be unrealistic for deployed embedding services with strict rate limits or closed APIs. In contrast, query-free transferable attacks better match leakage scenarios in real-world systems, yet their effectiveness depends on the availability of representative leaked pairs and the quality of the surrogate encoder.

The capability of LLMs can be greatly enhanced through well-crafted prompts, which has promoted the continued design of high-quality instructional prompts by service providers. These prompts have also become a valuable asset within their intellectual property rights (IPR). However, malicious users can cause a serious infringement of their IPR by recovering the invisible prompt via the model output. [56] trains an inversion model $\mathcal{M}(\cdot)$ with an encoder-decoder backbone, to map the model's next-token probabilities $\mathbf{v}$ back to initial tokens $x_{1:N}$. During the recovery, the output probability vector of the next token is first unrolled into a sequence of pseudo-embeddings, which are then encoded to provide conditions for the decoder to recover the target prompt. Since LLMs are usually available via API, the output confidence vectors may be inaccessible. Given the typical LLM setting that provides the argmax of the confidences and allows users to add a logit bias to adjust the output distribution, the probability of each token can be estimated based on its difference from the most probable word. This is calculated as the smallest logit bias required to make that word most likely. PILS [165] exploits the redundancy of language models' next-token distributions by replacing full vocabulary-sized logprob vectors with compact representations over selected tokens. It further incorporates multi-step logprob sequences into an encoder-decoder inverter, thereby improving hidden prompt recovery while reducing logprob access costs. Instead of estimating the confidence distribution, Output2prompt [60] directly uses LLM output sequences as model input to train the inversion model $\mathcal{M}(\cdot)$. Besides, a sparse encoder is employed to reduce the memory costs of training $\mathcal{M}(\cdot)$. These methods provide a systematic route to recover hidden prompts from model outputs, but their deployment practicality depends on the availability of informative output signals and the high cost of training an additional inversion model, which may fail to satisfy in black-box scenarios.

**Token Search.** In the black-box scenario, only model outputs are accessible, and the gradient back-propagation becomes impractical. Researchers choose optimization-free *search-based* strategies to handle this challenge. Given the sentence's prefix tokens $a = x_1, x_2, \ldots, x_i$, attackers attempt to search the most probable pathway for generating the remaining portion of the sentence,

i.e., $b = x_{i+1}, x_{i+2}, \ldots, x_n$. Based on the search objective for the subsequent tokens, these methods are further divided into three parts as follows.

**Perplexity-metric Search.** Carlini et al. [20] use *perplexity* to measure the probability of data sequences. Specifically, given a sequence $s = x_1 \ldots x_n$ and the target generative sequence model $f_\theta$ (i.e., LSTM, qRNN), the perplexity $\mathrm{P}_\theta(s)$ can be expressed as:

$$\mathrm{P}_\theta(s) = \exp\left(-\frac{1}{n}\sum_{i=1}^{n}\log f_\theta\left(x_i|x_1,\ldots,x_{i-1}\right)\right), \tag{8}$$

where $\mathrm{P}_\theta(s)$ quantifies the "surprise" of a given sequence, and a lower perplexity value indicates a higher probability that the language model has seen the sequence. In this manner, the problem is transformed into finding the sequence with minimal perplexity. However, directly enumerating all possible sequences and computing their perplexity leads to an exponentially growing search space. A series of mechanisms has been proposed to improve search efficiency. [20] uses a modification of the Dijkstra algorithm, which efficiently reduces the search range by avoiding visiting unnecessary sequence paths. The subsequent method [57] adopts a greedy-based algorithm that only retains tokens with top-$k$ confidence during each iteration of generating the next token. Moreover, they provide several techniques for improved generated sentence diversity and enhanced attack accuracy. [61] adopts a look-ahead mechanism to improve the quality of generated tokens. Instead of only depending on the next token's probability for the top-$k$ selection, they use the posterior probability of the subsequent sequence to inform the generation of the next token $x_t$.

**Confidence-metric Search.** Considering the sentiment classification task, [58] searches for the next token that maximizes the confidence score of target class $y$ as predicted by a fine-tuned BERT classifier. To counteract the target model's bias towards its high-frequency tokens, an auxiliary regularizer is introduced to penalize the frequent occurrence of certain tokens.

However, this method also faces the challenge of an exponentially growing search space. To tackle this, [54] first uses BERT with the original generation head to generate candidate tokens with top-$k$ confidence, which are further selected by maximizing the probability of label $y$ with the classification head.

**Difference-metric Search.** The pre-train and fine-tune paradigm has gained growing popularity, where models are first trained on a large, public dataset and then fine-tuned on a small, private dataset. Accordingly, several attacks appear to invert the fine-tuning dataset by exploiting the difference between the pre-trained model $f_\theta$ and fine-tuned model $f_{\theta'}$. Considering the token sequence $s = x_1 \ldots x_n$, [62] defines the difference score $DS_{f_\theta}^{f_{\theta'}} = \sum_{i=1}^{n} f_{\theta'}(x_i|x_{<i}) - f_\theta(x_i|x_{<i})$, which measures the difference in the output probabilities between $f_\theta$ and $f_{\theta'}$. Intuitively, a larger difference score value indicates a higher probability that the fine-tuned model $f_{\theta'}$ has seen sequence $s$ while the original $f_\theta$ does not, i.e., $s$ belongs to the fine-tuning dataset. By performing a beam search, they select the next tokens with the highest differential scores and achieve a remarkable attack success rate. Subsequently, [63] defines a novel metric $score_r$, which additionally considers the distribution-level difference between the target sentence and the current set of reconstructed sentences to more accurately quantify the difference. They first query the fine-tuned GPT-2 model $f_{\theta'}$ with an empty prompt. Then a complete sentence $S_f$ is generated by iteratively querying $f_{\theta'}$ until it outputs $EOS$ token. After iteratively querying $f_\theta$ with the first half of $S_f$ as the initial prompt, they obtain $S_g[l/2:]$ generated by the initial pre-trained model. As the larger difference between $S_g[l/2:]$ and $S_f[l/2:]$ indicates the greater possibility that $S_f$ comes from the fine-tuning dataset, the attacker calculates the $score_r$ based on $S_g[l/2:]$ and $S_f[l/2:]$ to determine whether the generated sentence belongs to the private fine-tuning dataset, thus achieving the aim of data extraction.

Overall, these search-based attacks provide a practical gradient-free route for text recovery, especially when only generated outputs or confidence scores are available. Nevertheless, they trade off accessibility for efficiency: the discrete search space grows exponentially with sequence length, and the attack performance heavily depends on the quality of the search heuristic and the availability of sufficient model queries.

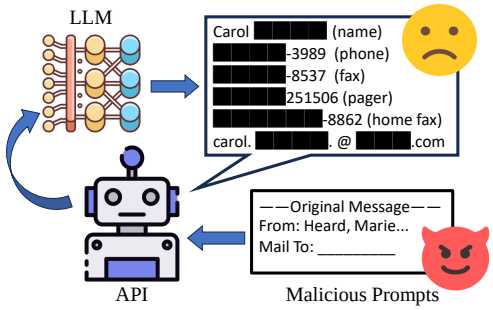



**Fig. 7**: An example of privacy leakage on LLMs.

**Malicious Prompt Design.** The conversational LLMs have demonstrated powerful capabilities and great potential. However, several studies have shown that some elaborately designed prompts can invert the model to output sensitive training data. As Figure 7 depicts, feeding malicious prompts into language models such as GPT-Neo can incur serious privacy information leakage [59] in both zero-shot and few-shot settings (i.e., providing several demonstrations as prior knowledge to guide LLM's predictions). More surprisingly, [23] suggests that data can also be extracted by simply asking ChatGPT to repeat a word multiple times. One possible reason is that this instruction causes the model to "escape" its aligning training and revert to its original language modeling objective. Beyond training-data extraction, malicious prompts may also induce conversational LLMs to disclose their hidden system instructions. [66] formulates this threat as a prompt extraction attack, where the adversary queries a conversational API with carefully designed instructions that encourage the model to repeat or summarize its prior context. By aggregating the returned generations, the attacker can identify the most plausible reconstruction of the secret prompt. This attack shows that malicious prompt design can expose not only memorized training content but also confidential instructions that govern LLM-based services.

Malicious prompt design offers a lightweight and practical attack surface because it only requires standard API access. Nevertheless, its effectiveness depends heavily on the model's instruction-following capability and deployed safety filters, making it less controllable than attacks based on explicit confidence scores, gradients, or learned inversion models.

## 5.2 Graph Data

Graph data [110–114] illustrates the linkage situation between any pair of nodes and indicates their mutual interaction. Graph Neural Networks (GNNs) [115–119] exhibit remarkable efficacy in processing complex graph-structured data and have achieved superior performance in non-Euclidean domains. The wide application of GNNs has brought prominent privacy leakage risks. For instance, the gradient inversion attack GraphLeak [120] proposes to invert patients' electronic health record graphs from shared gradients. In the area of MI attacks, given the target GNN $f_\theta(\cdot)$ pre-trained on the private graph $\mathcal{G}$, the adversary aims to infer the adjacency matrix $\hat{\mathcal{A}}$, which is then converted into graph $\hat{\mathcal{G}}$ to reconstruct the private training dataset. As shown in Figure 8, MI attacks on this modality can be characterized into *Adjacency Optimization*, *Inverse Mapping*, and *Relationship Reasoning*.

**Adjacency Optimization.** These attacks reconstruct the private graph by directly optimizing the adjacency matrix. Since back-propagated gradients are needed for updating, they generally require white-box settings. Denote the prediction vector or the embeddings output by the target GNN model as $\mathcal{H}$, i.e., $\mathcal{H} = f_\theta(\mathcal{G})$. An adjacency matrix $\hat{\mathcal{A}}_0$ is first initialized to obtain the dummy graph $\hat{\mathcal{G}}_0$ and the dummy output $\hat{\mathcal{H}}_0 = f_\theta(\hat{\mathcal{G}}_0)$. By minimizing the distance $\mathcal{L}_{rec}$ between $\hat{\mathcal{H}}_0$ and $\mathcal{H}$ with gradient descent, an attacker iteratively updates $\hat{\mathcal{A}}$ to find the optimal solution:

$$\hat{\mathcal{A}}_{t+1} = \hat{\mathcal{A}}_t - \eta_t \nabla_{\hat{\mathcal{A}}_t} \mathcal{L}_{rec}, \tag{9}$$

where $t$ is the number of iterations, $\hat{\mathcal{A}}_t$ is the adjusted adjacency matrix after each iteration, and $\eta_t$ is the corresponding learning rate. Based on this, [21] proposes GraphMI that uses a projected gradient module to tackle the discreteness of graph data while introducing the feature smoothness term $\mathcal{L}_s$ and the F-norm term $||\hat{\mathcal{A}}||_F$ for the regularization of feature smoothness and sparsity. RL-GraphMI [68] extends GraphMI to the more challenging hard-label black-box settings by adopting gradient estimation and reinforcement learning methods [121–125]. From the perspective of information theory, [67] proposes to regard $f_\theta(\cdot)$ as a Markov chain and iteratively optimizes $\hat{\mathcal{A}}$ via a flexible chain approximation, which further achieves excellent attack results. Specifically, they

adapt the original iterative optimization problem of (9) into a chain-based formulation as:

$$\hat{\mathcal{A}}^* = \arg\max_{\hat{\mathcal{A}}} \underbrace{\lambda_p \mathcal{I}(\mathcal{H}_{\mathcal{A}}; \mathcal{H}^i_{\hat{\mathcal{A}}})}_{\text{propagation approximation}} - \underbrace{\lambda_c \mathcal{S}(\hat{\mathcal{A}})}_{\text{complexity}} + \underbrace{\lambda_o \mathcal{I}(\boldsymbol{Y}_{\mathcal{A}}; \boldsymbol{Y}_{\hat{\mathcal{A}}}) + \lambda_s \mathcal{I}(Y; \boldsymbol{Y}_{\hat{\mathcal{A}}})}_{\text{outputs approximation}}, \tag{10}$$

where $\mathcal{I}(\cdot;\cdot)$ is the mutual information, $\mathcal{S}(\cdot)$ is the information entropy, $\mathcal{H}_{\mathcal{A}}$ denotes the embeddings of the original adjacency matrix $\mathcal{A}$, $\mathcal{H}^i_{\hat{\mathcal{A}}}$ denotes the representations of the dummy adjacency matrix $\hat{\mathcal{A}}$ output by the $i$-th layer ($1 \leq i \leq L$) from the $L$-layer GNN target model $f_\theta(\cdot)$, $\boldsymbol{Y}_{\mathcal{A}}$ denotes the classification outputs transformed from $\mathcal{H}_{\mathcal{A}}$ by the downstream linear layer with activation, $\boldsymbol{Y}_{\hat{\mathcal{A}}}$ denotes the classification outputs transformed from $\mathcal{H}^L_{\hat{\mathcal{A}}}$ by the downstream linear layer with activation, $Y$ denotes the labels for the classification task, and $\lambda_p, \lambda_c, \lambda_o, \lambda_s$ are the weight factors. By optimizing $\hat{\mathcal{A}}$ based on (10), a series of intermediate representations are also taken into consideration in addition to the final model output. Thus, the attackers can maximize the approximation of encoding and decoding processes of the target model to enhance the correlation between the ground-truth $\mathcal{A}$ and the estimated $\hat{\mathcal{A}}$, and minimize the complexity to avoid non-optimal solutions by reducing the graph density.

**Inverse Mapping.** Different from the adjacency optimization methods that iteratively optimize $\hat{\mathcal{A}}$, these attacks attempt to construct an inverse mapping that reverses the original input-to-output mapping of the target model $f_\theta(\cdot)$ to recover the private graph. Most of them are intended for black-box settings where only the GNN output $\mathcal{H}$ is available. Given the difficulty of reconstruction with limited information, some additional auxiliary knowledge is introduced. Specifically, they first train an autoencoder $\mathcal{F}_{auto}(\cdot)$ on an auxiliary graph dataset $\mathcal{G}_{aux}$ by minimizing the reconstruction loss between $\hat{\mathcal{G}}_{aux}$ and $\mathcal{G}_{aux}$, so that mutual conversion between graph data and embeddings can be adequately learned. Once $\mathcal{F}_{auto}(\cdot)$ has been fully trained, they disassemble the encoder $\mathcal{F}_{enc}(\cdot)$ and decoder $\mathcal{F}_{dec}(\cdot)$ from $\mathcal{F}_{auto}(\cdot)$, and utilize $\mathcal{F}_{dec}(\cdot)$ to map $\mathcal{H}$ back to $\hat{\mathcal{G}}$, i.e., $\hat{\mathcal{G}} = \mathcal{F}_{dec}(\mathcal{H})$. [69] leverages

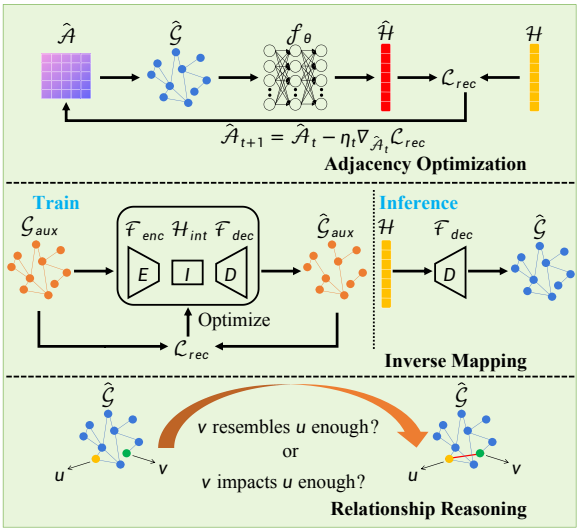



**Fig. 8**: Different types of MI attacks on graph data.

such inverse mapping methods and reconstructs $\hat{\mathcal{G}}$ with high accuracy to quantify privacy leakage in graph embeddings. However, the distribution of latent features $\mathcal{H}_{int}$ within the autoencoder $\mathcal{F}_{auto}(\cdot)$ may differ from the original distribution of $\mathcal{H}$. To alleviate this issue, [70] proposes to query $f_\theta(\cdot)$ with $\mathcal{G}_{aux}$ to acquire $\mathcal{H}_{aux}$, i.e., $\mathcal{H}_{aux} = f_\theta(\mathcal{G}_{aux})$, and fine-tune $\mathcal{F}_{dec}(\cdot)$ with the constructed $(\mathcal{H}_{aux}, \mathcal{G}_{aux})$ pair for enhanced reconstruction performance.

**Relationship Reasoning.** These attacks are also designed for black-box scenarios. To successfully conduct the more challenging black-box attacks, they generally require some extra prior knowledge in addition to the GNN output $\mathcal{H}$, such as node attributes in the training data. Unlike the above two types of methods that constantly update $\hat{\mathcal{A}}$ or disassemble $\mathcal{F}_{dec}(\cdot)$ from $\mathcal{F}_{auto}(\cdot)$, they reconstruct $\hat{\mathcal{G}}$ by reasoning about the relationships among pairs of nodes using the introduced auxiliary knowledge. [71] considers three dimensions of auxiliary knowledge (node attributes, partial graph, or an auxiliary dataset), and evaluates $2^3 = 8$ settings based on whether each of the three dimensions is available. [71] presumes that two nodes $u$ and $v$ are linked if they share more similar attributes or predictions. They add an edge between $u$ and $v$ in the reconstructed graph when the distance of feature vectors between them is below a given threshold,

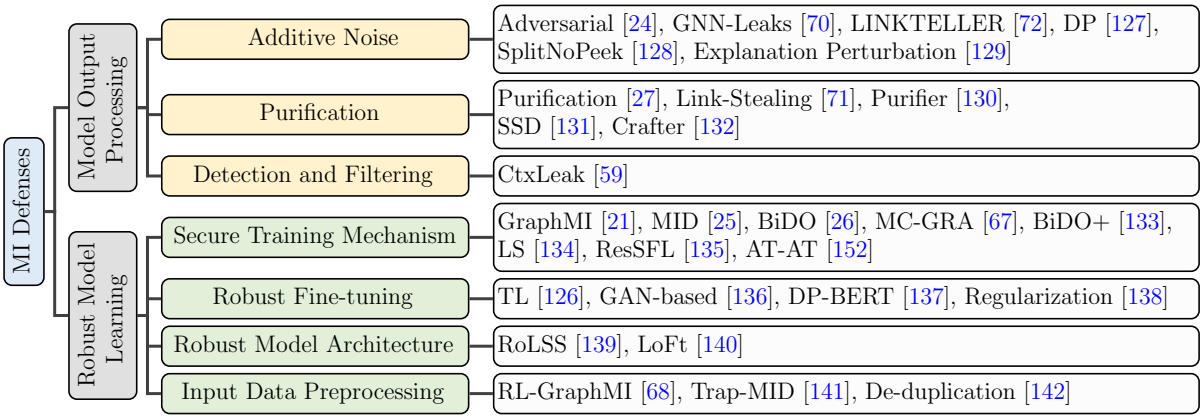



**Fig. 9**: Taxonomy of defense strategies against MI attacks.

and various metrics tailored for each of the 8 auxiliary knowledge settings are designed to calculate such distances. Contrary to the above settings, [72] assumes that the attackers only have access to node attributes in the training data. They suggest that if there is an edge between $u$ and $v$, the information of $u$ would be propagated to $v$ during training. Therefore, they assume that $u$ and $v$ are linked if changing the feature vector of $u$ can impact the prediction of $v$ to a certain extent.

Beyond algorithmic innovation, the practical deployment of graph MI attacks reveals critical trade-offs that merit further discussion. For instance, while adjacency optimization methods such as [21] achieve high-fidelity reconstruction in white-box settings, their reliance on gradient back-propagation limits applicability to offline or open-source models, rendering them ineffective against real-world online APIs; inverse mapping and relationship reasoning approaches [69, 71] operate in black-box scenarios but often depend on auxiliary graph datasets or node attributes that may be unavailable in privacy-sensitive domains such as healthcare. One potential solution is to develop hybrid graph MI attack methods that combine the strengths of these methods, leveraging transfer learning [126] to reduce the dependence on auxiliary graph datasets and attributes.

# 6 Defenses against MI Attacks

A rich quantity of defense strategies have been proposed to alleviate MI threats and achieved impressive resistance effects. Basically, these methods focus on improving MI robustness by eliminating the valid information stemming from the pre-trained model that an MI adversary might maliciously utilize. As presented in Figure 9, we primarily divide them into two categories: *Model Output Processing* and *Robust Model Learning*, as shown in Figure 10. Next, we illustrate these methods in accordance with the proposed taxonomy.

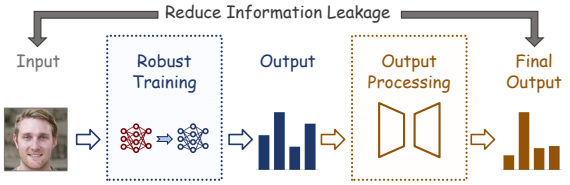



**Fig. 10**: General overview of the defense category.

## 6.1 Model Output Processing

Since MI attacks fundamentally exploit the redundant information contained in the victim model's output [25], a viable defense mechanism is to reduce this redundancy by obscuring the model output with various techniques.

**Additive noise.** Intuitively, [24] proposes perturbing the confidence scores with well-designed noise crafted through adversarial learning with an inversion model. The inversion model serves as an MI adversary, which aims to recover the input data from the confidence vector. Meanwhile, the adversarial noise is updated to resist the inversion model by maximizing the MSE between the original images and images reconstructed from the inversion model. To maintain the main task accuracy, they impose a constraint on the noise that

assures the predicted label unchanged. Considering the Differential Privacy (DP) [143–147] guarantee, [127] designs an approach that divides the score vector into multiple sub-ranges and applies an exponential mechanism to replace and normalize the values in each sub-range. This mechanism assures the target model's MI robustness with rigorous theoretical proof of differential privacy.

[128] further examines the scenario of split learning, where a global model is split into several components and distributed to disparate clients. During the training, each client calculates the features output from its partial model and then uploads them to the central server for aggregation. Considering the uploaded features also face the threat of being converted into private images by an attack model, [128] suggests a simple yet efficient resolution that incorporates Laplacian noise into the intermediate representations before transmitting them to the computing server, thereby obscuring the potential MI attacker.

To protect graph data, [70] also applies Laplacian noise to the model output $\mathcal{H}$. They release a noisy but usable version of output embeddings $\widetilde{\mathcal{H}} = \mathcal{H} + \Delta$, where $\Delta$ denotes the Laplacian noise. [72] also develops a differentially private GNN model and adds noise to the adjacency matrix to guarantee DP. [129] perturbs feature explanations using a randomized response mechanism and reduces the attack to a random guess. Since applying noise may result in performance degradation, it is essential to choose a decent level of noise and achieve a trade-off between utility and privacy.

**Purification.** This type of defense aims to purify the confidence scores by reducing the redundant information associated with privacy leakage, which contributes to significantly increasing the difficulty of MI attacks.

In light that a major mechanism of MI attacks is that the outputs are distinguishable for diverse input images, [27] seeks to defend by rendering model predictions less distinguishable and less sensitive to input data changes. Concretely, they propose an autoencoder-based purifier $P(\cdot)$ that encourages different input images to be mapped into the same or similar output by reducing the dispersion of the confidence scores from target classifier $f_\theta(\cdot)$. To enhance the purifier $P(\cdot)$, they employ an adversarial learning technique, which involves jointly learning an adversarial inversion model $\mathcal{M}(\cdot)$ via minimizing the inversion loss:

$$\mathcal{L}_{inv} = ||\mathcal{M}(P(f_\theta(\mathbf{x}))) - \mathbf{x}||_2^2, \tag{11}$$

where $\mathbf{x}$ denotes the input images. To train the purifier $\mathcal{P}(\cdot)$, they additionally introduce the confidence reconstruction loss $\mathcal{L}_{rec}$ and the classification loss $\mathcal{L}_{iden}$ as:

$$\begin{aligned} \mathcal{L}_{rec} &= ||P(f_\theta(\mathbf{x})) - f_\theta(\mathbf{x})||_2^2, \\ \mathcal{L}_{iden} &= \mathbb{CE}(P(f_\theta(\mathbf{x})), \arg\max f_\theta(\mathbf{x})). \end{aligned} \tag{12}$$

Finally, the purifier's optimization objective can be formulated as $\mathcal{L}_{total} = \mathcal{L}_{rec} + \lambda_1 \mathcal{L}_{iden} - \lambda_2 \mathcal{L}_{inv}$, where $\lambda_1$ and $\lambda_2$ are hyperparameters. [130] follows this pipeline and substitutes the autoencoder with a variational autoencoder with the predicted label as conditioning input, which enhances the output indistinguishability and thus ensures the model's MI resiliency. A novel defense framework is to intentionally mislead the attackers into reconstructing public samples [132]. They consider the edge computing scenario with similar settings to the split learning [128], where a user uploads the intermediate features $Enc(x)$ encoded from its private image $x$ to the central service provider. To protect the $Enc(x)$ from the threat of an MI attacker, the edge device carefully optimizes these vulnerable features to be closer to the features of the public sample, hence misleading the attacker into recovering insensitive public images. Besides, an Euclidean distance is adopted to restrict the deviation from the original features to further preserve the utility of perturbed features.

SSD [131] proposes a novel defense strategy specifically designed for black-box scenarios. They propose to postprocess output logits to minimize the conditional mutual information between inputs and outputs. This objective is formulated as a convex optimization problem and solved via a water-filling algorithm.

[71] considers guarding graph data by restricting the GNN model to output only the $k$ largest posterior probabilities, which purifies the confidences to diminish unnecessary information and thus decreases the potential risk under MI threats.

**Detection and Filtering.** A distinctive attribute of language models is that the model's output contents possess meaningful semantics and can be further analyzed for security concerns. Therefore, [59] proposes an effective

| Defense | Publication | Process Type | Core Feature |
|---|---|---|---|
| SplitNoPeek [128] | ICLR Workshop'21 | Additive noise | Apply Laplacian noise |
| Adversarial [24] | IEEE CSR'21 | Additive noise | Adversarial sample optimization |
| DP [127] | IEEE TIFS'22 | Additive noise | Differential privacy technique |
| Purification [27] | arXiv'20 | Purification | Autoencoder-based purifier |
| Purifier [130] | AAAI'23 | Purification | Conditional VAE-based purifier |
| Crafter [132] | NDSS'24 | Purification | Optimize to near public samples |
| SSD [131] | ICLR'25 | Purification | Reduce conditional mutual information |
| MID [25] | AAAI'21 | Secure training mechanism | Minimize mutual information |
| BiDO [26] | KDD'22 | Secure training mechanism | Bilateral dependency optimization |
| ResSFL [135] | CVPR'22 | Secure training mechanisms | Adversarial training with attacks |
| LS [134] | ICLR'24 | Secure training mechanism | Negative label smoothing |
| BiDO+ [133] | IEEE TPAMI'25 | Secure training mechanism | Outlier exposure regularizer |
| AT-AT [152] | ICLR'26 | Secure training mechanism | Adversarially non-robust feature |
| GAN-based [136] | IEEE TIFS'23 | Robust Fine-tuning | Adversarial training with attacks |
| TL [126] | CVPR'24 | Robust Fine-tuning | Freeze first few layers |
| RoLSS [139] | ECCV'24 | Robust model architecture | Remove skip connections |
| LoFt [140] | KDD'26 | Robust model architecture | Low-rank feature filtering |
| Trap-MID [141] | NeurIPS'24 | Input data preprocessing | Image trigger injection |

**Table 3**: Summary of existing MI defenses for visual tasks.

defense approach that detects and filters the privacy-sensitive information within the prediction. Specifically, this method includes a detection module to examine whether the output text contains sensitive information and a filtering module to take appropriate measures based on the detection module's action like refusing to answer.

Overall, the main challenge of output processing defenses is to reduce privacy leakage without severely harming model utility; moreover, since such methods operate by perturbing or purifying returned outputs, they are mainly applicable to black-box API settings and are generally unsuitable for open-source or offline-deployed models where attackers can directly access model parameters and internal information.

## 6.2 Robust Model Learning

Despite the considerable success of these model output processing methods, they are constrained to black-box scenarios, where an attacker is merely capable of making queries and obtaining the corresponding output. In contrast, in white-box scenarios, attackers gain complete access to the actual model output before the post-processing. To address this challenge, researchers seek to enhance the model's inherent robustness.

**Secure Training Mechanism.** This series of algorithms modifies the model training process to boost the MI robustness. The pioneering research MID [25] is a classic approach to exploring robust training and analyzes that the inversion attack greatly depends on the correlation between the input $X$ and output $\hat{Y}$ of the target model $f_\theta$, thereby proposing to penalize the mutual information between $X$ and $Y$ during model training. The training objective can be expressed as:

$$\min_{\theta} \mathbb{E}_{(x,y)\sim p_{X,Y}(x,y)} \mathcal{L}(f_\theta(x), y) + \lambda \mathcal{I}(X, \hat{Y}), \quad (13)$$

where $y$ is the ground truth label, $\mathcal{L}(\cdot,\cdot)$ denotes the loss for the main task, $\mathcal{I}(\cdot,\cdot)$ denotes the mutual information, and $\lambda$ is a coefficient balancing privacy and utility. However, it is impracticable to directly compute the $\mathcal{I}(X, \hat{Y})$ due to the difficulty of modeling the joint distribution of model input and output. Inspired by the work on information bottleneck, they substitute it with its upper bound $\mathcal{I}(X, Z)$ [148], where $Z$ represents a stochastic encoding of the input $X$ at some intermediate layers. Then a variational method is employed to approximate the $\mathcal{I}(X, \hat{Y})$ term.

Nonetheless, MID meets the dilemma between data privacy and model utility as the regularizer conflicts with the main task loss. [26] proposes

a bilateral dependency optimization (BiDO) to solve this problem. Instead of directly diminishing the correlation between inputs and outputs, BiDO minimizes the dependency $d(X, Z)$ between inputs $X$ and latent representations $Z$ while maximizing the dependency $d(Z, \hat{Y})$ between latent representations $Z$ and outputs $\hat{Y}$. The first term limits the propagation of redundant information from inputs to the latent representations, preventing misuse by the adversary. The latter term facilitates the latent layers to learn discriminative representations, ensuring model utility for classification. The overall training loss can be formulated as:

$$\begin{aligned}\mathcal{L}_{BiDO} = \mathcal{L}_{CE}(\hat{Y}, Y) + \lambda_x \sum_{i=1}^{m} d(X, Z_i) \\ -\lambda_y \sum_{i=1}^{m} d(Z_i, \hat{Y}),\end{aligned} \tag{14}$$

where $\lambda_x$ and $\lambda_y$ denote the hyperparameters controlling the intensity of loss items, and $Z_i, i = 1, \ldots, m$ denotes the $i$-th layer's latent representations specific to the input $X$. Additionally, the study evaluates both Constrained Covariance [149] and the Hilbert-Schmidt Independence Criterion (HSIC) [150] as distance metrics to quantify dependence and finds that HSIC yields better defense performance. BiDO+ [133] further augments BiDO with an outlier exposure (OE) regularizer to mitigate its insufficient OOD awareness, which further enhances the defense effectiveness.

Label smoothing [134], initially proposed to improve model generalization or calibration, is investigated by [151] to defend model inversion attacks. Given the initial one-hot label $\mathbf{y}$, the class number $C$, and the smoothing factor $\alpha$, the smoothed label is defined as:

$$\mathbf{y}^{LS} = (1 - \alpha) \cdot \mathbf{y} + \frac{\alpha}{C}. \tag{15}$$

Correspondingly, the training loss incorporating the label smoothing technique is presented as:

$$\mathcal{L}_{LS} = (1-\alpha)\mathcal{L}_{CE}(\hat{\mathbf{y}}, \mathbf{y}) + \frac{\alpha}{C} \sum_{k=1}^{C} \mathcal{L}_{CE}(\hat{\mathbf{y}}, \mathbb{1}), \tag{16}$$

where $\hat{\mathbf{y}}$ and $\mathbf{y}$ represent the predicted confidences and the ground truth, and $\mathbb{1}$ denotes an all one vector with $C$ entries. LS conducts sufficient experiments to analyze the influence of the smoothing factor $\alpha$ and reveals that larger values of $\alpha$ lead to greater vulnerability to MI attacks. In contrast to the standard training which adopts $\alpha$ greater than or equal to zero, LS encourages negative values of $\alpha$ to reduce the privacy threat.

The latest research AT-AT [152] challenges the prevailing view in the field of MI defenses, which holds that HSIC between private inputs and intermediate features renders models more vulnerable to training data privacy leakage. This work posits that adversarially non-robust features (*i.e.*, subtle discriminative signals imperceptible to the human eye that can flip classification predictions with minute pixel perturbations) constitute the root cause of privacy leakage in deep learning models. Existing defense methods achieve privacy protection only through unintentionally driving models to rely on non-robust features for classification, at the cost of a drastic degradation in model accuracy when exposed to minor adversarial perturbations. Accordingly, AT-AT proposes to deliberately suppress the model's reliance on robust features during training. Specifically, it appends an anti-adversarial loss as a regularization term on top of the standard clean-image classification loss, enabling a more favorable trade-off between classification accuracy and privacy preservation. The training objective can be expressed as:

$$\mathcal{L}_{AT-AT} = \mathcal{L}(f_\theta(x), y) + \lambda \mathcal{L}(f_\theta(x + \delta), y^*), \tag{17}$$

where $y$ is the ground truth label, $\mathcal{L}(\cdot, \cdot)$ denotes the main loss, $\delta$ is a targeted adversarial perturbation for a random wrong label $y^*$.

ResSFL [135] considers the split federated learning (SFL), where a central server might misuse the clients' uploaded intermediate features to reconstruct private data. A novel two-step framework is developed to deal with the MI threat. The first step adopts a publicly available dataset to pre-train a model with both high accuracy and strong MI resiliency. Similar to [27], this stage jointly trains a powerful inversion model as an adversary to achieve the MI robustness of the pre-trained model. Besides, ResSFL analyzes that higher dimensions of uploaded features indicate better MI vulnerability. Hence, they modify the client model structure to reduce the dimension of output features. In the second step, the

pre-trained model serves as the initial model for subsequent split learning. In addition to the regular main task loss, a weaker inversion model is incorporated to calculate the inversion loss that maximizes the distance between the private images and the images reconstructed by the inversion model. This allows for the optimization of the client-side model to achieve high accuracy while preserving MI resistance.

For language models, [137] reveals that language models trained with variants of the differential privacy-based optimizer (i.e, scaling up the batch size to millions to improve the utility of the DP-SGD step) exhibit more robust MI resistance while encountering the expense of performance degradation and increased computation. Moreover, [138] demonstrates the *memorization* of language models essentially has a positive connection to the model overfitting. Therefore, adequate regularization and early stopping are also served as reliable alternative defense strategies.

In graph learning, [21] considers differential privacy (DP) into the model training process by adding Gaussian noise to the clipped gradients in each training iteration for enhanced MI robustness. Similar to [25, 26], [67] proposes to diminish the correlation between the GNN's inputs and outputs by injecting stochasticity into adjacent matrix $\mathcal{A}$ and regularizing the mutual dependency among graph representations, adjacency, and predicted labels during the model training.

**Robust Fine-tuning.** As noted in [126], the above approaches to directly modify model training inevitably result in either a degradation in model utility or great efforts in hyperparameter tuning, given the model's sensitivity to minor alterations in hyperparameters. To solve this challenge, another research thread proposes to fine-tune the trained model instead of directly operating the model training phase.

A GAN-based method [136] discusses the potential of fine-tuning a trained model to intentionally fool an attacker into inverting insensitive public samples. Apart from the classifier $f_\theta(\cdot)$, the defender trains an additional public classifier $f_p(\cdot)$ and a GAN model $G_p(\cdot)$ using a publicly available dataset. Then, the defender uses the obtained GAN $G_p(\cdot)$ as image priors and conducts MI attacks on both the victim model $f_\theta(\cdot)$ and the public $f_p(\cdot)$ to reconstruct private and public images respectively. Afterwards, the defender fine-tunes the victim model with the generated samples, i.e., minimizing the classification loss of insensitive public samples while maximizing that of private ones. This operation injects misleading information into the victim model's predictions, hence deceiving the attacker into inverting samples that are significantly different from the private ones that should be protected.

Subsequently, TL [126] presents a straightforward and feasible strategy based on the transfer learning technique [153]. The Fisher information is first adopted to measure the influence of each intermediate layer on the ultimate MI attack performance and the clean classification accuracy. These findings indicate that the initial layers of a model exhibit a notable correlation with MI attacks, whereas the last several layers contribute more to the main task accuracy. Therefore, TL introduces a transfer learning mechanism since it typically fine-tunes the last few layers of a target model that are not relevant to MI attacks. Specifically, the algorithm involves pre-training the victim model on public datasets and then fine-tuning the last few layers on a private dataset, yielding excellent defense effects.

**Robust Model Architecture.** This type of defense strategy concentrates on designing a more robust model architecture to resist MI attacks. RoLSS [139] proposes pioneering work to analyze how DNN architectures affect MI attacks. They first demonstrate that skip connections facilitate MI attacks due to their positive effect on the gradient flow during the inversion process. Nevertheless, they also reveal that simply adopting a previous study, RepVGG [154], to convert a training-time multi-branch block (with skip connection) into an inference-time plain convolutional layer (without skip connection) performs poorly in resisting the MI attacks despite the removal of skip connections in the inference-time model architecture. Based on these findings and empirical study, they design novel model architecture techniques to improve MI robustness, including the removal of Last Stage Skip-Connection (RoLSS), Skip-Connection Scaling Factor (SSF), and Two-stage Training Scheme (TTS), respectively. Concretely, SSF introduces a scale factor $k \in [0, 1]$ to the skip connections of the last stage, controlling the range of gradients as follows:

$$z_{i+1} = block_i(z_i) + k \cdot z_i, \tag{18}$$

where $z_i$ and $z_{i+1}$ represent the input and output, $block_i$ indicates the intermediate block of the classifier (e.g., ResNet block [155]). In particular, the SSF method degrades to the RoLSS method when $k = 0$. TTS method first trains the model with full skip connections to obtain reasonable convergence of parameters and then deploys RoLSS to the training process to improve robustness.

LoFt [140] proposes a structural modification to the model by implementing a low-rank feature filtering strategy in the linear classification header. This approach compresses the input feature of the classification header into a lower-dimensional space, thereby reducing the information contained. After compression, the features are passed through a nonlinear activation function $Tanh$, which can introduce gradient vanishing issues. The final classification then relies solely on this compressed feature for decision-making. In summary, the low-rank compression strategy limits the amount of output information, and the gradient vanishing issue further hinders the attackers' optimization process, making it significantly harder to reconstruct private samples.

**Input Data Preprocessing.** These methods elaborately modify training data to enhance robust model learning. Trap-MID [141] defends against model inversion attacks by deliberately misleading the inversion process. Specifically, it injects a trigger into private training images, such that the reconstructed images are biased toward the trigger pattern instead of the genuine private features, thereby reducing privacy leakage.

[142] demonstrates that the data duplication strengthens the *memorization* of the victim language model, thereby increasing its risk of MI attacks. Therefore, training data de-duplication is an efficient approach to reduce the privacy breaches brought by MI attacks. In graph learning, [68] proves that ensuring DP cannot effectively prevent MI attacks and proposes an input processing technique, which conceals real edges by preprocessing the training graph, such as randomly flipping and rewriting. This operation tricks the attacker into reconstructing dummy data that significantly differs from the ground truth.

The main challenge of robust model learning lies in high deployment cost, since they typically require fine-tuning or retraining the target model with additional objectives, regularizers, or architectural modifications. Meanwhile, these interventions often interfere with the original optimization target, making them prone to substantial utility degradation and unstable performance in practice.

# 7 Social Impact

Given their effectiveness and practicality, MI attacks have become a significant privacy threat within the AI community. Case studies [23, 33] on production models (e.g., ChatGPT and Stable Diffusion) have intensified concerns about personal privacy and the urgent necessity for AI governance. Besides, the privacy risk is greatly amplified by the explosive development of AI generative content (AIGC), where numerous large models are trained on massive datasets that lack ownership verification. Currently, individuals may not be aware of which models have been trained on their personal data and cannot confirm whether model owners have adequately protected their privacy, which could be considered a continuous infringement of their rights. Therefore, model-level privacy protection and its interplay with official lawmaking are becoming a heated topic.

The study by [30] presents an interesting viewpoint: if a model is vulnerable to MI attacks, the model itself should also be regarded as a form of personal data, and correspondingly trigger a series of rights and obligations as defined by existing data protection laws (e.g., the EU General Data Protection Regulation). Nevertheless, it is a considerably complicated task to quantify a model's vulnerability to MI attacks, and directly categorizing models as personal data could disrupt the power relationship between model holders and individual data owners, which could have a detrimental impact on technical development. A more cautious approach for data users is to adopt robust and reliable defense strategies or establish access permissions to build responsible AI systems, thereby alleviating potential contradictions with individual data subjects. Besides, enacting new privacy legislation specific to model-level protection is indispensable, and introducing more regulatory policies is also strongly encouraged.

# 8 Conclusion and Research Directions

This survey presents an exhaustive review of the powerful MI attacks and defenses. First, we comprehensively categorize existing methods into several macroscopic taxonomies and present a systematic overview. We then briefly introduce early MI methods on traditional ML scenarios. Next, we characterize the mainstream attacks on DNNs from multiple perspectives, based on which we deeply illustrate their features and distinctions. Moreover, we investigate MI studies on more data modalities and provide detailed taxonomies. To facilitate the development of this field, we present a summary of several unsolved challenges and suggest promising directions for future research.

**Stronger Prior for MI attacks.** In visual tasks, previous approaches leveraged various GANs as image priors for reconstruction guidance. A notable contribution by [19] introduced diffusion models into MI attacks. They manually trained a conditional diffusion model [93] to generate $64 \times 64$ resolution images from the target distribution. To conquer the harder scenario of higher-resolution image recovery, future research could explore pre-trained diffusion models with rich image priors or alternative techniques to enhance generative model utilization.

**Defense From a Data Perspective.** Existing defenses for image data primarily operate on the model [25, 26] or its outputs [25], while little attention is paid to the training data. Therefore, one potential defense strategy is specially processing the data by applying well-designed masks or perturbations to create confusing reconstruction results [156]. Additionally, data synthesis [157] techniques have recently garnered significant attention. Since the synthesized images usually disguise the sensitive information, employing them in the training process emerges as a promising defensive approach against MI attacks.

**Certified Robustness for MI Defenses.** Despite existing studies on MI robustness, current defense approaches lack certified robustness guarantees. Certifiably robust approaches for DNNs have been developed against adversarial attacks [158] and backdoor attacks [159]. It is also imperative to establish a robust theoretical framework that can provide certified guarantees in the context of MI attacks. Additionally, exploring ways to integrate user feedback and preferences into the certification defense pipeline can enhance the practicality of ensuring certified robustness in conversational AI models like ChatGPT.

**Multi-Modal MI Attacks and Defenses.** Recently, advanced multi-modal models trained on cross-modal data pairs have shown sophisticated capabilities in handling complex and diverse data. While current MI methods have made great progress on unimodal models, there is a lack of research on multi-modal foundation models, e.g., the CLIP [160] and SAM [161]. Exploring potential vulnerabilities during the cross-modal inference process can be a promising and necessary direction for model inversion attacks and defenses.

**Model Inversion for Good.** Given the ability to recover diverse data from the target model's training set distribution, MI attacks can naturally be applied to various data-free learning scenarios [162, 163]. For instance, data reconstructed from a white-box teacher model can serve as a reliable dataset for training student models in data-free knowledge transfer and continual learning scenarios. Since advanced MI methodologies can generate data of higher quality, they are anticipated to enhance the efficacy and versatility of their application. In addition, future research could consider utilizing the synthesized data for a broader range of learning tasks and further improve their application performance.

# Declarations

**Conflict of Interest.** The authors declare that they have no conflict of interest. All authors confirm that there are no financial or non-financial interests, relationships, or affiliations (including but not limited to employment, consultancies, stock ownership, honoraria, grants, or other forms of compensation) with any organization or entity that could inappropriately influence or bias the work reported in this paper.

**Funding Declaration.** This work is supported in part by the National Natural Science Foundation of China under grant 62301189, 62576122, 62571298, and the Guangdong Basic and Applied Basic Research Foundation under grant 2026A1515011139.

**Data Availability.** The data supporting the visual comparison in Fig. 3 is from an openly available dataset FaceScrub [164], which can be obtained from its official website. Moreover, our open-source repository includes the code to generate the images shown in Fig. 3.

## Compliance with Ethical Standards

**Ethical Approval.** Not applicable. This study did not involve any experiments on human or animal participants, nor did it collect primary data requiring ethical review. Therefore, no ethical approval was required for this research. All data involved in this article were obtained from publicly available sources and were anonymized, ensuring no breach of privacy or confidentiality.

**Consent to Participate statement.** Not applicable. This review article is based exclusively on publicly available datasets and previously published literature. No human participants were involved in the conduct of this study, and therefore informed consent was not required.

## References


[1] Shen, D., Wu, G., Suk, H.-I.: Deep learning in medical image analysis. Annual review of biomedical engineering **19**(1), 221–248 (2017)

[2] Zhou, H.-Y., Lu, C., Chen, C., Yang, S., Yu, Y.: A unified visual information preservation framework for self-supervised pre-training in medical image analysis. IEEE Transactions on Pattern Analysis and Machine Intelligence **45**(7), 8020–8035 (2023)

[3] Shah, D., Campbell, W., Zulkernine, F.H.: A comparative study of lstm and dnn for stock market forecasting. In: 2018 IEEE International Conference on Big Data (big Data), pp. 4148–4155 (2018). IEEE

[4] Ozbayoglu, A.M., Gudelek, M.U., Sezer, O.B.: Deep learning for financial applications: A survey. Applied soft computing **93**, 106384 (2020)

[5] Zhang, S., Yao, L., Sun, A., Tay, Y.: Deep learning based recommender system: A survey and new perspectives. ACM computing surveys (CSUR) **52**(1), 1–38 (2019)

[6] Fredrikson, M., Lantz, E., Jha, S., Lin, S., Page, D., Ristenpart, T.: Privacy in pharmacogenetics: An {End-to-End} case study of personalized warfarin dosing. In: USENIX Security Symposium, pp. 17–32 (2014)

[7] Ganju, K., Wang, Q., Yang, W., Gunter, C.A., Borisov, N.: Property inference attacks on fully connected neural networks using permutation invariant representations. In: ACM SIGSAC Conference on Computer and Communications Security, pp. 619–633 (2018)

[8] Hu, H., Salcic, Z., Sun, L., Dobbie, G., Yu, P.S., Zhang, X.: Membership inference attacks on machine learning: A survey. ACM Computing Surveys (CSUR) **54**(11s), 1–37 (2022)

[9] Jegorova, M., Kaul, C., Mayor, C., O'Neil, A.Q., Weir, A., Murray-Smith, R., Tsaftaris, S.A.: Survey: Leakage and privacy at inference time. IEEE Transactions on Pattern Analysis and Machine Intelligence (2022)

[10] Wang, K.-C., Fu, Y., Li, K., Khisti, A., Zemel, R., Makhzani, A.: Variational model inversion attacks. In: Advances in Neural Information Processing Systems (2021)

[11] Fredrikson, M., Jha, S., Ristenpart, T.: Model inversion attacks that exploit confidence information and basic countermeasures. In: ACM SIGSAC Conference on Computer and Communications Security, pp. 1322–1333 (2015)

[12] Song, C., Ristenpart, T., Shmatikov, V.: Machine learning models that remember too much. In: ACM SIGSAC Conference on Computer and Communications Security, pp. 587–601 (2017)

[13] Yang, Z., Zhang, J., Chang, E.-C., Liang, Z.: Neural network inversion in adversarial setting via background knowledge alignment.

In: ACM SIGSAC Conference on Computer and Communications Security (2019)

[14] Zhang, Y., Jia, R., Pei, H., Wang, W., Li, B., Song, D.: The secret revealer: Generative model-inversion attacks against deep neural networks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (2020)

[15] Struppek, L., Hintersdorf, D., Correira, A.D.A., Adler, A., Kersting, K.: Plug & play attacks: Towards robust and flexible model inversion attacks. In: International Conference on Machine Learning (2022)

[16] Goodfellow, I.: Nips 2016 tutorial: Generative adversarial networks. arXiv preprint arXiv:1701.00160 (2016)

[17] Yuan, Z., Wu, F., Long, Y., Xiao, C., Li, B.: Secretgen: Privacy recovery on pre-trained models via distribution discrimination. In: European Conference on Computer Vision (2022)

[18] An, S., Tao, G., Xu, Q., Liu, Y., Shen, G., Yao, Y., Xu, J., Zhang, X.: Mirror: Model inversion for deep learning network with high fidelity. In: Network and Distributed System Security Symposium (2022)

[19] Liu, R., Wang, D., Ren, Y., Wang, Z., Guo, K., Qin, Q., Liu, X.: Unstoppable attack: Label-only model inversion via conditional diffusion model. IEEE Transactions on Information Forensics and Security **19**, 3958–3973 (2024)

[20] Carlini, N., Liu, C., Erlingsson, Ú., Kos, J., Song, D.: The secret sharer: Evaluating and testing unintended memorization in neural networks. In: USENIX Security Symposium (2019)

[21] Zhang, Z., Liu, Q., Huang, Z., Wang, H., Lu, C., Liu, C., Chen, E.: Graphmi: Extracting private graph data from graph neural networks. In: International Joint Conference on Artificial Intelligence (2021)

[22] Brown, T., Mann: Language models are few-shot learners. In: Advances in Neural Information Processing Systems, vol. 33, pp. 1877–1901 (2020)

[23] Nasr, M., Carlini, N., Hayase, J., Jagielski, M., Cooper, A.F., Ippolito, D., Choquette-Choo, C.A., Wallace, E., Tramèr, F., Lee, K.: Scalable extraction of training data from (production) language models. arXiv preprint arXiv:2311.17035 (2023)

[24] Wen, J., Yiu, S.-M., Hui, L.C.: Defending against model inversion attack by adversarial examples. In: 2021 IEEE International Conference on Cyber Security and Resilience (CSR), pp. 551–556 (2021). IEEE

[25] Wang, T., Zhang, Y., Jia, R.: Improving robustness to model inversion attacks via mutual information regularization. In: Proceedings of the AAAI Conference on Artificial Intelligence (2021)

[26] Peng, X., Liu, F., Zhang, J., Lan, L., Ye, J., Liu, T., Han, B.: Bilateral dependency optimization: Defending against model-inversion attacks. In: ACM SIGKDD Conference on Knowledge Discovery and Data Mining (2022)

[27] Yang, Z., Shao, B., Xuan, B., Chang, E.-C., Zhang, F.: Defending model inversion and membership inference attacks via prediction purification. arXiv preprint arXiv:2005.03915 (2020)

[28] Dibbo, S.V.: Sok: Model inversion attack landscape: Taxonomy, challenges, and future roadmap. In: 2023 IEEE 36th Computer Security Foundations Symposium (CSF), pp. 439–456 (2023). https://doi.org/10.1109/CSF57540.2023.00027

[29] Song, J., Namiot, D.: A survey of the implementations of model inversion attacks. In: International Conference on Distributed Computer and Communication Networks, pp. 3–16 (2022). Springer

[30] Veale, M., Binns, R., Edwards, L.: Algorithms that remember: model inversion

attacks and data protection law. Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences **376**(2133), 20180083 (2018)

[31] Parikh, R., Dupuy, C., Gupta, R.: Canary extraction in natural language understanding models. Annual Meeting of the Association for Computational Linguistics (ACL) (2022)

[32] Hidano, S., Murakami, T., Katsumata, S., Kiyomoto, S., Hanaoka, G.: Model inversion attacks for online prediction systems: Without knowledge of non-sensitive attributes. IEICE Transactions on Information and Systems **101**(11), 2665–2676 (2018)

[33] Carlini, N., Hayes, J., Nasr, M., Jagielski, M., Sehwag, V., Tramer, F., Balle, B., Ippolito, D., Wallace, E.: Extracting training data from diffusion models. In: USENIX Security Symposium, pp. 5253–5270 (2023)

[34] Yuan, X., Chen, K., Zhang, J., Zhang, W., Yu, N., Zhang, Y.: Pseudo label-guided model inversion attack via conditional generative adversarial network. In: Proceedings of the AAAI Conference on Artificial Intelligence (2023)

[35] Hitaj, B., Ateniese, G., Perez-Cruz, F.: Deep models under the gan: information leakage from collaborative deep learning. In: ACM SIGSAC Conference on Computer and Communications Security, pp. 603–618 (2017)

[36] Nguyen, N.-B., Chandrasegaran, K., Abdollahzadeh, M., Cheung, N.-M.: Re-thinking model inversion attacks against deep neural networks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 16384–16393 (2023)

[37] Qiu, Y., Fang, H., Yu, H., Chen, B., Qiu, M., Xia, S.-T.: A closer look at gan priors: Exploiting intermediate features for enhanced model inversion attacks. In: European Conference on Computer Vision (2024)

[38] Chen, S., Kahla, M., Jia, R., Qi, G.-J.: Knowledge-enriched distributional model inversion attacks. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (2021)

[39] Khosravy, M., Nakamura, K., Hirose, Y., Nitta, N., Babaguchi, N.: Model inversion attack by integration of deep generative models: Privacy-sensitive face generation from a face recognition system. IEEE Transactions on Information Forensics and Security **17**, 357–372 (2022) https://doi.org/10.1109/TIFS.2022.3140687

[40] Qi, G., Chen, Y., Mao, X., Hui, B., Li, X., Zhang, R., Xue, H.: Model inversion attack via dynamic memory learning. In: ACM International Conference on Multimedia, pp. 5614–5622 (2023)

[41] Kahla, M., Chen, S., Just, H.A., Jia, R.: Label-only model inversion attacks via boundary repulsion. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (2022)

[42] Xu, Y., Liu, X., Hu, T., Xin, B., Yang, R.: Sparse black-box inversion attack with limited information. In: IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (2023)

[43] Ye, Z., Luo, W., Naseem, M.L., Yang, X., Shi, Y., Jia, Y.: C2fmi: Corse-to-fine black-box model inversion attack. IEEE Transactions on Dependable and Secure Computing (2023)

[44] Zhao, X., Zhang, W., Xiao, X., Lim, B.: Exploiting explanations for model inversion attacks. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (2021)

[45] Nguyen, N.-B., Chandrasegaran, K., Abdollahzadeh, M., Cheung, N.-m.: Label-only model inversion attacks via knowledge transfer. In: Advances in Neural Information Processing Systems (2023)

[46] Zhou, S., Zhu, T., Ye, D., Yu, X., Zhou,

W.: Boosting model inversion attacks with adversarial examples. IEEE Transactions on Dependable and Secure Computing **21**(3), 1451–1468 (2023)

[47] Zhang, Z., Wang, X., Huang, J., Zhang, S.: Analysis and utilization of hidden information in model inversion attacks. IEEE Transactions on Information Forensics and Security (2023)

[48] Zhang, Z., Huang, J.: Aligning the domains in cross domain model inversion attack. Neural Networks **178**, 106490 (2024)

[49] Han, G., Choi, J., Lee, H., Kim, J.: Reinforcement learning-based black-box model inversion attacks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (2023)

[50] Liu, Y., Zhang, W., Wu, D., Lin, Z., Gu, J., Wang, W.: Prediction exposes your face: Black-box model inversion via prediction alignment. In: European Conference on Computer Vision, pp. 288–306 (2024). Springer

[51] Zhang, Z., Wen, J., Huang, M.: Ethicist: Targeted training data extraction through loss smoothed soft prompting and calibrated confidence estimation. In: Annual Meeting of the Association for Computational Linguistics (ACL) (2023)

[52] Pan, X., Zhang, M., Ji, S., Yang, M.: Privacy risks of general-purpose language models. In: 2020 IEEE Symposium on Security and Privacy (SP), pp. 1314–1331 (2020). IEEE

[53] Song, C., Raghunathan, A.: Information leakage in embedding models. In: ACM SIGSAC Conference on Computer and Communications Security, pp. 377–390 (2020)

[54] Elmahdy, A., Salem, A.: Deconstructing classifiers: Towards a data reconstruction attack against text classification models. In: Proceedings of the Fifth Workshop on Privacy in Natural Language Processing, pp. 143–158 (2024)

[55] Li, H., Xu, M., Song, Y.: Sentence embedding leaks more information than you expect: Generative embedding inversion attack to recover the whole sentence. In: Annual Meeting of the Association for Computational Linguistics (ACL), pp. 14022–14040 (2023)

[56] Morris, J.X., Zhao, W., Chiu, J.T., Shmatikov, V., Rush, A.M.: Language model inversion. In: The Twelfth International Conference on Learning Representations (ICLR) (2024)

[57] Carlini, N., Tramèr, F., Wallace, E., Jagielski, M., Herbert-Voss, A., Lee, K., Roberts, A., Brown, T., Song, D., Erlingsson, Ú., Oprea, A., Raffel, C.: Extracting training data from large language models. In: USENIX Security Symposium, pp. 2633–2650 (2021). https://www.usenix.org/conference/usenixsecurity21/presentation/carlini-extracting

[58] Elmahdy, A., Inan, H.A., Sim, R.: Privacy leakage in text classification a data extraction approach. In: Proceedings of the Fourth Workshop on Privacy in Natural Language Processing, pp. 13–20 (2022)

[59] Huang, J., Shao, H., Chang, K.C.-C.: Are large pre-trained language models leaking your personal information? In: Findings of the Association for Computational Linguistics: EMNLP 2022, pp. 2038–2047 (2022)

[60] Zhang, C., Morris, J.X., Shmatikov, V.: Extracting prompts by inverting llm outputs. In: Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, pp. 14753–14777 (2024)

[61] Yu, W., Pang, T., Liu, Q., Du, C., Kang, B., Huang, Y., Lin, M., Yan, S.: Bag of tricks for training data extraction from language models. In: International Conference on Machine Learning, pp. 40306–40320 (2023). PMLR

[62] Zanella-Béguelin, S., Wutschitz, L., Tople, S., Rühle, V., Paverd, A., Ohrimenko, O., Köpf, B., Brockschmidt, M.: Analyzing information leakage of updates to natural

language models. In: ACM SIGSAC Conference on Computer and Communications Security (2020)

[63] Panchendrarajan, R., Bhoi, S.: Dataset reconstruction attack against language models. In: CEUR Workshop (2021)

[64] Chen, Y., Lent, H., Bjerva, J.: Text embedding inversion security for multilingual language models. In: Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 7808–7827 (2024)

[65] Huang, Y.-H., Tsai, Y., Hsiao, H., Lin, H.-Y., Lin, S.-D.: Transferable embedding inversion attack: Uncovering privacy risks in text embeddings without model queries. In: Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 4193–4205 (2024)

[66] Zhang, Y., Carlini, N., Ippolito, D.: Effective prompt extraction from language models. In: First Conference on Language Modeling (2024)

[67] Zhou, Z., Zhou, C., Li, X., Yao, J., Yao, Q., Han, B.: On strengthening and defending graph reconstruction attack with markov chain approximation. In: International Conference on Machine Learning (2023)

[68] Zhang, Z., Liu, Q., Huang, Z., Wang, H., Lee, C.-K., Chen, E.: Model inversion attacks against graph neural networks. IEEE Transactions on Knowledge and Data Engineering **35**(9), 8729–8741 (2023) https://doi.org/10.1109/TransactionsonKnowledgeandDataEngineering.2022.3207915

[69] Duddu, V., Boutet, A., Shejwalkar, V.: Quantifying privacy leakage in graph embedding. In: International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services (MobiQuitous) (2020)

[70] Zhang, Z., Chen, M., Backes, M., Shen, Y., Zhang, Y.: Inference attacks against graph neural networks. In: USENIX Security Symposium (2022)

[71] He, X., Jia, J., Backes, M., Gong, N.Z., Zhang, Y.: Stealing links from graph neural networks. In: USENIX Security Symposium (2021)

[72] Wu, F., Long, Y., Zhang, C., Li, B.: Linkteller: Recovering private edges from graph neural networks via influence analysis. In: 2022 IEEE Symposium on Security and Privacy (2022). https://doi.org/10.1109/SP46214.2022.9833806

[73] Chen, Y., Ma, X., Zou, D., Jiang, Y.-G.: Extracting training data from unconditional diffusion models. arXiv preprint arXiv:2406.12752 (2024)

[74] Fang, H., Chen, B., Wang, X., Wang, Z., Xia, S.-T.: Gifd: A generative gradient inversion method with feature domain optimization. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 4967–4976 (2023)

[75] Fang, H., Kong, J., Yu, W., Chen, B., Li, J., Wu, H., Xia, S.-T., Xu, K.: One perturbation is enough: On generating universal adversarial perturbations against vision-language pre-training models. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 4090–4100 (2025)

[76] Yu, W., Fang, H., Chen, B., Sui, X., Chen, C., Wu, H., Xia, S.-T., Xu, K.: Gi-nas: Boosting gradient inversion attacks through adaptive neural architecture search. IEEE Transactions on Information Forensics and Security (2025)

[77] Le Roux, Q., Bourbao, E., Teglia, Y., Kallas, K.: A comprehensive survey on backdoor attacks and their defenses in face recognition systems. IEEE Access **12**, 47433–47468 (2024)

[78] Zhou, Z., Zhu, J., Yu, F., Li, X., Peng, X., Liu, T., Han, B.: Model inversion attacks: A

survey of approaches and countermeasures. arXiv preprint arXiv:2411.10023 (2024)

[79] Kamali, F., Wynne, H.: Pharmacogenetics of warfarin. Annual review of medicine **61**(1), 63–75 (2010)

[80] Wang, Y., Si, C., Wu, X.: Regression model fitting under differential privacy and model inversion attack. In: International Joint Conference on Artificial Intelligence (2015)

[81] Zhang, J., Zhang, Z., Xiao, X., Yang, Y., Winslett, M.: Functional mechanism: regression analysis under differential privacy. Proceedings of the VLDB Endowment **5**(11), 1364–1375 (2012)

[82] Wu, X., Fredrikson, M., Jha, S., Naughton, J.F.: A methodology for formalizing model-inversion attacks. In: 2016 IEEE 29th Computer Security Foundations Symposium (CSF), pp. 355–370 (2016). IEEE

[83] Dibbo, S.V., Breuer, A., Moore, J., Teti, M.: Improving robustness to model inversion attacks via sparse coding architectures. In: European Conference on Computer Vision, pp. 117–136 (2025). Springer

[84] Rozell, C.J., Johnson, D.H., Baraniuk, R.G., Olshausen, B.A.: Sparse coding via thresholding and local competition in neural circuits. Neural computation **20**(10), 2526–2563 (2008)

[85] Simonyan, K., Vedaldi, A., Zisserman, A.: Deep inside convolutional networks: Visualising image classification models and saliency maps. arXiv preprint arXiv:1312.6034 (2013)

[86] Bach, S., Binder, A., Montavon, G., Klauschen, F., Müller, K.-R., Samek, W.: On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. PloS one **10**(7), 0130140 (2015)

[87] Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation. In: Medical Image Computing and Computer-assisted intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18, pp. 234–241 (2015). Springer

[88] Guo, C., Gardner, J., You, Y., Wilson, A.G., Weinberger, K.: Simple black-box adversarial attacks. In: International Conference on Machine Learning, pp. 2484–2493 (2019). PMLR

[89] Peng, X., Han, B., Liu, F., Liu, T., Zhou, M.: Pseudo-private data guided model inversion attacks. Advances in Neural Information Processing Systems **37**, 33338–33375 (2024)

[90] Peng, X., Han, B., Yu, F., Liu, T., Liu, F., Zhou, M.: Generative model inversion through the lens of the manifold hypothesis. In: The Thirty-ninth Annual Conference on Neural Information Processing Systems (2025)

[91] Li, Z., Zhang, H., Wang, J., Chen, M., Hu, H., Yi, W., Xu, X., Yang, M., Ma, C.: From head to tail: efficient black-box model inversion attack via long-tailed learning. In: Proceedings of the Computer Vision and Pattern Recognition Conference, pp. 29288–29298 (2025)

[92] Beyer, H.-G., Schwefel, H.-P.: Evolution strategies–a comprehensive introduction. Natural computing **1**(1), 3–52 (2002)

[93] Ho, J., Salimans, T.: Classifier-free diffusion guidance. arXiv preprint arXiv:2207.12598 (2022)

[94] Zhang, R., Isola, P., Efros, A.A., Shechtman, E., Wang, O.: The unreasonable effectiveness of deep features as a perceptual metric. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (2018)

[95] Karras, T., Laine, S., Aila, T.: A style-based generator architecture for generative adversarial networks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (2019)

[96] Rosca, M., Lakshminarayanan, B., Warde-Farley, D., Mohamed, S.: Variational approaches for auto-encoding generative adversarial networks. arXiv preprint arXiv:1706.04987 (2017)

[97] Kingma, D.P., Dhariwal, P.: Glow: Generative flow with invertible 1x1 convolutions. In: Bengio, S., Wallach, H., Larochelle, H., Grauman, K., Cesa-Bianchi, N., Garnett, R. (eds.) Advances in Neural Information Processing Systems, vol. 31. Curran Associates, Inc. (2018). https://proceedings.neurips.cc/paper_files/paper/2018/file/d139db6a236200b21cc7f752979132d0-Paper.pdf

[98] Haarnoja, T., Zhou, A., Abbeel, P., Levine, S.: Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In: International Conference on Machine Learning, pp. 1861–1870 (2018). PMLR

[99] Sheikh, H.R., Sabir, M.F., Bovik, A.C.: A statistical evaluation of recent full reference image quality assessment algorithms. IEEE Transactions on image processing **15**(11), 3440–3451 (2006)

[100] Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., Hochreiter, S.: Gans trained by a two time-scale update rule converge to a local nash equilibrium. In: Advances in Neural Information Processing Systems (2017)

[101] Kynkäänniemi, T., Karras, T., Laine, S., Lehtinen, J., Aila, T.: Improved precision and recall metric for assessing generative models. Advances in neural information processing systems **32** (2019)

[102] Naeem, M.F., Oh, S.J., Uh, Y., Choi, Y., Yoo, J.: Reliable fidelity and diversity metrics for generative models. In: International Conference on Machine Learning, pp. 7176–7185 (2020). PMLR

[103] Xu, C., Ren, J., Zhang, D., Zhang, Y., Qin, Z., Ren, K.: Ganobfuscator: Mitigating information leakage under gan via differential privacy. IEEE Transactions on Information Forensics and Security **14**(9), 2358–2371 (2019)

[104] Arjovsky, M., Chintala, S., Bottou, L.: Wasserstein generative adversarial networks. In: International Conference on Machine Learning, pp. 214–223 (2017). PMLR

[105] Croitoru, F.-A., Hondru, V., Ionescu, R.T., Shah, M.: Diffusion models in vision: A survey. IEEE Transactions on Pattern Analysis and Machine Intelligence **45**(9), 10850–10869 (2023)

[106] Rombach, R., Blattmann, A., Lorenz, D., Esser, P., Ommer, B.: High-resolution image synthesis with latent diffusion models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 10684–10695 (2022)

[107] Saharia, C., Chan, W., Saxena, S., Li, L., Whang, J., Denton, E.L., Ghasemipour, K., Gontijo Lopes, R., Karagol Ayan, B., Salimans, T., *et al.*: Photorealistic text-to-image diffusion models with deep language understanding. Advances in neural information processing systems **35**, 36479–36494 (2022)

[108] Gu, X., Du, C., Pang, T., Li, C., Lin, M., Wang, Y.: On memorization in diffusion models. Transactions on Machine Learning Research (2023)

[109] Devlin, J., Chang, M.-W., Lee, K., Toutanova, K.: BERT: Pre-training of deep bidirectional transformers for language understanding. In: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers) (2019)

[110] Xia, F., Sun, K., Yu, S., Aziz, A., Wan, L., Pan, S., Liu, H.: Graph learning: A survey. IEEE Transactions on Artificial Intelligence **2**(2), 109–127 (2021)

[111] Rong, Y., Xu, T., Huang, J., Huang, W.,

Cheng, H., Ma, Y., Wang, Y., Derr, T., Wu, L., Ma, T.: Deep graph learning: Foundations, advances and applications. In: Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, pp. 3555–3556 (2020)

[112] Qiao, L., Zhang, L., Chen, S., Shen, D.: Data-driven graph construction and graph learning: A review. Neurocomputing **312**, 336–351 (2018)

[113] Chami, I., Abu-El-Haija, S., Perozzi, B., Ré, C., Murphy, K.: Machine learning on graphs: A model and comprehensive taxonomy. Journal of Machine Learning Research **23**(89), 1–64 (2022)

[114] Aggarwal, C.C., Wang, H.: Graph data management and mining: A survey of algorithms and applications. Managing and mining graph data, 13–68 (2010)

[115] Wu, Z., Pan, S., Chen, F., Long, G., Zhang, C., Philip, S.Y.: A comprehensive survey on graph neural networks. IEEE Transactions on Neural Networks and Learning Systems **32**(1), 4–24 (2020)

[116] Dai, E., Zhao, T., Zhu, H., Xu, J., Guo, Z., Liu, H., Tang, J., Wang, S.: A comprehensive survey on trustworthy graph neural networks: Privacy, robustness, fairness, and explainability. Machine Intelligence Research **21**(6), 1011–1061 (2024)

[117] Scarselli, F., Gori, M., Tsoi, A.C., Hagenbuchner, M., Monfardini, G.: The graph neural network model. IEEE transactions on neural networks **20**(1), 61–80 (2008)

[118] Gupta, A., Matta, P., Pant, B.: Graph neural network: Current state of art, challenges and applications. Materials Today: Proceedings **46**, 10927–10932 (2021)

[119] Xie, Y., Xu, Z., Zhang, J., Wang, Z., Ji, S.: Self-supervised learning of graph neural networks: A unified review. IEEE Transactions on Pattern Analysis and Machine Intelligence **45**(2), 2412–2429 (2022)

[120] Zhang, X.S., Guan, W., Lu, J., Qiu, Z., Cheng, J., Wu, X., Zheng, Y.: Graphleak: Patient record leakage through gradients with knowledge graph. In: Proceedings of the ACM Web Conference 2024, pp. 4706–4716 (2024)

[121] Kaelbling, L.P., Littman, M.L., Moore, A.W.: Reinforcement learning: A survey. Journal of artificial intelligence research **4**, 237–285 (1996)

[122] Li, Y.: Deep reinforcement learning: An overview. arXiv preprint arXiv:1701.07274 (2017)

[123] Wiering, M.A., Van Otterlo, M.: Reinforcement learning. Adaptation, learning, and optimization **12**(3), 729 (2012)

[124] Arulkumaran, K., Deisenroth, M.P., Brundage, M., Bharath, A.A.: Deep reinforcement learning: A brief survey. IEEE Signal Processing Magazine **34**(6), 26–38 (2017)

[125] Ernst, D., Louette, A.: Introduction to reinforcement learning. Feuerriegel, S., Hartmann, J., Janiesch, C., and Zschech, P.(2024). Generative ai. Business & Information Systems Engineering **66**(1), 111–126 (2024)

[126] Ho, S.-T., Hao, K.J., Chandrasegaran, K., Nguyen, N.-B., Cheung, N.-M.: Model inversion robustness: Can transfer learning help? In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 12183–12193 (2024)

[127] Ye, D., Shen, S., Zhu, T., Liu, B., Zhou, W.: One parameter defense—defending against data inference attacks via differential privacy. IEEE Transactions on Information Forensics and Security **17**, 1466–1480 (2022)

[128] Titcombe, T., Hall, A.J., Papadopoulos, P., Romanini, D.: Practical defences against model inversion attacks for split neural networks. arXiv preprint arXiv:2104.05743 (2021)

[129] Olatunji, I.E., Rathee, M., Funke, T., Khosla, M.: Private graph extraction via feature explanations. Proceedings on Privacy Enhancing Technologies **2**, 59–78 (2023)

[130] Yang, Z., Wang, L., Yang, D., Wan, J., Zhao, Z., Chang, E.-C., Zhang, F., Ren, K.: Purifier: Defending data inference attacks via transforming confidence scores. In: Proceedings of the AAAI Conference on Artificial Intelligence, pp. 10871–10879 (2023)

[131] Zhuang, T., Yu, H., Qiu, Y., Fang, H., Chen, B., Xia, S.-T.: Stealthy shield defense: A conditional mutual information-based approach against black-box model inversion attacks. In: The Thirteenth International Conference on Learning Representations (2025)

[132] Wang, S., Ji, Z., Xiang, L., Zhang, H., Wang, X., Zhou, C., Li, B.: Crafter: Facial feature crafting against inversion-based identity theft on deep models. In: Network and Distributed System Security (NDSS) Symposium (2024)

[133] Peng, X., Liu, F., Wang, N., Lan, L., Liu, T., Cheung, Y.-M., Han, B.: Unknown-aware bilateral dependency optimization for defending against model inversion attacks. IEEE Transactions on Pattern Analysis and Machine Intelligence (2025)

[134] Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., Wojna, Z.: Rethinking the inception architecture for computer vision. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2818–2826 (2016)

[135] Li, J., Rakin, A.S., Chen, X., He, Z., Fan, D., Chakrabarti, C.: Ressfl: A resistance transfer framework for defending model inversion attack in split federated learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (2022)

[136] Gong, X., Wang, Z., Li, S., Chen, Y., Wang, Q.: A gan-based defense framework against model inversion attacks. IEEE Transactions on Information Forensics and Security (2023)

[137] Anil, R., Ghazi, B., Gupta, V., Kumar, R., Manurangsi, P.: Large-scale differentially private bert. In: Findings of the Association for Computational Linguistics: EMNLP 2022, pp. 6481–6491 (2022)

[138] Ishihara, S.: Training data extraction from pre-trained language models: A survey. In: Proceedings of the 3rd Workshop on Trustworthy Natural Language Processing (TrustNLP 2023), pp. 260–275 (2023)

[139] Hao, K.J., Ho, S.-T., Nguyen, N.-B., Cheung, N.-M.: On the vulnerability of skip connections to model inversion attacks. In: European Conference on Computer Vision, pp. 140–157 (2024). Springer

[140] Yu, H., Qiu, Y., Fang, H., Zhuang, T., Chen, B., Yu, S., Wang, B., Xia, S.-T., Xu, K.: Rank matters: Understanding and defending model inversion attacks via low-rank feature filtering. In: Proceedings of the 32nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining (2026)

[141] Liu, Z.-T., Chen, S.-T.: Trap-mid: Trapdoor-based defense against model inversion attacks. Advances in Neural Information Processing Systems **37**, 88486–88526 (2024)

[142] Kandpal, N., Wallace, E., Raffel, C.: Deduplicating training data mitigates privacy risks in language models. In: International Conference on Machine Learning (2022)

[143] Dwork, C.: Differential privacy. In: International Colloquium on Automata, Languages, and Programming, pp. 1–12 (2006). Springer

[144] Dwork, C.: Differential privacy: A survey of results. In: International Conference on Theory and Applications of Models of Computation, pp. 1–19 (2008). Springer

[145] Abadi, M., Chu, A., Goodfellow, I., McMahan, H.B., Mironov, I., Talwar, K., Zhang, L.: Deep learning with differential privacy. In: Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, pp. 308–318 (2016)

[146] Ji, Z., Lipton, Z.C., Elkan, C.: Differential privacy and machine learning: a survey and review. arXiv preprint arXiv:1412.7584 (2014)

[147] Ha, T., Dang, T.K., Dang, T.T., Truong, T.A., Nguyen, M.T.: Differential privacy in deep learning: an overview. In: 2019 International Conference on Advanced Computing and Applications (ACOMP), pp. 97–102 (2019). IEEE

[148] Kolchinsky, A., Tracey, B.D., Wolpert, D.H.: Nonlinear information bottleneck. Entropy **21**(12), 1181 (2019)

[149] Gretton, A., Herbrich, R., Smola, A., Bousquet, O., Schölkopf, B., Hyvärinen, A.: Kernel methods for measuring independence. Journal of Machine Learning Research **6**(12) (2005)

[150] Gretton, A., Bousquet, O., Smola, A., Schölkopf, B.: Measuring statistical dependence with hilbert-schmidt norms. In: International Conference on Algorithmic Learning Theory, pp. 63–77 (2005). Springer

[151] Struppek, L., Hintersdorf, D., Kersting, K.: Be careful what you smooth for: Label smoothing can be a privacy shield but also a catalyst for model inversion attacks. In: International Conference on Learning Representations (2024). https://openreview.net/forum?id=1SbkubNdbW

[152] Torp, R., Smith, S., Breuer, A.: Reducing information dependency does not cause training data privacy. adversarially non-robust features do. In: The Fourteenth International Conference on Learning Representations (2026)

[153] Pan, S., Yang, Q.: A survey on transfer learning. IEEE Transaction on Knowledge Discovery and Data Engineering, 22 (10). IEEE press (2010)

[154] Ding, X., Zhang, X., Ma, N., Han, J., Ding, G., Sun, J.: Repvgg: Making vgg-style convnets great again. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 13733–13742 (2021)

[155] He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778 (2016)

[156] Tan, Q., Li, Q., Zhao, Y., Liu, Z., Guo, X., Xu, K.: Defending against data reconstruction attacks in federated learning: An information theory approach. In: 33rd USENIX Security Symposium (USENIX Security 24), pp. 325–342 (2024)

[157] Lei, S., Tao, D.: A comprehensive survey of dataset distillation. IEEE Transactions on Pattern Analysis and Machine Intelligence (2023)

[158] Cohen, J., Rosenfeld, E., Kolter, Z.: Certified adversarial robustness via randomized smoothing. In: International Conference on Machine Learning (2019)

[159] Wang, B., Cao, X., Gong, N.Z., et al.: On certifying robustness against backdoor attacks via randomized smoothing. arXiv preprint arXiv:2002.11750 (2020)

[160] Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., *et al.*: Learning transferable visual models from natural language supervision. In: International Conference on Machine Learning (2021)

[161] Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., Berg, A.C., Lo, W.-Y., *et al.*: Segment anything. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 4015–4026 (2023)

[162] Yin, H., Molchanov, P., Alvarez, J.M., Li, Z., Mallya, A., Hoiem, D., Jha, N.K., Kautz, J.: Dreaming to distill: Data-free knowledge transfer via deepinversion. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 8715–8724 (2020)

[163] Fang, G., Song, J., Wang, X., Shen, C., Wang, X., Song, M.: Contrastive model inversion for data-free knowledge distillation. In: International Joint Conference on Artificial Intelligence (2021)

[164] Ng, H.-W., Winkler, S.: A data-driven approach to cleaning large face datasets. In: 2014 IEEE International Conference on Image Processing (ICIP), pp. 343–347 (2014). IEEE

[165] Nazir, M., Finlayson, M., Morris, J., Ren, X., Swayamdipta, S.: Better language model inversion by compactly representing next-token distributions. In: Advances in Neural Information Processing Systems **38**, 8636–8659 (2026)