Membership Inference Attack with Partial Features

Xurun Wang
Harbin Institute of Technology
wangxr@stu.hit.edu.cn

Guangrui Liu
Harbin Institute of Technology
grliu@hit.edu.cn

Xinjie Li
Harbin Institute of Technology
23B903086@stu.hit.edu.cn

Haoyu He
Monash University
haoyu.he@monash.edu

Lin Yao
Dalian University of Technology
yaolin@dlut.edu.cn

Weizhe Zhang
Harbin Institute of Technology
wzzhang@hit.edu.cn

***Abstract*—Machine learning models have been shown to be susceptible to membership inference attack, which can be used to determine whether a given sample appears in the training data. Existing membership inference methods commonly assume that the adversary has full access to the features of the target sample. This assumption, however, does not hold in many real-world scenarios where only partial features information is available, thereby limiting the applicability of these methods. In this work, we study an inference scenario where the adversary observes only partial features of each sample and aims to infer whether this observed subset was present in the training set of the target model. We define this problem as Partial Feature Membership Inference (PFMI). To address this problem, we propose MRAD (Memory-guided Reconstruction and Anomaly Detection), a two-stage attack framework. In the first stage, MRAD optimizes the unknown feature values to minimize the loss of the sample. In the second stage, it measures the deviation between the reconstructed sample and the training distribution using anomaly detection. Empirical results demonstrate that MRAD is effective across a range of datasets, and maintains compatibility with various off-the-shelf anomaly detection techniques. For example, on STL-10, our attack achieves an AUC of around 0.6 even with 40% of the missing features.**

I. Introduction

Machine learning has been widely applied across various domains, such as medical diagnosis [1], traffic analysis [2], and autonomous driving [3]. However, current machine learning models remain vulnerable to privacy inference attacks. Adversaries can exploit interactions with the target model to infer privacy information from its training data, posing significant threats to user confidentiality and data privacy. A well-known example of such attacks is the Membership Inference Attack (MIA) [4], [5], which can be used to determine whether certain samples were included in the training set of a target model. This type of attack highlights the risk of information leakage during inference, where adversaries can extract sensitive training information at a low cost.

Membership inference attack has been shown to compromise the privacy of various machine learning models [6]–[9]. In this paper, we focus on this attack against classification models. Existing membership inference attack can be roughly categorized into two types based on the attacker's objective. The first type, exact-match MIA, aims to determine whether a given sample exactly matches one of the samples in the training set of the target model, which is where most existing attacks fall [10]–[12]. These can be further divided into confidence-based attacks [13], [14], label-only attacks [10], [15], and blind attacks [16]. The second type, range-match MIA, does not seek an exact sample match. Instead, it aims to infer whether the model's training set contains any sample with features in a specified range. In recent work, Tao et al. [17] were the first to formally define and investigate this setting, highlighting that membership inference can be extended beyond exact-match assumptions.

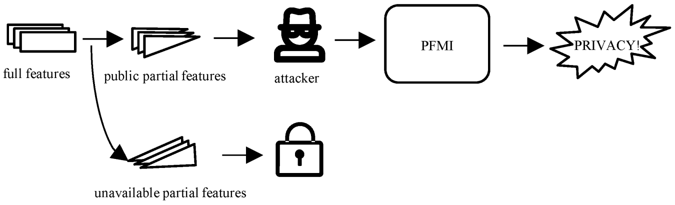


Fig. 1. Our goal is to infer membership information when only partial features of a sample are available.

Inspired by Tao et al. [17], we pose a question: *How much private information can membership inference attack extract when some features are missing?* This setting aligns more closely with practical scenarios. For instance, consider an attacker attempting to determine whether an individual, Alice, has a particular disease by querying a diagnostic system deployed in a hospital she frequently visits. If Alice is indeed diagnosed with the disease, it is likely that her data is included in the training set. However, due to the difficulty of acquiring complete information, the attacker may only know a subset of her attributes, such as age, gender, and race. Applying a standard membership inference attack in this setting would require exhaustively enumerating all possible combinations of the unknown features, which is computationally infeasible and inefficient.

As Figure 1 shows, in this paper, we consider a setting where the attacker has access to only a subset of a sample's features and aims to infer whether this partial feature combination was present in the training set of a target model. We refer to this type of attack as ***P**artial **F**eature **M**embership **I**nference* (PFMI). In this setting, the observed partial features is termed known features, while the remaining unobserved ones are referred to as unknown features. The property existence attack, proposed as a generalization of membership inference [18], shares certain similarities with PFMI in that both seek to infer the presence of specific feature combinations. However, in property existence attack, the adversary can directly access complete samples containing the target feature combination [18], [19], whereas PFMI operates under stricter constraints by relying solely on partial observations. Although the attacker's limited access to only partial feature information makes the inference task more challenging, this constraint also reflects a more realistic threat model. It shows that even with incomplete information, a resource-constrained adversary can still launch effective privacy inference attack. This amplifies the risks posed to current privacy protection mechanisms, which often assume attackers have access to complete samples.

To tackle this problem, we propose **MRAD**—a two-stage attack framework that combines Memory-guided Reconstruction with Anomaly Detection. MRAD is specifically tailored to the partial feature setting and is capable of accurately inferring membership even when some input features are missing. Specifically, when we try to minimize the loss of the target sample by updating unknown features, reconstructed samples with non-member features often deviate significantly from the original distribution, resulting in anomalous data. We leverage anomaly detection techniques to identify such deviations, thereby determining whether the observed partial feature combination exists in the training set. Our main contributions can be summarized as follows.

- **Attack Formalization under Partial Observability**: We investigate a more realistic setting for membership inference attack, in which the adversary has access only to partial features of the target sample. We refer to this scenario as PFMI, and we provide a formal definition of the attack framework under this constraint.
- **Two-Stage Attack Framework**: We propose a two-stage attack framework designed for this new inference scenario. The framework is modular and can be integrated with a variety of off-the-shelf anomaly detection techniques.
- **Extensive Evaluation**: We extensively evaluate our attack framework on both image and tabular datasets, demonstrating its compatibility with various anomaly detection methods. In the STL-10 and Epsilon datasets, our method achieves an average AUC of 0.6 even with 40% missing features.
- **In-depth Analysis & Case Study**: We show how hyperparameter settings and the importance of observed features affect attack performance by experiments. We also present a case study to demonstrate how our attack can lead to privacy leakage in a real world scenario.

The remainder of this paper is organized as follows. Section II reviews related work on membership inference and anomaly detection. Section III introduces the PFMI setting, including the attack objective, adversary capability and attack definition. Section IV presents the technical details of the proposed two-stage attack framework. Section V presents experimental results, including studies that validate design intuition, followed by a comprehensive evaluation of the attack performance in various settings. Section VI discusses potential defense mechanisms, analyzes the relationship between PFMI and other inference attacks, and outlines directions for future work. Section VII concludes the paper.

II. Related Work

In this section, we review prior work on membership inference attack. Then we will give a brief introduction on anomaly detection.

A. Membership Inference Attack

Membership Inference Attack Based on Exact Sample Matching: Membership inference attack against machine learning models was first introduced by Shokri et al. [4]. A binary classifier was trained as the attack model using prediction vectors produced by shadow models. To launch the attack, the target model's prediction on a given input is provided to the attack model, which decides whether the input was included in the training data. Most follow-up studies adopt a similar setting, where the adversary has access to both the full prediction vector and the ground-truth label of the sample. These methods typically rely on the observation that member samples tend to yield lower loss values than non-members [5], [12], [13]. Song et al. [13] predict membership with a modified loss function. Ye et al. [14] proposed a membership inference attack that achieves an arbitrary false positive rate by using a non-member set to calibrate the decision threshold. Carlini et al. [5] trained multiple shadow models to examine the loss distributions of samples when they are in and out of the training set, further improving the attack performance under low false positive rates. Instead of using loss values for membership inference, Zarifzadeh et al. [11] computed a likelihood ratio based on the output confidences and used it as the inference metric.

In the label-only scenario, the adversary can only observe the predicted label rather than the full confidence vector [15], [20]. Li et al. [15] leveraged the fact that member samples are often farther from the decision boundary, while Peng et al. [10] crafted minimally perturbed adversarial examples using a shadow model and inferred membership based on whether the perturbed samples were misclassified by the target model. Since their method relies solely on the final prediction of the model, it requires only a single query.

Another line of work, which is called blind membership inference, considers the setting where the adversary has access to the full prediction vector of the target sample but lacks knowledge of the true label. Hui et al. [16] proposed a blind black-box attack that performs differential comparisons between samples to infer membership without ground-truth guidance.

Membership Inference Attack Based on Feature Range Matching: Tao et al. [17] proposed a novel form of membership inference attack that shifts the focus from individual

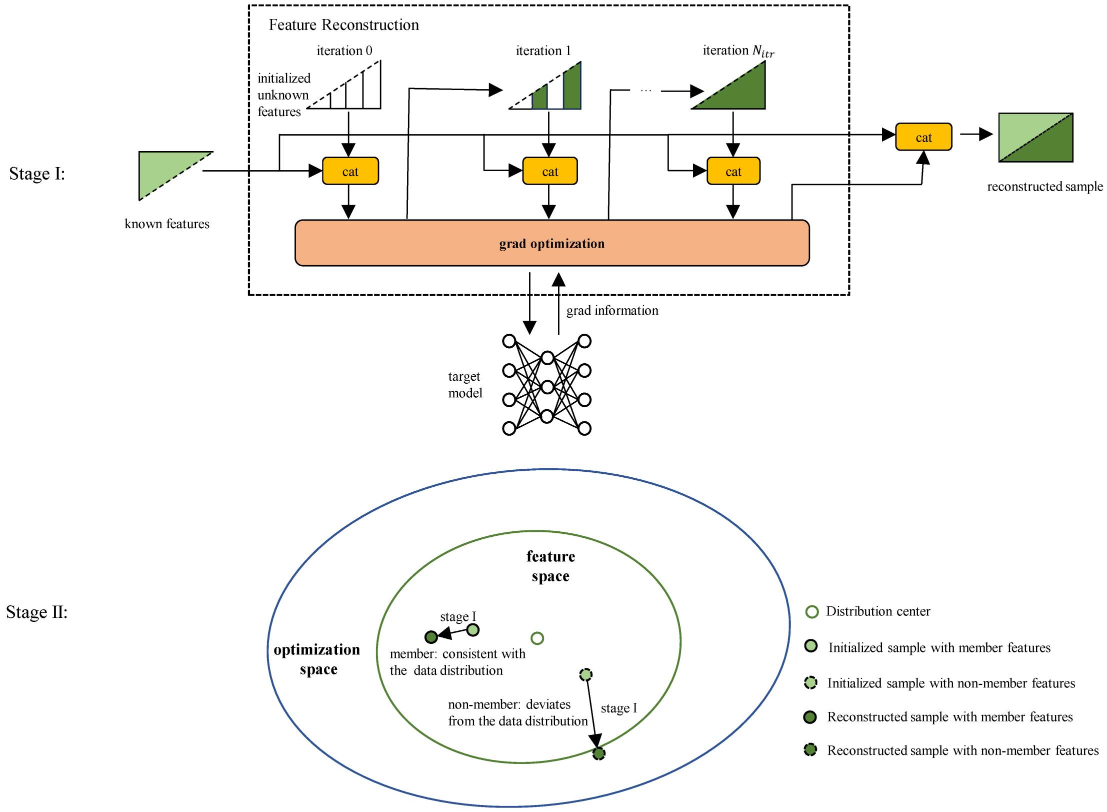

Feature Reconstruction
iteration 0
iteration 1
iteration N_{itr}
initialized unknown features
Stage I:
known features
cat
cat
cat
cat
reconstructed sample
grad optimization
grad information
target model
Stage II:
optimization space
feature space
stage I
member: consistent with the data distribution
non-member: deviates from the data distribution
stage I
Distribution center
Initialized sample with member features
Initialized sample with non-member features
Reconstructed sample with member features
Reconstructed sample with non-member features

Fig. 2. Attack framework:We first implement a simple feature reconstruction algorithm to obtain a complete sample, and then distinguish between member and non-member features based on their deviation distance.

samples to feature regions. Rather than determining whether a specific sample was included in the training set, their method aims to infer whether any data point within a given range appears in the training data. To perform the attack, they first sample instances from the specified range, apply standard membership inference techniques to assign scores to these samples, and then aggregate the scores to reach a final inference decision.

B. Anomaly Detection

Anomalous data refers to samples that deviate from the expected or learned data distribution. Two common categories of such anomalies are concept drift and out-of-distribution (OOD) samples. Concept drift occurs when the test-time data distribution shifts from the training distribution, leading to degraded model performance [21]. Yang et al. [22] detect such drift by modeling input data with autoencoders and computing Median Absolute Deviation(MAD) values between the current and original distributions. Wan et al. [23] proposed MCD-DD, which uses a dynamically updated encoder and Mahalanobis distance-based criteria to detect distributional changes in time series. OOD detection has also received significant attention, as the presence of OOD samples can bias model behavior. Park et al. [24] improved OOD detection by reducing overconfidence in OOD regions. Reiss et al. [25] designed a novel contrastive loss function that enhances contrastive learning for OOD detection. Liu et al. [26] exploited the clustering effect in the penultimate layer of neural networks to develop a post-hoc OOD detection method.

III. Problem Formulation

In this section, we present the problem formulation of the proposed Partial Feature Membership Inference (PFMI) attack. We begin by describing the attack model , followed by a formal definition of the attack.

A. Attack Model

Attack Objective: In this work, the goal of the attacker is similar to that of traditional MIA, but with a key constraint: the attacker only has access to partial features of the target sample. The objective is to determine whether this particular feature combination exists in the training set of the target model. We refer to such feature combinations as member features if they appear in the training set, and non-member features otherwise. Apparently, any combination drawn from the training set is considered as member features.

Adversary Capability: We consider a white-box threat model in which the attacker has full access to the model, including its output, training algorithm, internal parameters, architecture, and gradients. Although the specific target sample may have missing features due to privacy concerns or incomplete collection, we assume that the attacker has knowledge of the training data distribution and can sample fully observed auxiliary data from it to support the attack, which is consistent with the prior inference attack settings [4], [11], [13], [14]. This is feasible in practice, as attackers may obtain partial data through social engineering or cyberattacks. However, the attacker is unable to obtain any complete sample that contains the exact feature combination to be inferred.

B. Attack Definition

Before introducing our proposed attack definition, we first revisit the standard definition of membership inference attack, as formalized in prior work [11], [27] using an indistinguishability game.

Definition 1: (Membership Inference Attack). Let π represent the underlying data distribution, θ the parameters of the target model, $\mathcal{T}$ the training algorithm, and $\mathcal{A}$ the attack algorithm.

I The Challenger samples a dataset $\mathcal{D} \sim \pi$ to train a model $\theta \leftarrow \mathcal{T}(\mathcal{D})$

II The Challenger flips a fair coin b, if $b = 1$, then $x \sim \mathcal{D}$, otherwise $x \sim \pi/\mathcal{D}$. The Challenger sends the model θ and the target sample x to the Adversary

III The Adversary gets the prediction $\hat{b} \leftarrow \mathcal{A}(\theta, x \odot M, \pi)$

IV If $\hat{b} = b$, output 1. Otherwise, output 0.

Previous membership inference attack generally assume that the adversary has access to complete feature information of a sample. In contrast, we consider a more realistic and constrained setting where only partial feature values are available. To formalize this, we introduce a binary mask M that indicates which features are known and which are unknown. Using this mask-based representation, we extend the standard indistinguishability framework and define the Partial Feature Membership Inference (PFMI) problem.

Definition 2: (Partial Feature Membership Inference). Let π represent the underlying data distribution, θ the parameters of the target model, $\mathcal{T}$ the training algorithm, and $\mathcal{A}$ the attack algorithm. We introduce a feature mask $M \in \{0, 1\}^d$, where $M_i = 1$ indicates that the i-th feature is known to the attacker, and $M_i = 0$ indicates it is unknown. We use $\odot$ to denote element-wise product.

I The Challenger samples a dataset $\mathcal{D} \sim \pi$ to train a model $\theta \leftarrow \mathcal{T}(\mathcal{D})$

II The Challenger flips a fair coin b, if $b = 1$, then $x \sim \mathcal{D}$, otherwise $x \sim \pi/\mathcal{D}$ such that $x \odot M \notin \mathcal{D} \odot M$.

III The Challenger gets the known features $x_{kno} \leftarrow x \odot M$. Let $\pi' \leftarrow \pi/\{x' : x' \odot M = x_{kno}\}$, then the Challenger sends (θ, x_{kno}, π') to the Adversary.

IV The Adversary gets the prediction $\hat{b} \leftarrow \mathcal{A}(\theta, x \odot M, \pi)$

V If $\hat{b} = b$, output 1. Otherwise, output 0.

In this paper, for the sake of generality, we define partial features as a subset of the full feature vector, where the known features can appear at arbitrary positions. That is, the known and unkonwn features are not required to form a contiguous block or follow any fixed pattern. This flexible definition better captures real-world scenarios where feature availability is influenced by privacy constraints or unstable data sources.

IV. MRAD: Memory-Guided Feature Reconstruction & Anomaly Detection

In this section, we first introduce the underlying design intuition of our attack, and then detail the implementation process of the MRAD framework.

A. Design Intuition

In designing our attack, we build on a well-established observation: overfitted models memorize their training data, causing training members to occupy regions of the input space associated with low loss values [4], [28], [29]. Specifically, for a supervised machine learning algorithm $\mathcal{T}$, the training process aims to learn a model that can effectively distinguish between samples from different classes. Since it is impractical to access the full data distribution, a finite subset of labeled examples $\mathcal{D}_{train}$, the training set, is typically sampled to represent it. A function f_θ, parameterized by θ, is then learned to map each input x from $\mathcal{D}_{train}$ to its corresponding label k by minimizing a loss function $\mathcal{L}_\theta(x, k)$. Because the optimization only considers the training set, the model tends to assign lower loss values to training samples compared to unseen ones. This results in loss valleys around training points in the input space, a phenomenon that forms the basis of many membership inference attack [27], [29].

However, Designing a membership inference attack under partial feature knowledge introduces several challenges. We next introduce the challenges and our solutions one by one.

Challenge 1: Incomplete inputs cannot be directly fed into the neural network, which makes it hard to get feedback from the model. The first step is therefore to determine how to fill in the missing features. Several studies have demonstrated that model inversion attacks can successfully recover input features [30]–[35]. These attacks can be broadly categorized into two types. The first type aims to reconstruct representative features of an entire class, typically in scenarios where all samples from that class correspond to the same entity [30]–[32], and these attacks are generally conducted without prior knowledge of partial features. For example, in a facial recognition system, Fredrikson et al. [36] introduced the Face-Rec attack, which reconstructs a human face using the target person's label. The label, representing the target identity in the recognition system, serves as a guiding signal to steer the reconstruction toward the correct person. However, such attacks are not directly applicable to our setting, since in our case a class label does not uniquely correspond to a single entity. The second type of attack, also known as attribute inference attack, attempts to recover unknown private attributes based on the known features of a sample [33]–[35]. Yet, most existing works in this category assume that the attacker has prior knowledge of the possible values of the missing features, which is unrealistic in practice.

Solution to Challenge 1: We adopt a reconstruction strategy similar to that of Fredrikson et al. [36], leveraging the model's inherent memorization of its training data. Specifically, instead of using labels, we treat the known features as anchors and apply backpropagation to optimize the unknown features, aiming to minimize the sample's loss value. Even when the known features originate from a training member, our goal is not to exactly reconstruct the original sample, but to recover what the model "remembers" , that is, the internal impression it has formed during training, typically corresponding to a point near a local minimum of the loss function. According to our observations, when the known features come from a member sample, the reconstruction tends to better align with the true data distribution, as the model has already formed internal representations based on the complete training inputs. In contrast, if the known feature combination is not a member, the optimization may converge to a random local minimum, resulting in a reconstruction that deviates more from the training distribution.

Challenge 2: It is difficult to distinguish members from non-members. Because multiple samples may share the same known features, the reconstruction often becomes a blended representation influenced by several candidates, rather than a faithful replica of any individual one. As a result, the reconstructed sample differs from the original, making exact-match membership inference attack no longer applicable. Moreover, without access to the full ground-truth features, it is difficult to quantify the distance between the reconstructed and original samples, which also hinders the applicability of range-match membership inference attacks.

Solution to Challenge 2: We shift our focus from exact sample matching to assessing how well the reconstructed samples align with the underlying data distribution. Specifically, we employ anomaly detection techniques to identify reconstructed samples that deviate from the true data distribution, enabling us to distinguish between member and non-member feature combinations.

We provide a preliminary validation of the design intuition in Section V-B.

B. Detailed Attack Procedure

As shown in Figure 2, to perform a partial membership inference attack, we first leverage the target model to conduct memory-guided reconstruction of the unknown features. This process can be formalized as

$$\min_{x_{unk}} \mathcal{L}_\theta(\text{cat}(x_{kno}, x_{unk}, M), k), \tag{1}$$

where cat denotes the feature concatenation function that merges the known features x_{kno} and the unknown features x_{unk} according to their original feature order. The mask M serves as a positional indicator, specifying which elements in the input vector correspond to known or unknown features. we adopt the loss function $\mathcal{L}_\theta$ used during model training as our optimization objective. Since multiple samples from different classes may share the same known features, the label y can be arbitrarily chosen from among those associated with the known feature values. We then apply anomaly detection techniques to determine whether the reconstructed sample $\hat{x}$ deviates from the true distribution of class-k samples. Our framework is compatible with any existing anomaly detection method, and we also propose a simple yet effective algorithm. The remainder of this section introduces our approach from two perspectives: memory-guided feature reconstruction and anomaly detection.

Memory-Guided Feature Reconstruction: As outlined in SectionIV-A, we reconstruct the unknown features by minimizing the loss of the model on the sample. As shown in Algorithm 1, for a sample with missing feature values, we first initialize unknown features (line 1). Since the reconstruction process relies on information from the known features, it is important that the model maintains its reliance on them after initialization, in order to prevent the initial values of the unknown features from dominating the early gradient updates. To this end, we initialize all unknown features to zero. We then combine the known features and the initialized unknown features according to their original order(line 2). Since the attacker have access to the loss function during the training process, then we can iteratively update the unknown features via backpropagation to minimize the loss of the sample (lines 3–7).

Anomaly Detection: As shown in Algorithm 2, to assess the degree of deviation of a sample, we first model the distribution of real data (lines 1–8). This involves sampling an auxiliary inference set $\mathcal{D}_{aux}$ from the data distribution and computing the centroid c_k and dispersion for each class. For measuring dispersion, we adopt an approach similar to CADE [22], using the Median Absolute Deviation (MAD) as a robust indicator of spread. Then we compute the distance $\hat{d}$ between the reconstructed sample $\hat{x}_k$ and the centroid of class k, and derive the corresponding base deviation distance δ (lines 9–10). To further enhance inference performance, we train a shadow model θ_s on data drawn from the same distribution and compute a shadow deviation distance δ_s on shadow model based on the same known features, following the same reconstruction procedure. In summary, both δ and δ_s are derived using identical feature values, but from the target and shadow models respectively. The shadow deviation δ_s serves as a reference for how typical the deviation would be if the known features is not a member of the training set. Then, we define a membership inference rule based on the relative deviation δ_s/δ. Let MRAD denote this relative deviation and the inference rule can be formulated as:

$$\text{MRAD}(x_{kno}, \theta, \pi') > \tau, \tag{2}$$

If this condition is satisfied, we infer that the known feature combination belongs to a member of the training set, where τ is a predefined threshold. By varying τ, we can plot the ROC curve to evaluate the effectiveness of this detection strategy.

Threshold Selection: Previous work has shown that MIA is only meaningful when the false positive rate is sufficiently low [5]. Therefore, we adopt a threshold selection strategy similar to that proposed by Ye et al. [14]. Specifically, the adversary can sample a batch of samples $\mathcal{D}_{non}$ from the overall non-member data space π'. Given a predefined acceptable false positive rate α, the adversary then selects a threshold that satisfies

$$\frac{|\{x_{kno} \in \mathcal{D}_{non} \odot M : \text{MRAD}(x_{kno}, \theta, \pi') > \tau\}|}{|\mathcal{D}_{non}|} = \alpha. \tag{3}$$

Algorithm 1 Memory-guided Reconstruction

```
Input: Known features $x_{kno}$ from class k, target model $\theta$,
    number of iterations $N_{itr}$, step length $\eta$, known feature
    mask $M$, loss function $\mathcal{L}$ during the training process of
    the target model.
$x_{unk} \leftarrow 0$
$\hat{x} \leftarrow \text{cat}(x_{kno}, x_{unk}, M)$ // Combine the known and
    unknown features according to their original feature order.
for $i = 1$ to $N_{itr}$ do
    $g \leftarrow grad(\mathcal{L}_\theta(\hat{x}, k))$
    $(g_{kno}, g_{unk}) \leftarrow \text{split}(g, M)$ // Split the gradient vec-
        tor into gradients of known features and gradients of
        unknown features.
    $x_{unk} \leftarrow x_{unk} - \eta \cdot g_{unk}$
    $\hat{x} \leftarrow \text{cat}(x_{kno}, x_{unk}, M)$
end for
return $\hat{x}$
```

Algorithm 2 Anomaly Detection

```
Input: sample $\hat{x}_k$ generated from Algorithm 1 with target
    model, $\hat{x}_{s,k}$ generated from Algorithm 1 with shadow
    model, label $k \in \{1, \ldots, K\}$, auxiliary data $x_{aux,k}$ drawn
    from data distribution, number of samples $n_k$ per class in
    the auxiliary data.
for k=1 to K do
    $c_k \leftarrow \frac{1}{n_k}\sum_{i=1}^{n_k} x^i_{aux,k}$
    for $i = 1$ to $n_k$ do
        $d^i_{aux,k} \leftarrow \|x^i_{aux,k} - c_k\|_2$
    end for
    $\tilde{d}_k \leftarrow \text{median}(d^i_{aux,k}),\ i = 1, \ldots, n_k$
    $\text{MAD}_k \leftarrow \text{median}(|d^i_{aux,k} - \tilde{d}_k|)$
end for
$\hat{d} \leftarrow \|\hat{x}_k - c_k\|_2$, $\hat{d}_s \leftarrow \|\hat{x}_{s,k} - c_k\|_2$
$\delta \leftarrow \frac{|\hat{d} - \tilde{d}_k|}{\text{MAD}_k}$, $\delta_s \leftarrow \frac{|\hat{d}_s - \tilde{d}_k|}{\text{MAD}_k}$
return $\delta_s / \delta$
```

V. Evaluation

Next, we evaluate the attack performance of our proposed algorithm. We begin by presenting the experimental setup, followed by a comprehensive assessment of the attack performance with widely used benchmark datasets. We then analyze the sensitivity of algorithm to hyperparameters and examine how the importance of known features influence performance. Finally, we conduct a case study to demonstrate the practical applications of our attack.

A. Experiment setup

Dataset & Model: We evaluate our attack with three widely used image datasets and one tabular dataset. For all image datasets, the target model is AlexNet, while for the tabular dataset, we employ a multilayer perceptron (MLP). We train each model for 500 epochs with a learning rate of 0.001. We provide a detailed overview of each dataset as below.

- **CIFAR-10** [37]: CIFAR-10 is a widely used benchmark dataset for image classification. It contains 50,000 training images and 10,000 test images, each being a 32×32 RGB image.
- **Fashion-MNIST** [38]: Fashion-MNIST is a drop-in replacement for the original MNIST dataset, featuring 60,000 training images and 10,000 test images. Each image is a 28×28 grayscale representation of an article of clothing.
- **STL-10** [39]: STL-10 contains 13,000 labeled images and 100,000 unlabeled images, making it suitable for supervised, unsupervised, and self-supervised learning. In our experiments, we use only the labeled subset. Each image is a 96×96 RGB image.
- **Epsilon** [40]: Epsilon was derived from the PASCAL 2008 challenge, contains simulated physical data for a binary classification task. Each sample consists of 2,000 normalized numerical features.

Known Features & Unknown Features: We identify unknown features using a randomly generated mask. For image data, the mask is applied over the 2D pixel grid. In the case of a three-channel image, if a pixel is selected as unknown, all three channels at that pixel location are treated as unknown features.

Evaluate Metrics: We evaluate the attack performance using standard metrics commonly used in membership inference research. Specifically, we use the Area Under the ROC Curve (AUC) to measure the overall effectiveness of the attack, and the true positive rate at a false positive rate of 0.1 (TPR@0.1FPR) to assess how well the attack performs under a strict low-false-positive setting.

Anomaly Detection Methods: In addition to our proposed method, we integrate three representative anomaly detection techniques: CADE [22], MSAD [25], and NCI [26]. These comparisons demonstrate the compatibility of our framework with various existing techniques and allow for a comprehensive evaluation across different methodologies. The three baseline methods are summarized as follows:

- CADE: CADE is a concept drift detection method that leverages contrastive learning with an autoencoder. It minimizes the distance between embeddings of samples from the same class while maximizing the distance between those from different classes, causing distribution-shifted samples to deviate significantly in the embedding space.
- MSAD: MSAD is an out-of-distribution (OOD) detection approach. It fine-tunes a pre-trained model using a specially designed contrastive loss and then detects anomalies based on the similarity between a sample and in-distribution samples. In our implementation, we use a locally trained shadow model as the pre-trained backbone

for MSAD.

- NCI: NCI is a post-hoc OOD detection method based on the phenomenon of neural collapse. It observes that in-distribution (ID) samples tend to align with their class prototype vectors in the penultimate layer, whereas OOD samples do not. Additionally, ID features tend to cluster near the origin. We adopt the publicly available implementation from github [41].

B. Intuitive Experiment

To offer a preliminary assessment of the effectiveness of our method, we first perform a simple validation experiment based on our core design intuition. In this experiment, we compute the basic deviation δ using the target model θ for both member and non-member features and compare the results. For each dataset, we randomly designate half of the features as known. Because the gradient magnitude is small, we set a larger step size of $\eta = 100$ and perform only one optimization step (the effect of hyperparameters will be further discussed in Section V-D). As shown in Figure3, across all four datasets, member samples consistently show much lower deviation values than non-member samples. These results support the soundness of our intuition and provide initial evidence for the viability of the proposed attack.

C. Performance Evaluation

We begin by evaluating the attack performance under different proportions of known features. We vary the known feature percentage from 10% to 90% in increments of 10% and conduct experiments at each level. In addition, we include a leave-one-out setting where only a single feature is unknown. The results are shown in Figures 4 and 5, where the x-axis indicates the percentage of known features. The leave-one-out setting is marked as ”loo” on the x-axis. We also plot the full ROC curves on a log scale presented in Figure 6. The evaluation results indicate that the attack begins to demonstrate meaningful effectiveness when the known feature percentage reaches 50%. As the proportion of known features increases, an overall improvement is observed in both evaluation metrics. Notably, even the simple anomaly detection method proposed in this work delivers strong performance. In particular, it consistently achieves higher true positive rates under low false positive rate conditions across most datasets.

We attribute the upward trend to the increased amount of information available to the attacker as the proportion of known features grows. With more known features, the optimization space becomes more constrained, allowing the attack easier to converge. However, we observe a notable performance drop when the known feature percentage reaches 90%. This decline is likely due to the limited number of unknown features. Since the known features remain unchanged, a high known percentage leads to reconstructed samples looks like real ones. In such cases, the known features dominate the anomaly detection process, making it harder for the model to capture meaningful deviations. At 100% known features, no modification is needed, which results in the failure of all attack methods. Consequently, the task reduces to a standard Membership Inference Attack, for which conventional MIA techniques can be applied directly.

D. Hyperparameter Sensitivity

To have a better understanding of the behavior of our attack, we further analyze the influence of two key hyperparameters with a known percentage of 80%.

Performance VS Number of Iterations: We first assess the sensitivity of our algorithm to the number of optimization iterations, N_{itr}. With the step size η fixed at 100, we vary N_{itr} and report the corresponding AUC and TPR@0.1FPR, as shown in Table I and Table II. The results indicate that the performance of our algorithm remains largely stable across different iteration counts, suggesting low sensitivity to this parameter. Remarkably, even a single iteration is sufficient to achieve strong performance.

Performance vs. Step Length: Next, we evaluate how the step length parameter η affects the attack performance. The results are summarized in Table III and Table IV. In this experiment, we fix the number of optimization iterations to $N_{\text{itr}} = 1$. We find that when η is too small, the attack becomes less effective. This is because samples with known features exhibit low gradient magnitudes. As η increases, attack performance improves consistently across various datasets and settings.

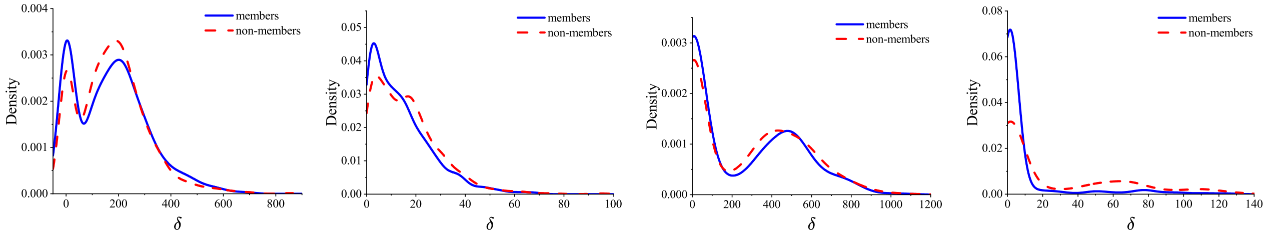
E. Ablation Study

To further validate the effectiveness of our algorithm, we skip the first stage and directly perform anomaly detection using random initialized unknown features. The results are shown in Figure 7 and Figure 8. As observed, when the unknown features are not optimized through loss minimization, the anomaly detection methods fail to discriminate between reconstructed samples containing member features and those containing non-member features. This outcome is intuitive and further highlights the critical role of the first-stage reconstruction in enabling successful PFMI.

F. Impact of Feature Importance on Attack Performance

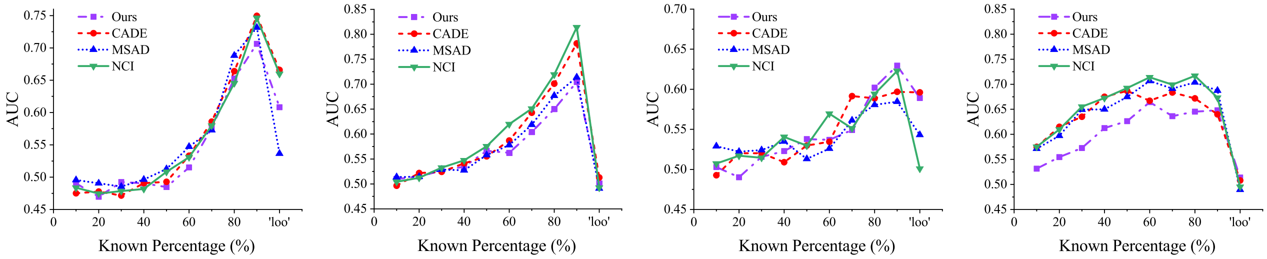
Since models perform inference based on feature values and different features contribute unequally to the model output, we investigate how the importance of known features influences attack performance. In this section, we use SHAP [42] to quantify the contribution of each feature to the prediction of the target model. SHAP is a widely used method for interpreting matchine learning models by assigning an importance value to each feature. It is based on the concept of Shapley value from game theory. Features that contribute more to the output are assigned higher importance values. We evaluate the attack performance under different known feature proportions, using two selection strategies:

- K=IF (Keep Important Features as Known Features): The known features are chosen by ranking all features according to their importance scores and selecting the top ones in descending order.



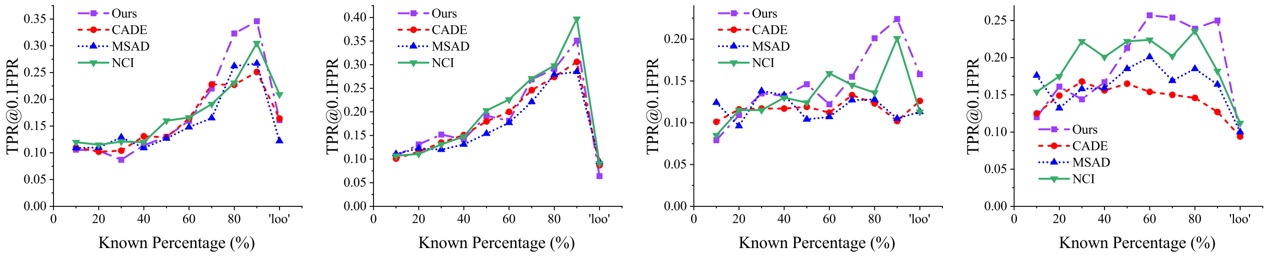
(a) CIFAR-10 (b) STL-10 (c) Fashion-MNIST (d) Epsilon

Fig. 3. Distribution of the base deviation δ for member (blue) and non-member (red) samples across different datasets.



(a) CIFAR-10 (b) STL-10 (c) Fashion-MNIST (d) Epsilon

Fig. 4. Attack performance (AUC) under varying known feature proportions. The x-axis represents the percentage of known features, and the y-axis shows the corresponding attack AUC. Each curve corresponds to one of the four integrated anomaly detection methods within our attack framework. We denote the case where only one feature is unknown as “loo”.



(a) CIFAR-10 (b) STL-10 (c) Fashion-MNIST (d) Epsilon

Fig. 5. Attack performance (TPR@0.1FPR) under varying known feature proportions. The x-axis represents the percentage of known features, and the y-axis shows the corresponding attack TPR when TPR=0.1. Each curve corresponds to one of the four integrated anomaly detection methods within our attack framework. We denote the case where only one feature is unknown as “loo”.

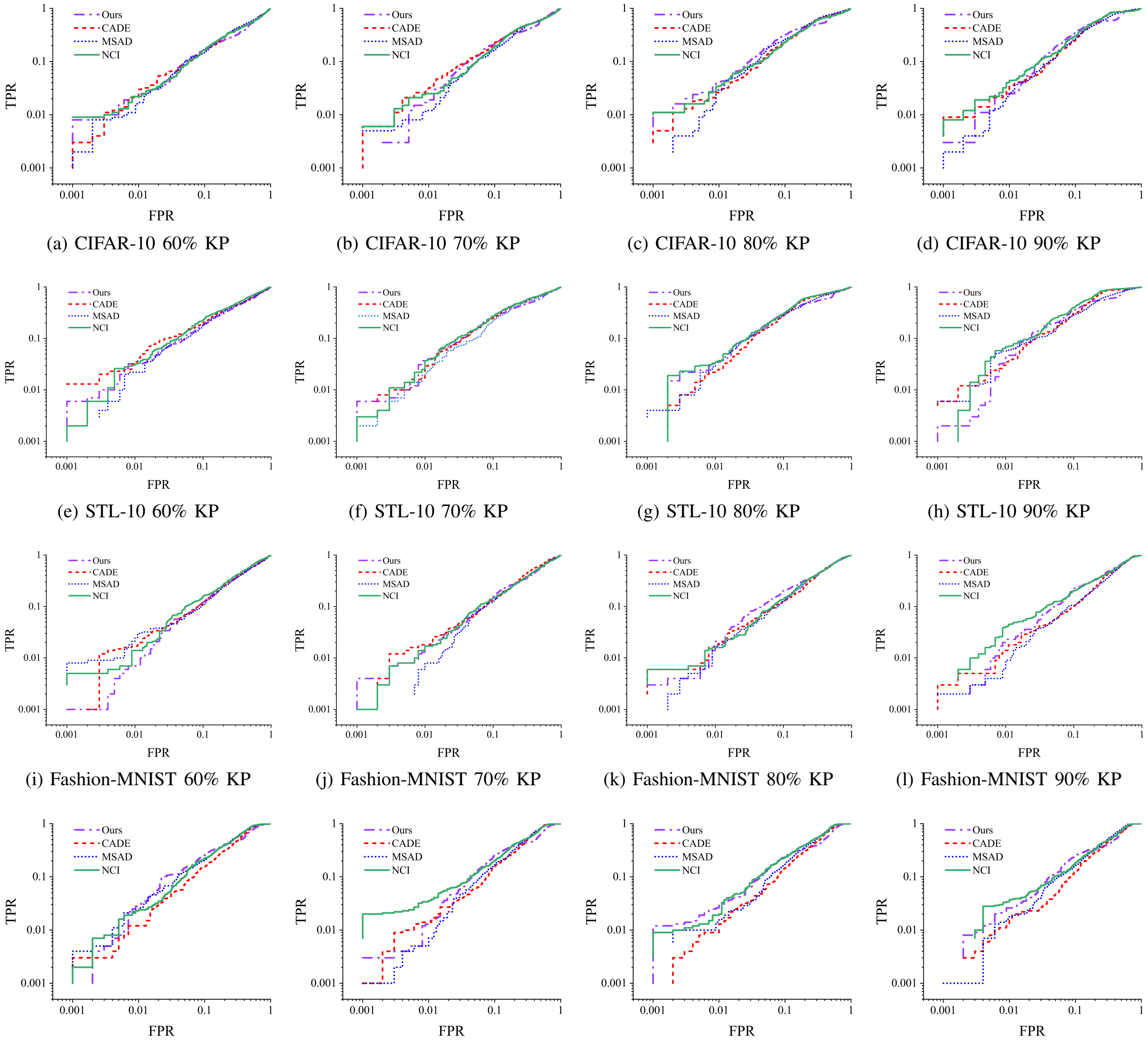
TABLE I
AUC PERFORMANCE UNDER DIFFERENT NUMBERS OF OPTIMIZATION ITERATIONS (N_{itr}) ACROSS VARIOUS DATASETS.

N_{itr}	CIFAR-10				STL-10				Fashion-MNIST				Epsilon			
	Ours	CADE	MSAD	NCI	**Ours**	CADE	MSAD	NCI	**Ours**	CADE	MSAD	NCI	**Ours**	CADE	MSAD	NCI
1	0.658	0.670	0.672	0.652	0.636	0.705	0.668	0.738	0.590	0.576	0.583	0.615	0.662	0.667	0.701	0.735
2	0.641	0.655	0.660	0.684	0.670	0.694	0.683	0.714	0.577	0.582	0.566	0.606	0.656	0.674	0.690	0.723
5	0.627	0.668	0.659	0.669	0.675	0.697	0.687	0.701	0.570	0.598	0.571	0.590	0.634	0.673	0.710	0.748
10	0.645	0.647	0.668	0.643	0.664	0.690	0.682	0.690	0.567	0.575	0.579	0.582	0.643	0.679	0.723	0.719
20	0.623	0.647	0.676	0.665	0.677	0.694	0.690	0.687	0.567	0.601	0.569	0.599	0.649	0.687	0.725	0.742

- K=UF (Keep Unimportant Features as Unknown Features): The known features are selected as those with the lowest importance scores, based on an ascending ranking of all features.

The results are presented in Figure 9. Several observations emerge:

- When the known features are those with high SHAP importance, the attack achieves significantly better performance compared to when unimportant features are known. This is likely because important features drive the model prediction, while unimportant known features lead to vague optimization directions and weak gradient



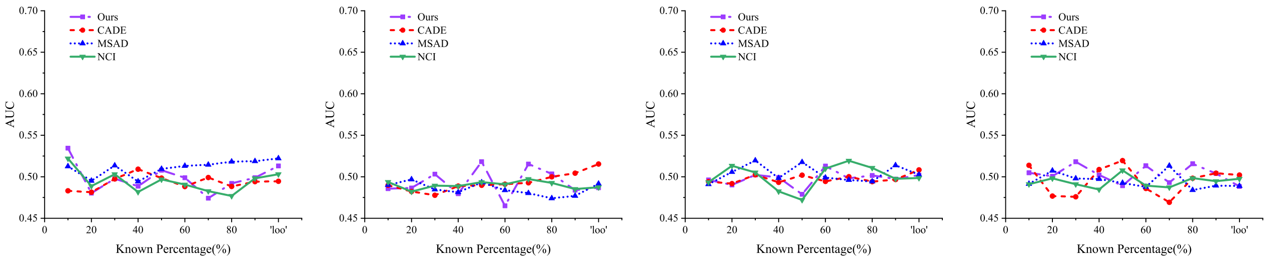
(a) CIFAR-10 60% KP (b) CIFAR-10 70% KP (c) CIFAR-10 80% KP (d) CIFAR-10 90% KP

(e) STL-10 60% KP (f) STL-10 70% KP (g) STL-10 80% KP (h) STL-10 90% KP

(i) Fashion-MNIST 60% KP (j) Fashion-MNIST 70% KP (k) Fashion-MNIST 80% KP (l) Fashion-MNIST 90% KP

(m) Epsilon 60% KP (n) Epsilon 70% KP (o) Epsilon 80% KP (p) Epsilon 90% KP

Fig. 6. Log-scale ROC curves on different datasets under varying known feature percentages (KP: Known Percentage).



(a) CIFAR-10 (b) STL-10 (c) Fashion-MNIST (d) Epsilon

Fig. 7. AUC results when the unknown features are randomly initialized without optimization. We denote the case where only one feature is unknown as “loo”.

TABLE II

TPR@0.1FPR UNDER DIFFERENT NUMBERS OF OPTIMIZATION ITERATIONS (N_{itr}) ACROSS VARIOUS DATASETS.

N_{itr}	CIFAR-10				STL-10				Fashion-MNIST				Epsilon			
	Ours	CADE	MSAD	NCI	**Ours**	CADE	MSAD	NCI	**Ours**	CADE	MSAD	NCI	**Ours**	CADE	MSAD	NCI
1	0.283	0.261	0.252	0.217	0.264	0.314	0.268	0.324	0.192	0.117	0.128	0.177	0.244	0.157	0.184	0.241
2	0.295	0.260	0.243	0.286	0.328	0.280	0.266	0.333	0.172	0.124	0.127	0.187	0.237	0.162	0.155	0.212
5	0.273	0.259	0.264	0.266	0.371	0.284	0.315	0.326	0.158	0.133	0.127	0.147	0.238	0.161	0.175	0.254
10	0.265	0.243	0.254	0.259	0.341	0.309	0.318	0.310	0.162	0.124	0.123	0.152	0.227	0.182	0.211	0.201
20	0.240	0.232	0.261	0.282	0.382	0.308	0.314	0.332	0.179	0.119	0.132	0.157	0.229	0.204	0.222	0.221

TABLE III

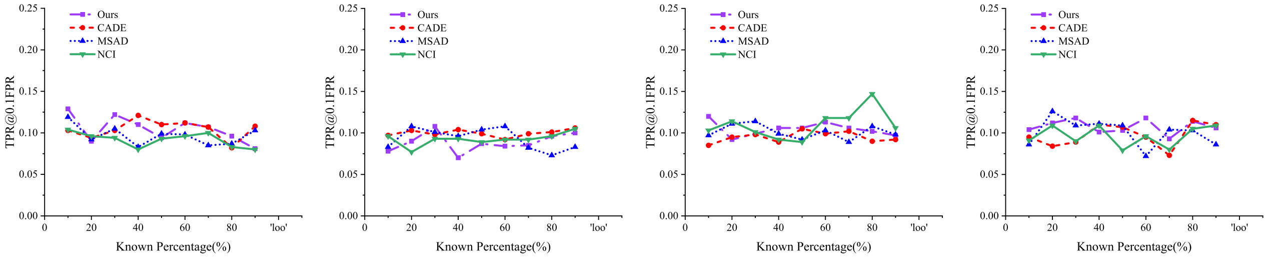
AUC PERFORMANCE OF OUR ATTACK ON DIFFERENT DATASETS UNDER VARYING STEP LENGTHS (η).

η	CIFAR-10				STL-10				Fashion-MNIST				Epsilon			
	Ours	CADE	MSAD	NCI	**Ours**	CADE	MSAD	NCI	**Ours**	CADE	MSAD	NCI	**Ours**	CADE	MSAD	NCI
0.01	0.488	0.495	0.518	0.497	0.489	0.517	0.477	0.484	0.471	0.500	0.507	0.500	0.541	0.495	0.498	0.509
0.1	0.457	0.496	0.521	0.493	0.478	0.517	0.477	0.486	0.484	0.503	0.504	0.505	0.512	0.482	0.496	0.485
1	0.460	0.490	0.513	0.498	0.457	0.516	0.471	0.488	0.536	0.538	0.535	0.517	0.481	0.457	0.504	0.553
10	0.623	0.645	0.530	0.635	0.466	0.486	0.483	0.610	0.588	0.580	0.570	0.607	0.605	0.595	0.659	0.701
100	0.656	0.678	0.654	0.635	0.657	0.706	0.672	0.728	0.593	0.580	0.557	0.592	0.648	0.666	0.702	0.758
1000	0.639	0.667	0.680	0.670	0.663	0.713	0.716	0.723	0.606	0.595	0.578	0.588	0.680	0.711	0.763	0.777

TABLE IV

TPR@0.1FPR OF OUR ATTACK ON DIFFERENT DATASETS UNDER VARYING STEP LENGTHS (η).

η	CIFAR-10				STL-10				Fashion-MNIST				Epsilon			
	Ours	CADE	MSAD	NCI	**Ours**	CADE	MSAD	NCI	**Ours**	CADE	MSAD	NCI	**Ours**	CADE	MSAD	NCI
0.01	0.097	0.108	0.103	0.093	0.096	0.116	0.077	0.095	0.074	0.089	0.108	0.111	0.155	0.090	0.081	0.110
0.1	0.062	0.089	0.105	0.096	0.075	0.102	0.084	0.098	0.079	0.097	0.093	0.126	0.092	0.079	0.095	0.091
1	0.074	0.105	0.097	0.110	0.075	0.101	0.071	0.101	0.092	0.111	0.092	0.123	0.026	0.077	0.092	0.117
10	0.260	0.207	0.112	0.210	0.084	0.080	0.098	0.189	0.189	0.115	0.119	0.172	0.146	0.111	0.160	0.189
100	0.302	0.222	0.255	0.227	0.279	0.313	0.274	0.296	0.147	0.119	0.119	0.157	0.233	0.153	0.166	0.236
1000	0.271	0.244	0.248	0.275	0.347	0.328	0.320	0.291	0.177	0.139	0.137	0.146	0.291	0.147	0.248	0.281



(a) CIFAR-10 (b) STL-10 (c) Fashion-MNIST (d) Epsilon

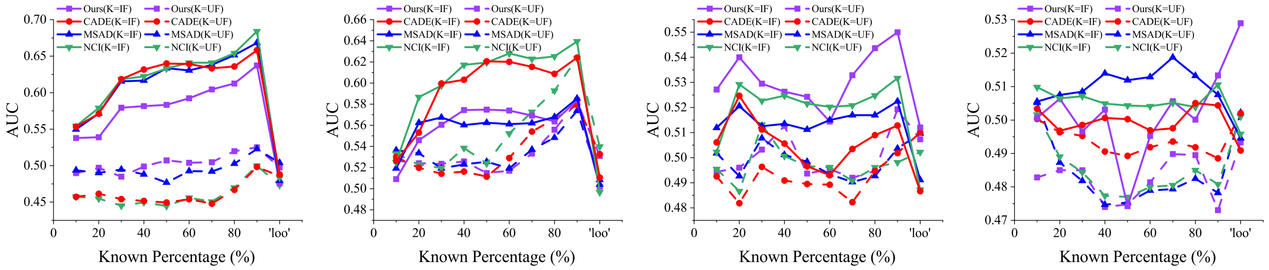
Fig. 8. TPR@0.1FPR results when the unknown features are randomly initialized without optimization. We denote the case where only one feature is unknown as "loo".

signals.

- When the percentage of known features is small enough, using important features as known features outperforms random selection. We attribute this to the fact that important features are more likely to activate memorized patterns in the model, which helps ensure the effectiveness of the reconstruction even when only a small number of such features are available.
- When the percentage of known features exceeds a certain threshold, using only important features as known ones may yield lower performance compared to random selection. This is because when all important features are already known and the remaining few features are non-important, the unknown features have little influence on the loss. As a result, feature reconstruction can hardly yield additional useful membership information.

G. Case Study

We now present a realistic scenario to demonstrate how our proposed attack can be applied in a real-world context.



(a) CIFAR-10 (b) STL-10 (c) Fashion-MNIST (d) Epsilon

Fig. 9. Impact of known feature importance on attack performance (AUC) across different datasets. Solid lines represent scenarios where known features are the most important ones (K=IF), while dashed lines indicate scenarios where the least important features are known (K=UF). We denote the case where only one feature is unknown as "loo".

TABLE V
AUC PERFORMANCE OF CASE STUDY.

	Ours	CADE	MSAD	NCI
AUC	0.63	0.58	0.65	0.58

Consider a small community where a local hospital provides medical services to most residents. The hospital has developed a patient risk prediction system based on historical records of diabetic patients to forecast the likelihood of readmission.

Suppose there is an adversary within a community who aims to infer private information about a specific individual named Alice. The attacker is assumed to know certain publicly accessible demographic information about Alice, such as her race, age, and gender, which helps narrow down the population scope. Using this partial information and the PFMI attack, the attacker successfully determines that Alice was part of the training dataset of the hospital. Since all training records in the dataset correspond to diabetic patients, the attacker can reasonably conclude that Alice is very likely a diabetic patient, thereby exposing sensitive medical information.

To simulate this scenario, we use the UCI-Diabetes dataset. We select 10 features for model training and designate race, age, and gender, common demographic attributes easily accessible to an attacker, as known features. Other experimental settings remain consistent with previous sections. The attack results are shown in Table V, demonstrating that our method can effectively compromise individual privacy in practical settings.

VI. Discussion

A. Defenses

As with other inference attacks, overfitting plays a key role in the success of the attack. Therefore, any technique that mitigates overfitting, such as early stopping [43], dropout [44], and other regularization methods, can serve as a potential defense.

Differential privacy [45], as a provable privacy-preserving framework, also serves as a viable defense strategy. By introducing randomness into the system, it effectively reduces the risk of inferring the presence of any individual in the whole data group. Abadi et al. [46] introduced differential privacy into the field of machine learning by proposing the DP-SGD training framework, which provides formal privacy guarantees during model optimization. This approach has been widely adopted as a defense against membership inference attack.

B. Relationship to Other Attacks

Relationship to Property Existence Attack: Property existence attack [18], as a special case of distribution inference attack [18], [47], aims to determine whether a given feature combination exists in the training set. The only difference from our attack lies in whether the attacker has access to the unknown features. Clearly, an attacker capable of performing PFMI can easily carry out a property existence attack, but it is difficult for a property existence attacker to perform PFMI because of the absence of some features.

Relationship to Model Inversion Attack: A common premise of existing model inversion attacks is that the adversary knows the target to be present in the training set [30]–[35]. Even though the membership status of the target is unknown in the attack scenario considered in this paper, the first stage of our attack can still be regarded as a form of model inversion attack. Leveraging existing model inversion techniques thus represents a feasible approach for improving performance.

C. Limitation and Future Work

While our proposed method has demonstrated promising attack performance, several limitations remain.

First, the attack presented in this paper is based on a white-box setting, which may limit its applicability in certain scenarios. To bridge this gap, one possible direction is to explore gradient approximation techniques [48], [49], which allow attackers to approximate gradients without full model access. Additionally, model stealing methods [50], [51] could be leveraged to construct a white-box surrogate of the target model, thereby enabling the application of our attack in a black-box context.

Secondly, when the percentage of known features becomes too large, the attack performance tends to degrade. Based on our experimental results, one potential solution is to selectively discard some known features during inference—particularly those that, while somewhat influential to the prediction of the model, are not essential for preserving the representativeness of the sample with respect to the underlying individual. This strategy may help mitigate performance drops and enhance the overall effectiveness of the attack.

Furthermore, due to the missing feature values, the mapping between the known features and the label may be one-to-many. This paper assumes that the attacker can obtain at least one valid label, which is a realistic assumption in practice. However, exploring label-free attacks could further enhance the capability of the attacker in more extreme scenarios and is therefore a worthwhile direction for future research.

VII. Conclusion

In this paper, we study membership inference attack under the setting where some feature values are missing. To address this scenario, we introduce a practical two-stage attack framework named MRAD. Our framework is compatible with arbitrary anomaly detection methods, enabling attackers to flexibly conduct attacks under various knowledge settings.

We conduct extensive experiments on multiple datasets to evaluate the effectiveness of our method. The results show that our framework consistently delivers strong performance across a wide range of known feature percentages. In most datasets, once the known feature ratio exceeds 50%, the attack begins to produce reliable results. Remarkably, even when 40% of the features are missing, the attack still achieves an AUC of 0.6. In addition, we use SHAP to measure feature importance and experimentally confirm that the importance of the known features has a significant impact on the attack's performance. Finally, through a case study, we simulate a real-world application scenario to further highlight the practical relevance of our attack method.

Our findings demonstrate that membership inference attacks can be mounted using publicly available partial data, even when a sample's private features are protected. This highlights a new dimension in privacy risk assessment.

Acknowledgment

The authors would like to thank...

References

[1] I. Bilionis, R. C. Berrios, L. Fernandez-Luque, and C. Castillo, "Disparate model performance and stability in machine learning clinical support for diabetes and heart diseases," *arXiv preprint arXiv:2412.19495*, 2024.

[2] B. Hui, D. Yan, H. Chen, and W.-S. Ku, "Trajnet: A trajectory-based deep learning model for traffic prediction," in *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, 2021, pp. 716–724.

[3] L. Chen, P. Wu, K. Chitta, B. Jaeger, A. Geiger, and H. Li, "End-to-end autonomous driving: Challenges and frontiers," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024.

[4] R. Shokri, M. Stronati, C. Song, and V. Shmatikov, "Membership inference attacks against machine learning models," in *2017 IEEE symposium on security and privacy (SP)*. IEEE, 2017, pp. 3–18.

[5] N. Carlini, S. Chien, M. Nasr, S. Song, A. Terzis, and F. Tramer, "Membership inference attacks from first principles," in *2022 IEEE symposium on security and privacy (SP)*. IEEE, 2022, pp. 1897–1914.

[6] Y. Pang and T. Wang, "Black-box membership inference attacks against fine-tuned diffusion models," *arXiv preprint arXiv:2312.08207*, 2023.

[7] G. Chen, Y. Zhang, and F. Song, "Slmia-sr: Speaker-level membership inference attacks against speaker recognition systems," *arXiv preprint arXiv:2309.07983*, 2023.

[8] X. Chi, X. Zhang, Y. Wang, L. Qi, A. Beheshti, X. Xu, K.-K. R. Choo, S. Wang, and H. Hu, "Shadow-free membership inference attacks: recommender systems are more vulnerable than you thought," *arXiv preprint arXiv:2405.07018*, 2024.

[9] X. Wang and W. H. Wang, "Link membership inference attacks against unsupervised graph representation learning," in *Proceedings of the 39th Annual Computer Security Applications Conference*, 2023, pp. 477–491.

[10] Y. Peng, J. Roh, S. Maji, and A. Houmansadr, "Oslo: one-shot label-only membership inference attacks," *arXiv preprint arXiv:2405.16978*, 2024.

[11] S. Zarifzadeh, P. Liu, and R. Shokri, "Low-cost high-power membership inference attacks," in *International Conference on Machine Learning*. PMLR, 2024, pp. 58 244–58 282.

[12] A. Sablayrolles, M. Douze, C. Schmid, Y. Ollivier, and H. Jégou, "White-box vs black-box: Bayes optimal strategies for membership inference," in *International Conference on Machine Learning*. PMLR, 2019, pp. 5558–5567.

[13] L. Song and P. Mittal, "Systematic evaluation of privacy risks of machine learning models," in *30th USENIX Security Symposium (USENIX Security 21)*, 2021, pp. 2615–2632.

[14] J. Ye, A. Maddi, S. K. Murakonda, V. Bindschaedler, and R. Shokri, "Enhanced membership inference attacks against machine learning models," in *Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security*, 2022, pp. 3093–3106.

[15] Z. Li and Y. Zhang, "Membership leakage in label-only exposures," in *Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security*, 2021, pp. 880–895.

[16] B. Hui, Y. Yang, H. Yuan, P. Burlina, N. Z. Gong, and Y. Cao, "Practical blind membership inference attack via differential comparisons," *arXiv preprint arXiv:2101.01341*, 2021.

[17] J. Tao and R. Shokri, "Range membership inference attacks," in *2025 IEEE Conference on Secure and Trustworthy Machine Learning (SaTML)*. IEEE, 2025, pp. 346–361.

[18] H. Chaudhari, J. Abascal, A. Oprea, M. Jagielski, F. Tramer, and J. Ullman, "Snap: Efficient extraction of private properties with poisoning," in *2023 IEEE Symposium on Security and Privacy (SP)*. IEEE, 2023, pp. 400–417.

[19] L. Wang, J. Wang, J. Wan, L. Long, Z. Yang, and Z. Qin, "Property existence inference against generative models," in *33rd USENIX Security Symposium (USENIX Security 24)*, 2024, pp. 2423–2440.

[20] C. A. Choquette-Choo, F. Tramer, N. Carlini, and N. Papernot, "Label-only membership inference attacks," in *International conference on machine learning*. PMLR, 2021, pp. 1964–1974.

[21] S. Wares, J. Isaacs, and E. Elyan, "Data stream mining: methods and challenges for handling concept drift," *SN Applied Sciences*, vol. 1, pp. 1–19, 2019.

[22] L. Yang, W. Guo, Q. Hao, A. Ciptadi, A. Ahmadzadeh, X. Xing, and G. Wang, "{CADE}: Detecting and explaining concept drift samples for security applications," in *30th USENIX Security Symposium (USENIX Security 21)*, 2021, pp. 2327–2344.

[23] K. Wan, Y. Liang, and S. Yoon, "Online drift detection with maximum concept discrepancy," in *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2024, pp. 2924–2935.

[24] J. Park, Y. G. Jung, and A. B. J. Teoh, "Nearest neighbor guidance for out-of-distribution detection," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2023, pp. 1686–1695.

[25] T. Reiss and Y. Hoshen, "Mean-shifted contrastive loss for anomaly detection," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 37, no. 2, 2023, pp. 2155–2162.

[26] L. Liu and Y. Qin, "Detecting out-of-distribution through the lens of neural collapse," *arXiv preprint arXiv:2311.01479*, 2023.

[27] S. Yeom, I. Giacomelli, M. Fredrikson, and S. Jha, "Privacy risk in machine learning: Analyzing the connection to overfitting," in *2018 IEEE 31st computer security foundations symposium (CSF)*. IEEE, 2018, pp. 268–282.

[28] A. Salem, G. Cherubin, D. Evans, B. Köpf, A. Paverd, A. Suri, S. Tople, and S. Zanella-Béguelin, "Sok: Let the privacy games begin! a unified treatment of data inference privacy in machine learning," in *2023 IEEE Symposium on Security and Privacy (SP)*. IEEE, 2023, pp. 327–345.

[29] H. Hu, Z. Salcic, G. Dobbie, and X. Zhang, “Membership inference attacks on machine learning: A survey. corr abs/2103.07853 (2021),” *arXiv preprint arXiv:2103.07853*, 2021.

[30] S. V. Dibbo, D. L. Chung, and S. Mehnaz, “Model inversion attack with least information and an in-depth analysis of its disparate vulnerability,” in *2023 IEEE Conference on Secure and Trustworthy Machine Learning (SaTML)*. IEEE, 2023, pp. 119–135.

[31] M. Kahla, S. Chen, H. A. Just, and R. Jia, “Label-only model inversion attacks via boundary repulsion,” in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 15 045–15 053.

[32] S. An, G. Tao, Q. Xu, Y. Liu, G. Shen, Y. Yao, J. Xu, and X. Zhang, “Mirror: Model inversion for deep learning network with high fidelity,” in *Proceedings of the 29th Network and Distributed System Security Symposium*, 2022.

[33] M. S. M. S. Annamalai, A. Gadotti, and L. Rocher, “A linear reconstruction approach for attribute inference attacks against synthetic data,” in *33rd USENIX Security Symposium (USENIX Security 24)*, 2024, pp. 2351–2368.

[34] S. Mehnaz, S. V. Dibbo, E. Kabir, N. Li, and E. Bertino, “Are your sensitive attributes private? novel model inversion attribute inference attacks on classification models,” in *31st USENIX Security Symposium (USENIX Security 22)*, 2022, pp. 4579–4596.

[35] E. Kabir, L. Craig, and S. Mehnaz, “Disparate privacy vulnerability: Targeted attribute inference attacks and defenses,” in *34st USENIX Security Symposium (USENIX Security 25)*, 2025.

[36] M. Fredrikson, S. Jha, and T. Ristenpart, “Model inversion attacks that exploit confidence information and basic countermeasures,” in *Proceedings of the 22nd ACM SIGSAC conference on computer and communications security*, 2015, pp. 1322–1333.

[37] A. Krizhevsky, G. Hinton *et al.*, “Learning multiple layers of features from tiny images,” 2009.

[38] H. Xiao, K. Rasul, and R. Vollgraf, “Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms,” *arXiv preprint arXiv:1708.07747*, 2017.

[39] A. Coates, A. Ng, and H. Lee, “An analysis of single-layer networks in unsupervised feature learning,” in *Proceedings of the fourteenth international conference on artificial intelligence and statistics*. JMLR Workshop and Conference Proceedings, 2011, pp. 215–223.

[40] PASCAL, “Pascal large scale learning challenge,” https://www.csie.ntu.edu.tw/˜cjlin/libsvmtools/datasets/binary.html, 2008.

[41] J. Zhang, J. Yang, P. Wang, H. Wang, Y. Lin, H. Zhang, Y. Sun, X. Du, Y. Li, Z. Liu, Y. Chen, and H. Li, “Openood v1.5: Enhanced benchmark for out-of-distribution detection,” *arXiv preprint arXiv:2306.09301*, 2023.

[42] S. M. Lundberg and S.-I. Lee, “A unified approach to interpreting model predictions,” in *Advances in Neural Information Processing Systems 30*, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Eds. Curran Associates, Inc., 2017, pp. 4765–4774. [Online]. Available: http://papers.nips.cc/paper/7062-a-unified-approach-to-interpreting-model-predictions.pdf

[43] A. Steier, L. Ramaswamy, A. Manoel, and A. Haushalter, “Synthetic data privacy metrics,” *arXiv preprint arXiv:2501.03941*, 2025.

[44] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, “Dropout: a simple way to prevent neural networks from overfitting,” *The journal of machine learning research*, vol. 15, no. 1, pp. 1929–1958, 2014.

[45] C. Dwork, “Differential privacy,” in *International colloquium on automata, languages, and programming*. Springer, 2006, pp. 1–12.

[46] M. Abadi, A. Chu, I. Goodfellow, H. B. McMahan, I. Mironov, K. Talwar, and L. Zhang, “Deep learning with differential privacy,” in *Proceedings of the 2016 ACM SIGSAC conference on computer and communications security*, 2016, pp. 308–318.

[47] A. Suri, Y. Lu, Y. Chen, and D. Evans, “Dissecting distribution inference,” in *2023 IEEE Conference on Secure and Trustworthy Machine Learning (SaTML)*. IEEE, 2023, pp. 150–164.

[48] A. Wibisono, M. J. Wainwright, M. Jordan, and J. C. Duchi, “Finite sample convergence rates of zero-order stochastic optimization methods,” in *Advances in Neural Information Processing Systems*, F. Pereira, C. Burges, L. Bottou, and K. Weinberger, Eds., vol. 25. Curran Associates, Inc., 2012.

[49] X. Lian, H. Zhang, C.-J. Hsieh, Y. Huang, and J. Liu, “A comprehensive linear speedup analysis for asynchronous stochastic parallel optimization from zeroth-order to first-order,” in *Advances in Neural Information Processing Systems*, D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett, Eds., vol. 29. Curran Associates, Inc., 2016.

[50] J.-B. Truong, P. Maini, R. J. Walls, and N. Papernot, “Data-free model extraction,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2021, pp. 4771–4780.

[51] H. Zhu, W. Hu, S. Liang, F. Li, W. Wang, and S. Wang, “Efficient and effective model extraction,” in *ICASSP 2025 - 2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2025, pp. 1–5.