When Better Features Mean Greater Risks: The Performance-Privacy Trade-Off in Contrastive Learning

Ruining Sun
Seroney_Sun@outlook.com
School of Mathematics and Computational Science, Xiangtan University
Xiangtan, Hunan, China

Hongsheng Hu
hongsheng.hu@newcastle.edu.au
School of Information and Physical Sciences, University of Newcastle
Newcastle, NSW, Australia

Wei Luo
wei.luo@deakin.edu.au
School of Information Technology, Deakin University
Burwood, VIC, Australia

Zhaoxi Zhang
zhaoxi.zhang-1@student.uts.edu.au
School of Computer Science, University of Technology Sydney
Ultimo, NSW, Australia

Yanjun Zhang
Yanjun.Zhang@uts.edu.au
School of Computer Science, University of Technology Sydney
Ultimo, NSW, Australia

Haizhuan Yuan*
yhz@xtu.edu.cn
School of Mathematics and Computational Science, Xiangtan University
Xiangtan, Hunan, China

Leo Yu Zhang*
leo.zhang@griffith.edu.au
School of Information and Communication Technology, Griffith University
Southport, QLD, Australia

Abstract

With the rapid advancement of deep learning technology, pretrained encoder models have demonstrated exceptional feature extraction capabilities, playing a pivotal role in the research and application of deep learning. However, their widespread use has raised significant concerns about the risk of training data privacy leakage. This paper systematically investigates the privacy threats posed by membership inference attacks (MIAs) targeting encoder models, focusing on contrastive learning frameworks. Through experimental analysis, we reveal the significant impact of model architecture complexity on membership privacy leakage: As more advanced encoder frameworks improve feature-extraction performance, they simultaneously exacerbate privacy-leakage risks. Furthermore, this paper proposes a novel membership inference attack method based on the p-norm of feature vectors, termed the Embedding Lp-Norm Likelihood Attack (LpLA). This method infers membership status, by leveraging the statistical distribution characteristics of the p-norm of feature vectors. Experimental results across multiple datasets and model architectures demonstrate that LpLA outperforms existing methods in attack performance and robustness, particularly under limited attack knowledge and query volumes. This study not only uncovers the potential risks of privacy leakage in contrastive learning frameworks, but also provides a practical basis for privacy protection research in encoder models. We hope that this work will draw greater attention to the privacy risks associated with self-supervised learning models and shed light on the importance of a balance between model utility and training data privacy. Our code is publicly available at: https://github.com/SeroneySun/LpLA_code.

CCS Concepts

• **Security and privacy**;

Keywords

Contrastive learning, membership inference attack, likelihood estimation, privacy leakage, trustworthy AI

ACM Reference Format:
Ruining Sun, Hongsheng Hu, Wei Luo, Zhaoxi Zhang, Yanjun Zhang, Haizhuan Yuan, and Leo Yu Zhang. 2025. When Better Features Mean Greater Risks: The Performance-Privacy Trade-Off in Contrastive Learning. In *ACM Asia Conference on Computer and Communications Security (ASIA CCS '25), August 25–29, 2025, Hanoi, Vietnam.* ACM, New York, NY, USA, 13 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 Introduction

In recent years, self-supervised learning (SSL) [15, 44] has become a powerful model pretraining method, gaining widespread attention due to its efficient training paradigm and transferability. SSL leverages pretext tasks such as context-based [12, 13, 25, 35], masking-based [1, 19, 46, 50], and contrast-based [5, 14, 20, 45] approaches to enable models to train on large-scale unlabeled data and develop powerful encoder mapping inputs to a representation space. This

*Haizhuan Yuan and Leo Yu Zhang are corresponding authors.

ASIA CCS '25, Hanoi, Vietnam

ACM ISBN 978-x-xxxx-xxxx-x/YYYY/MM
https://doi.org/10.1145/nnnnnnn.nnnnnnn

technology not only addresses the challenges of limited and expensive labeled data but also provides a highly transferable pre-trained encoder, facilitating better performance in downstream tasks such as image classification and object detection.

Among these, contrastive learning has gained prominence due to its stronger transferability and stable, fast convergence during the training process. However, these models may inadvertently expose private information contained within training data, such as gradient inversion [52], inference attack [4, 33, 39], adversarial example [47, 49], and data poisoning [43]. They expose vulnerabilities in model training and deployment, raising widespread societal concerns and prompting extensive policy discussions. For example, legal frameworks like the General Data Protection Regulation (GDPR) and the California Consumer Privacy Act (CCPA) impose stringent requirements to protect the collection and usage of user data. Therefore, a critical challenge in machine learning research is how to effectively protect data privacy while maintaining model performance.

Among various privacy risks studies [9], membership inference attack (MIA) [22, 39], as one of the most prevalent privacy attacks, aims to determine whether a given target sample was part of the training dataset based on model's output (or other information). The emergence of MIA highlights the vulnerability of data privacy protection in machine learning models, especially when the model is overfitting, as it may leak more crucial information about the training dataset, posing a threat to user's privacy. This concern is especially pronounced in applications involving sensitive personal information, such as healthcare, finance, or social media. In these domains, unauthorized usage of user data for commercial profit or malicious exploitation purposes can result in severe harm and losses.

Beyond its role as a privacy attack, MIA has increasingly been adopted as a compliance auditing tool under regulations. This dual role—as both a privacy threat and a regulatory metric—underscores the critical importance of understanding and mitigating MIA risks, particularly in self-supervised learning (SSL) frameworks where traditional defenses may fall short. Most existing MIA studies focus on enhancing the efficiency of membership inference in classification tasks [2, 29, 41] or other specific tasks [16, 18, 40], whereas how to balance model performance and privacy protection is currently lacking [31, 48]. This gap is particularly evident in self-supervised learning models, where traditional MIA strategies are often ineffective due to the models' unsupervised training objectives and frameworks. These characteristics make understanding the privacy risks in SSL challenging, let alone developing effective mitigation strategies.

Our work: We take a step further in addressing the above challenges by focusing on the performance-privacy trade-off in contrastive learning. By constructing an assessment framework for evaluating the utility and privacy of encoders, we analyze the mechanisms of privacy leakage in contrastive learning models and their behavior under membership inference attacks. Additionally, based on the insights gained from this assessment, we propose a novel attack method that leverages the distribution of feature vectors' p-norm, referred to as Embedding Lp-Norm Likelihood Attack (LpLA).

Contributions: The main contributions are summarized as follows:

- A systematic assessment of the performance-privacy trade-off in contrastive learning. This study reveals, for the first time, that both the different contrastive learning frameworks and the backbone architectures significantly influence the performance and membership privacy leakage of SSL pre-trained encoders. Furthermore, we summarize how different attacks leverage different types of information to perform membership inference attacks.
- Proposal of a membership inference attack method based on p-norm of feature vectors (LpLA). Unlike existing methods that rely on model confidence, loss values, or similarity calculations, LpLA uses the p-norm of feature vectors as the attack signal for membership inference. Experiments show that LpLA performs comparably to, or even outperforms, existing attacks across a variety of scenarios, while requiring fewer attack knowledge and query volumes, providing a novel perspective for privacy analysis.
- Comprehensive experiments for evaluating performance-privacy trade-off and LpLA performance in contrastive learning. Through comparative analysis of multiple attack methods and experimental validation, we offer a comprehensive assessment of the performance-privacy trade-off and LpLA's effectiveness. These findings serve as valuable references for future research on the privacy of contrastive learning models.

2 Preliminaries and Related Work

2.1 Visual Self-Supervised Representation Learning

Self-supervised representation learning aims to automatically learn effective and robust feature representations from unlabeled data, providing a solid technical foundation for various downstream tasks in the field of computer vision, such as image classification, object detection, and image segmentation. To achieve this goal, researchers have proposed a variety of approaches. In the domain of computer vision, these approaches can generally be categorized into three main types. Context-based methods [12, 13, 25, 35] leverage the inherent contextual relationships between data instances (e.g., spatial structures, local-to-global consistency) to construct learning objectives that help models capture more representative features. Masking-based methods [1, 19, 46, 50] typically consist of an encoder-decoder architecture. The encoder maps partially masked inputs into a low-dimensional feature space to extract critical features, while the decoder attempts to reconstruct the original input based on these features.

Contrastive-based methods. Compared with the previous two categories, contrastive learning has gained popularity due to its stronger transferability and stable, fast convergence during the training process. Unlike context-based and masking-based methods, which train encoders indirectly through complex pretext tasks, contrastive-based self-supervised representation learning directly relies on a simple discrimination task to train the encoder. Since its introduction in [45], instance discrimination-based contrastive learning methods have quickly become the mainstream approach in this field. The core mechanism involves generating sample pairs through data augmentation, where each sample is paired with both

positive and negative examples. The model then optimizes a contrastive loss function to bring the feature vectors of positive pairs closer together while pushing apart the feature vectors of negative pairs. This allows the encoder to learn feature representations capable of distinguishing between similar and dissimilar inputs. However, early contrastive learning methods were constrained by the simplified design of their training frameworks and the limitations of their data augmentation strategies, which prevented them from fully boosting their efficient feature extraction capabilities.

The capabilities of contrastive learning were fully realized with the introduction of SimCLR and MoCo in [5, 20]. In [5], positive and negative sample pairs are generated through richer data augmentation, and feature vectors are produced using an encoder and a multi-layer perceptron (MLP) structure. By treating other samples within the same batch as negatives to compute the contrastive loss, this approach significantly improves the quality of learned representations. In contrast, [20] introduced MoCo for the first time, which proposes a momentum update strategy to maintain a sufficiently large momentum dictionary, ensuring consistency in dictionary representations while constructing sample pairs. This method not only significantly reduces the computational overhead during training but also further optimizes the effectiveness of feature extraction capabilities.

Subsequently, more sophisticated contrastive learning methods have been proposed. For instance, SwAV [3] introduces the concept of clustering to replace simple pairwise comparisons, while BYOL [14] and SimSiam [7] adopt a self-distillation framework to discard negative examples and rely solely on positive examples for optimization. These innovative approaches have inspired many new research directions in the field of self-supervised representation learning.

2.2 Membership Inference Attack

The privacy issues of deep learning models have garnered widespread attention in recent years. Among privacy risks studies [9], membership inference attack (MIA) [22], a widely used privacy attack, aims to infer whether a specific data sample was part of a model's training dataset by observing the model's output behavior on the specific sample. MIA, first introduced by Shokri et al. [39], exploits the outputs of machine learning models to infer membership status, raising significant privacy concerns in real-world applications. Since then, MIA research has rapidly expanded to cover various types of victim models, including regression models [16], classification models [39], generative models [18], and encoder models [40]. At the same time, a range of defense methods has been proposed to effectively mitigate MIA attacks while maintaining the utility of the model [9, 23, 32]. Now, it's not only a privacy attack on deep learning models but also serves as an audit metric for evaluating the privacy risk of models or algorithms.

Classic MIAs. Existing works of MIA mainly focus on classification models, where an adversary leverages various information unintentionally leaked under different settings to perform inference. Most of these methods follow Shokri using the model's output, which is the most straightforward information, as an attack signal. Specifically, in a black-box setting [2, 29, 36, 41], the adversary often uses the confidence score from a classification model to train a binary classifier. In contrast, in a white-box setting [34, 36, 37], the adversary can exploit richer information about the target model (such as gradient information from intermediate layers), combining them into higher-dimensional features, which are then used as an attack signal to infer membership status.

In addition, other types of attack signals have also been explored in previous research for MIA. For example, [27] demonstrated that by calculating entropy or logits from the confidence score, the adversary can use this data as an attack signal and determine a threshold to infer membership status. In strictly black-box scenarios [29], the adversary can also construct an attack signal using only the label information output by the classification model to complete the inference. Furthermore, LiRA [2] stands out for introducing a likelihood ratio statistic as the attack signal, which, in combination with the model's loss distribution on samples, achieves highly accurate membership inference. LiRA provides a novel perspective for optimizing attacks by focusing on the distribution of statistical metrics. Beyond classification models, there are also studies on non-classic attack signals in encoder models. For example, [40] used the similarity between word vectors as an attack signal in text encoders, implementing a highly representative attack method.

MIA against encoder models. In the field of membership inference attacks targeting visual self-supervised learning, [30] is the first (and, at the time of writing, the only) study to propose a membership inference attack targeting contrastive learning models. The method, EncoderMI, leverages the differences in cosine similarity between augmented samples' embedding of member and non-member data to train a binary classifier for distinguishing membership status. EncoderMI has laid the foundation for subsequent research on membership inference attacks against visual representation learning models. For example, [11] proposed a strategy to construct attack features based on similarity for the person re-identification task, successfully compromising the privacy of person re-identification encoders under black-box scenarios. Furthermore, significant progress has been made in membership inference attacks targeting masked pretraining encoders. For instance, [51] extended the EncoderMI method by computing the similarity between feature vectors of different parts of the same image, achieving more general attack performance. Similarly, [28] combined the shadow model technique, leveraging a locally trained shadow decoder to mimic the behavior of the target model and construct a pseudo-loss function, where lower pseudo-loss corresponds to higher membership confidence.

Overall, utilizing the similarity-based attack signal plays an indispensable role in membership inference attacks targeting encoder models. This is largely due to the common understanding in existing research that, compared to the outputs of classification models, the feature vectors output by encoders often lack explicit and direct semantic information [11, 30]. As a result, straightforward strategies that directly use model outputs as attack signal [22, 38] tend to perform sub-optimally on encoder models. However, through a series of experiments, we find that under some contrastive learning frameworks, directly using the feature vectors output by encoders can also achieve effective membership inference attacks. Furthermore, we propose a likelihood estimation attack method based on p-norm of feature vectors. This method demonstrates significant and robust attack performance against encoder models, providing

a novel perspective for membership privacy research in encoder models.

3 Systematically Evaluating Privacy Risks of Encoders

In this section, we first introduce the framework for evaluating the utility and privacy of encoders. Next, we specifically introduce several contrastive learning models and existing membership inference attacks, which are used as target encoders and privacy metrics to understand the trade-offs between model utility and privacy risks. Last, we introduce the experimental settings and summarize the findings from the experimental results.

The systematic evaluation of the privacy risks of encoders aims to answer the two research questions (RQs) as follows:

- **RQ1**: Do encoders trade off the utility of the model with privacy risks? Put differently, does a higher utility model suffer more from privacy leakage?
- **RQ2**: How do attackers leverage the information produced by the target model to perform membership inference attacks on the target sample?

Answering **RQ1** and **RQ2** is crucial to understanding the fundamental attack mechanism of membership inference on encoder models. To answer **RQ1**, we propose our evaluation mechanism, which consists of two main parts: The first part measures the feature extraction capability of the encoder model by examining the performance on downstream tasks, which serves as the utility indicator of the encoder; The second part uses membership inference attack, a widely used privacy attack as the privacy metric, to measure the privacy risks of the encoder. To answer **RQ2**, we evaluate benchmarks of membership inference attacks on a series of encoders to obtain the attack performance on them, and we summarize how different attacks leverage different information to perform attacks.

3.1 Overview of the Evaluation Mechanism

We present the evaluation mechanism in Figure 1. Specifically, it consists of two parts: the utility measurement part and the privacy measurement part. Before we detail the two parts, we introduce the threat model in this paper.

Threat model. We consider that there is a victim who owns a self-supervised learning model, which is referred to target model. There is also an adversary who has only black-box access to the target model. That is, the adversary can send query data examples to the target model and receive model outputs. The adversary's goal is to infer the private information about the training dataset of the target model. For membership inference attacks in this paper, the attacker is to predict whether a given data sample was in the training dataset or not.

Given the defined threat model, we now detail the utility measurement and privacy measurement in the evaluation mechanism.

Utility measurement: To measure the utility of encoder models, we consider that the victim holds an unlabeled private dataset $\mathcal{T}_p$ and uses a specific contrastive learning framework $\mathcal{M}$ for self-supervised training. After the training process, a pre-trained encoder model $\mathcal{E}$ is obtained, which can be used for transfer learning in downstream tasks. The encoder $\mathcal{E}(\cdot)$ takes as input an image sample x and outputs its high-dimensional feature vector v in the representation space. This helps improve the performance of downstream tasks during transfer learning. We use the performance of $\mathcal{E}$ on downstream tasks as a measure of the encoder model's feature extraction capability.

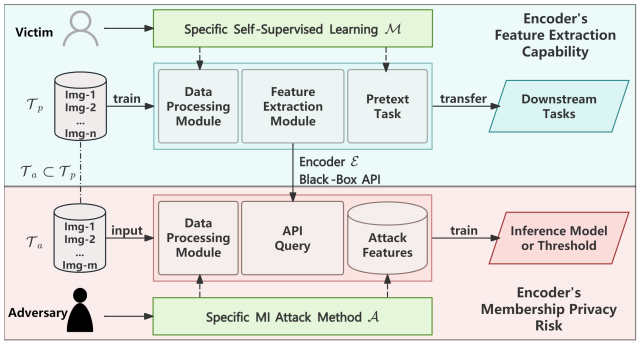


Figure 1: An illustration of the evaluation framework. The framework consists of two parts: a utility measurement part for evaluating the model performance and a privacy measurement for evaluating the model's privacy risks.

Privacy measurement: Following [31, 38, 51], we consider that the adversary can access a partial training dataset $\mathcal{T}_a \subset \mathcal{T}_p$. Formally, the adversary's goal is to determine whether a target sample x was used to train the encoder model $\mathcal{E}$ or not, i.e., whether $x_{\text{target}} \in \mathcal{T}_p$ or not. In this paper, we consider the standard membership inference game where we pick the target sample from the training dataset with 50% probability. To comprehensively measure the privacy risks of the target encoder, the adversary leverages a series of membership inference methods $\mathcal{A}$ that use different attack signals or features to construct the attack model. We use the performance of the attack model on the target encoder $\mathcal{E}$ as its privacy risk metric. Intuitively, a better membership inference attack performance means the encoder is less private.

3.2 Target Contrastive Learning Models

In this paper, we concentrate on the contrastive learning models of the MoCo series, including the MoCo-v1 [20], MoCo-v2 [6], and MoCo-v3 [8]. The MoCo series progressively refined its framework to enhance the feature extraction capabilities and demonstrate outstanding performance across multiple visual tasks. These models have been successfully applied to various encoder architectures, including Convolutional Neural Networks (CNNs) [21] and Vision Transformers (ViTs) [10]. The core idea of MoCo is to pull together different augmented views of the same image (positive samples) while pushing apart the features of different images stored in the momentum dictionary (negative samples). We note that there are other contrastive learning models such as SimCLR [5] and BYOL [14] etc. In this paper, we do not select them because MoCo's progressive refinement facilitates our organized and systematic evaluation of the influence between model complexity and privacy leakage to answer RQ1. Below, we summarize the major improvements in architectural design, loss optimization objectives, and dictionary maintenance strategies across the different versions of MoCo.

MoCo-v1 is the foundational framework of the MoCo series, which consists of three core components: An image encoder (f), a momentum encoder (f_m), and a dynamic dictionary (Γ). The image encoder f is responsible for generating feature vectors for an augmented input, while the momentum encoder f_m, updating at a slower rate using a momentum update strategy, is responsible for generating key vectors for another augmented input. The dynamic dictionary Γ maintains a queue to store key vectors produced by the momentum encoder for inputs from previous mini-batches.

Specifically, given a mini-batch of N inputs, MoCo-v1 generates two augmented versions, x_{q} and x_{k}, for each input x. These augmented inputs are processed by f and f_m, to produce a query vector q_x and key vector k_x respectively:

$$\begin{cases} f(x) = \mathrm{Pred.}(\mathcal{E}(x)), \\ q_x = f(x_{\mathrm{q}}), \\ k_x = f_m(x_{\mathrm{k}}), \end{cases} \tag{1}$$

where $\mathrm{Pred.}(\cdot)$ is a linear layer that goes end to end with $\mathcal{E}(\cdot)$ (i.e., serves as the pretext task in Figure 1). The parameters of image encoder f are optimized using a contrastive loss function (InfoNCE loss [17]), while the parameters of momentum encoder f_m are updated using a momentum-based update rule as follows:

$$\ell(x) = -\log \frac{\exp(\mathrm{sim}(q_x, k_x)/\tau)}{\exp(\mathrm{sim}(q_x, k_x)/\tau) + \sum_{z\in\Gamma} \exp(\mathrm{sim}(q_x, z)/\tau)}, \tag{2}$$

$$\theta_{\mathrm{k}} \leftarrow m\theta_{\mathrm{k}} + (1-m)\theta_{\mathrm{q}}, \tag{3}$$

where $\mathrm{sim}(\cdot,\cdot)$ denotes the cosine similarity function, τ is the temperature parameter, and m controls the update speed of the momentum encoder, which is typically set to 0.999. MoCo-v1 dynamically updates the dictionary by removing the oldest batch of key vectors and adding newly generated ones after each mini-batch training iteration. This mechanism ensures continuous optimization by maintaining a diverse and up-to-date set of negative samples in the dictionary.

MoCo-v2 is the second framework of the MoCo series refined based on Moco-v1. Firstly, it adopted stronger data augmentation strategies, such as multi-scale cropping and color jittering. Secondly, it replaces the single fully connected layer in Pred. with a two-layer multilayer perceptron (MLP). Experimental results demonstrated that this modification produces representations better suited for more transfer learning tasks, enabled MoCo-v2 to achieve performance on multiple unsupervised learning benchmarks closely approached of supervised methods.

MoCo-v3 is the SOTA framework of the MoCo series, which is further refined by introducing significant adjustments in the model framework, loss function, and dictionary maintenance. Firstly, in terms of the model framework, which is the primary focus of RQ1, the image encoder f now consists of three components: the backbone network $\mathcal{E}$, a projection head Proj., and a prediction head Pred.. In contrast, the momentum encoder f_m is composed of only a backbone network $\mathcal{E}$ and a projection head Pred., where Pred. and Proj. are all constructed using fully connected layers, normalization layers, and stacked ReLU activation functions,

$$\begin{cases} f(x) = \mathrm{Proj.}(\mathrm{Pred.}(\mathcal{E}(x))), \\ f_m(x) = \mathrm{Pred.}(\mathcal{E}(x)). \end{cases} \tag{4}$$

Additionally, there are also changes in the loss function and dictionary maintenance strategy. Instead of maintaining a large dictionary Γ, MoCo-v3 uses all non-self samples within the mini-batch as negative samples for loss computing. Under this configuration, given two augmented versions x_1 and x_2 of each input x, there would be two pairs of output from f and f_m respectively, denoted by query vectors q_1, q_2 and key vectors k_1, k_2 respectively:

$$q_i = f(x_i),\ k_i = f_m(x_i),\ i = 1, 2. \tag{5}$$

In the framework of MoCo-v3, the contrastive loss is then computed as follows:

$$\begin{cases} \ell(x_1) = -\log \dfrac{\exp(\mathrm{sim}(q_1, k_2^+)/\tau)}{\exp(\mathrm{sim}(q_1, k_2^+))/\tau) + \sum_{z\in x_2^-} \exp(\mathrm{sim}(q_1, z)/\tau)}, \\ \ell(x_2) = -\log \dfrac{\exp(\mathrm{sim}(q_2, k_1^+)/\tau)}{\exp(\mathrm{sim}(q_2, k_1^+))/\tau) + \sum_{z\in x_1^-} \exp(\mathrm{sim}(q_2, z)/\tau)}, \\ \mathcal{L}(x) = \ell(x_1) + \ell(x_2), \end{cases} \tag{6}$$

where x^- denotes non-self samples within the same mini-batch, and $\mathcal{L}(x)$ represents the final optimization objective used to update f. The momentum encoder f_m is still updated in the same manner using equation (3) (obviously, updating $\mathcal{E}$ and Pred. only). Through these adjustments, MoCo-v3 achieves much better performance on many downstream tasks and is more suitable for the ViT type backbones.

Why MoCo series are selected. The MoCo series have progressively evolved from MoCo-v1 to v3. Innovations such as momentum updates, dynamic dictionaries, the upgrade of prediction head modules, and the introduction of projection head modules have not only significantly improved the performance of pre-trained encoders but also expanded their applicability to more types of backbone.

Meanwhile, they facilitate our organized and systematic evaluation of the influence between model complexity (contrastive learning framework and backbone architecture) and privacy leakage in encoder models, helping us answer **RQ1**.

3.3 Benchmarks of Membership Inference Attacks

In this section, we present an overview of a suite of membership inference attacks proposed in prior studies [11, 30, 31], including EncoderMI [30], SD-MI [11], and Feature-based MI [31]. These attacks serve as the privacy risk evaluation foundation for assessing the target models' privacy leakage.

Feature-based MI. In existing works of membership inference attacks on contrastive learning models, one of the most common attack methods is feature-based MI (referred to as Fe-MI). When targeting classification models, attackers can use the confidence scores or class labels directly as the attack signal for membership inference [39], and there are numerous studies [31, 38] having widely demonstrated its effectiveness against classification models. However, previous research [11, 30] suggests that the feature vectors output by encoder models primarily serve as representations and lack specific semantic information. Consequently, directly using these feature vectors for binary classifier training could not achieve effective attack performance as in classification models.

To comprehensively explore the potential membership inference attack signal and privacy risks faced by encoder models, we consider Fe-MI as a baseline in the evaluation experiments. Counterintuitively, as we will show in Section 3.5, our experimental results demonstrate that this method can also achieve successful inference attacks against encoder models in some attack scenarios.

The adversary directly collects the feature vectors v through the target encoder's API to form an attack dataset, and uses a simple neural network as the attack model (denoted by $Attacker$), which is detailed as follows:

$$\mathcal{I}(x) = \text{Attacker}(v_x). \tag{7}$$

Similarity-based MI. In addition to feature-based MI, there are some MI attack methods specifically designed for encoder models, which typically rely on similarity between feature vectors to construct the attack model. In this paper, we use two representative attack methods as baselines for evaluation: EncoderMI [30] and SD-MI [11]. We will refer to them as similarity-based MI for simplification.

EncoderMI is the first membership inference attack method designed for contrastive learning pre-trained encoders. It leverages the inherent characteristics of contrastive learning optimization objectives, wherein contrastive learning tends to generate similar feature vectors for different augmented views of one input.

To perform membership inference, EncoderMI first generates n augmented views of a given image x using the same data augmentation strategies as the target encoder's training. These augmented views are denoted as $\{x_1, x_2, \ldots, x_{\text{n}}\}$. Subsequently, these views are used to query the model API and produce the corresponding feature vectors $V_x = \{v_x^1, v_x^2, \ldots, v_x^{\text{n}}\}$. As for a specific target sample x, EncoderMI computes an attack feature based on the pairwise similarities of the vectors in V_x as follows:

$$Sig(x) = \{\text{sim}(v_x^i, v_x^j) \mid \ i, j = 1, 2, \ldots, \text{n}, \ j > i\}, \tag{8}$$

where $\text{sim}(\cdot, \cdot)$ represents the cosine similarity. EncoderMI then sorts the resulting $\text{n} \cdot (\text{n} - 1)/2$ similarity scores, and uses these sorted $Sig(x)$ as the attack signal to train an inference model, which is usually a simple binary classifier as follows:

$$\mathcal{I}(x) = \text{Attacker}(Sig(x)). \tag{9}$$

SD-MI is another similarity distribution-based membership inference attack against Re-ID models (Person Re-Identification), which is a type of representation learning, whose training objective is similar to contrastive learning. Re-ID targets optimizing an encoder to map the features of paired (or grouped) image samples as closely as possible while ensuring that the feature mappings of unpaired (or different groups of) image samples remain distinct.

Specifically, SD-MI first randomly selects anchor images X_{anchors} to construct a reference dataset, and query $\mathcal{E}(\cdot)$ with anchors to create an ordered set of feature vectors $V_{\text{anchor}} = \{v_{\text{anc}}^1, v_{\text{anc}}^2, \ldots, v_{\text{anc}}^n\}$. Subsequently, the similarity between target image features v_x and these ordered anchor features V_{anchor} is calculated as a similarity distribution of the target image, which serves as the attack signal of SD-MI. The formulation is as follows:

$$Sig(x) = [\text{dist}(v_x, v_{\text{anc}}^1), \text{dist}(v_x, v_{\text{anc}}^2), \ldots, \text{dist}(v_x, v_{\text{anc}}^n)], \tag{10}$$

where $\text{dist}(\cdot, \cdot)$ represents the Euclidean distance between two feature vectors. SD-MI then feeds this similarity distribution $Sig(x)$ into a binary attack classifier for membership inference.

To further enhance the attack performance, SD-MI introduces a mechanism named *Anchor Selector*, which assigns weights to the similarity distribution $Sig(x)$ based on the target sample's feature vector v_x. By assigning different weights, it enhances the privacy leakage caused by $Sig(x)$ and enables a significant attack improvement compared to the EncoderMI. The complete SD-MI consists of two modules as follows:

$$\mathcal{I}(x) = \text{Attacker}(Sig(x) \odot \text{Selector}(v_x)), \tag{11}$$

where $\odot$ denotes the Hadamard product.

Summary of baseline attacks. The three baseline attack methods are selected for a comprehensive and rigorous evaluation of the privacy risks of encoder models. They respectively represent three different attack mechanisms where each of them considers a different type of attack signal for leaking the information of the training data: (1) the encoder model's output vector v itself; (2) the similarity between outputs of the same sample; (3) the similarity of the target sample with auxiliary datasets. Using such baseline attacks for evaluation helps us to answer **RQ2**: they enable a comprehensive analysis and comparison of the performance across different attack signals for membership inference against encoder models.

3.4 Experimental Setting

Datasets. We select three commonly used datasets in membership inference attack research for experimental evaluation: CIFAR-10, CIFAR-100, and Tiny-ImageNet. These datasets encompass different categories and complexities, effectively satisfying the requirements of our experimental evaluation. As Table 1 shows, we assume the target model is pre-trained with a privacy dataset (20k samples), and tested with a test dataset (10k samples). On the other hand, an adversary uses an attack dataset composed of 2k samples from the privacy dataset and 2k from the test dataset separately. And there is another inference test dataset that is used to evaluate the attack performance, composed of 8k samples from the privacy dataset and 8k from the test dataset disjointed with the attack dataset.

- CIFAR [24]: The CIFAR-10 dataset contains 60,000 color images divided into 10 categories, with each image having a resolution of 32×32×3. Similarly, the CIFAR-100 dataset contains 60,000 color images but with an extended 100 categories, and the images maintain the same resolution of 32×32×3.
- Tiny-Imagenet [26]: This dataset consists of 200 categories, with a total of 100,000 training images and 10,000 test images. Each image has a resolution of 64×64×3.

Target models. Concretely, we consider four widely used backbone networks as the feature extractors, i.e., ResNet-18, ResNet-50 [21], ViT-Small, and ViT-Base [10] in three versions of the MoCo model. Following the setting in [42], we set the size of the momentum dictionary to 16,384 in MoCo-v1 and MoCO-v2, which differs from the official setting of 65,536. This choice is made because a dictionary size that is as large as possible while still smaller than the total training sample size is more conducive to achieving better training performance. Furthermore, following similar settings in [30, 51], all target models were trained for 2,000 epochs.

Table 1: Default implementation details in the experiments.

Model	Train-Set	Test-Set	Batch-Size	Epoch	Other
MoCo-v1&2	$\mathcal{T}_p$ (20k samples)	(None)	256	2k	16,384 for Γ
MoCo-v3	$\mathcal{T}_p$ (20k samples)	(None)	512	2k	(None)
K-NN test	$\mathcal{T}_p$ (20k samples)	$\mathcal{T}_{test}$ (10k samples)	(None)	(None)	k=20
EncoderMI	$(2k \in \mathcal{T}_p) \cup (2k \in \mathcal{T}_{test})$	$(8k \in \mathcal{T}_p) \cup (8k \in \mathcal{T}_{test})$	128	200	10 augmentation
SD-MI	$(2k \in \mathcal{T}_p) \cup (2k \in \mathcal{T}_{test})$	$(8k \in \mathcal{T}_p) \cup (8k \in \mathcal{T}_{test})$	128	200	2k anchors
Fe-MI	$(2k \in \mathcal{T}_p) \cup (2k \in \mathcal{T}_{test})$	$(8k \in \mathcal{T}_p) \cup (8k \in \mathcal{T}_{test})$	128	500	(None)

Attack model. We follow the existing works of EncoderMI [30] and Fe-MI to use a simple three-layer multilayer perceptron structure with ReLU activation functions as the attack model, training with 200 epochs. For SD-MI [11], we adopt the official model design and attack settings, which include an *Anchor Selector* composed of two linear layers with ReLU activation functions, and an attack model consisting of five linear layers with Tanh activation functions.

Metrics. To measure the utility of the target model, we follow the contrastive learning works and use the K-NN clustering algorithm [45] to classify the testing samples based on their representative vectors. A higher accuracy of the K-NN clustering algorithm means a better feature extraction capability of the encoder. To measure the privacy risks of the encoder model, we use the attack performance of the membership inference attack as a metric. To quantify the attack performance, we use four widely used metrics of accuracy, precision, recall, and TPR@0.1%FPR, as the membership inference attack task is essentially a binary classification problem.

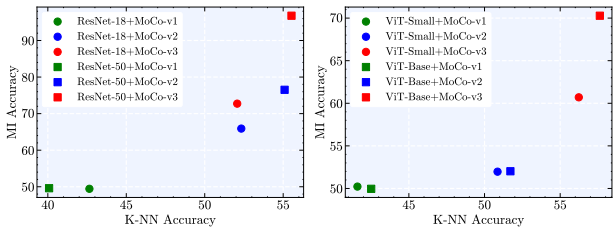
Encoder training. We use the official public code for training MoCo-v1, MoCo-v2[1], and MoCo-v3[2], K-NN clustering[3], and membership inference attack[4]. Additional training configurations are presented in Table 1.

3.5 Experiment Results

We first address the **RQ1:** "Do encoders trade off the utility of the model with privacy risks? Put differently, does a higher utility model suffer more from privacy leakage?" To answer this question, we present the K-NN classification performance (i.e., encoder's utility) and membership inference attack performance (i.e., encoder's privacy) against the encoders trained on CIFAR-100 in Table 2.

Table 2 indicates that as the encoder model becomes progressively more sophisticated, the classification accuracy of the K-NN algorithm on the representative vectors of the testing samples improves correspondingly. This reflects an enhancement in the feature extraction capability of the encoders, i.e., the encoder's utility is better from MoCo-v1 to MoCo-v3. However, alongside the increase in model utility, a notable rise in membership inference accuracy, precision, recall and TPR@0.1%FPR can also be observed. For example, when the target encoder is trained using the simplest MoCo-v1 framework, the accuracy of all attack methods is close to 50% (i.e., random guess), indicating that the adversary cannot infer the membership information. As the MoCo framework is progressively upgraded, the accuracy of all membership inference methods increases against target models trained with MoCo-v2 and MoCo-v3. This reveals that while more complex contrastive learning frameworks

[1]MoCo-v1&v2: https://github.com/facebookresearch/moco
[2]MoCo-v3: https://github.com/facebookresearch/moco-v3
[3]InstDisc: https://github.com/zhirongw/lemniscate.pytorch
[4]SD-MI: https://github.com/Vill-Lab/2023-AAAI-SDMIA



(a) More complex: ResNet (b) More complex: ViT

Figure 2: An illustration of the performance-privacy trade-off when the encoder model becomes progressively more sophisticated. Not only a complex contrastive learning framework, but an advanced backbone leads a higher utility and privacy risks.

enhance feature extraction capabilities, they also lead to higher risks of privacy leakage.

To further support the finding that the encoder models trade their utility for privacy risks, we conduct the experiments on a series of backbone feature extractors, i.e., the ResNet family of ResNet18 and ResNet50 and ViT family of ViT-Small and ViT-Base, across MoCo-v1, MoCo-v2, and MoCo-v3. The experimental results are provided in Figure 2. In each plot in Figure 2, the x-axis represents the model utility and the y-axis represents the model's privacy risks. A data point closer to the bottom right indicates a model with both high utility and privacy preservation. As we can see, the more advanced contrastive learning framework can indeed improve the feature extraction capabilities of the encoders, while the risk of privacy leakage of the model is also increased. In addition, an interesting finding is that as the complexity of backbone architecture increases (e.g., replacing ResNet18 to ResNet50, or replacing ViT-Small to ViT-Base), the K-NN classification accuracy improves, while the membership inference accuracy exhibits a simultaneous increase. This further suggests that the encoder model inherently trades utility for heightened privacy risks.

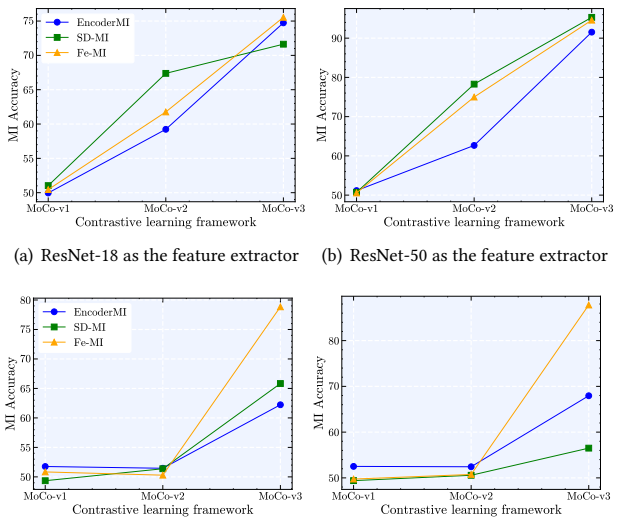
> **Takeaway 1:** In contrastive learning, encoder models trade off the utility with privacy risks. An encoder model having a higher utility usually leads to higher privacy leakage risks.

We now address the **RQ2:** "How do attackers leverage the information produced by the target model to perform membership inference attacks on the target sample?" To answer this question, we compare the performance of different attacks, where each uses different information as the membership signal. We present the experimental results of the three attacks on different feature extractors trained with MoCo-v1, MoCo-v2, and MoCo-v3 in Figure 3.

From Figure 3, we find that Fe-MI not only achieves highly competitive attack performance under several target training settings but even performs on par with (and in some cases exceeded) the other two similarity-based methods under the MoCo-v3 framework. Note that Fe-MI directly leverages the feature vector of the target sample for membership inference, while EncoderMI and SD-MI convert the feature vector into the similarity information for the attack. The experimental results suggest that the feature vector of

Table 2: The K-NN classification accuracy and membership inference attack performance against different encoder models using a different feature extractor.

Backbone	Model	K-NN Accuracy	Encoder-MI				SD-MI				Fe-MI			
			Accuracy	Precision	Recall	TPR@0.1%FPR	Accuracy	Precision	Recall	TPR@0.1%FPR	Accuracy	Precision	Recall	TPR@0.1%FPR
ResNet-18	MoCo-v1	0.427	0.500	0.500	0.522	0.100	0.510	0.513	0.407	0.105	0.505	0.506	0.373	0.103
	MoCo-v2	0.523	0.592	0.595	0.576	0.147	0.674	0.685	0.644	0.217	0.618	0.608	0.660	0.155
	MoCo-v3	0.521	0.747	0.744	0.755	0.291	0.716	0.755	0.640	0.309	0.755	0.770	0.728	0.336
ViT-Small	MoCo-v1	0.416	0.518	0.520	0.453	0.109	0.494	0.494	0.510	0.098	0.508	0.508	0.810	0.104
	MoCo-v2	0.509	0.515	0.517	0.432	0.107	0.514	0.513	0.551	0.105	0.509	0.509	0.503	0.104
	MoCo-v3	0.562	0.622	0.606	0.699	0.154	0.658	0.616	0.841	0.161	0.788	0.792	0.782	0.382



(a) ResNet-18 as the feature extractor (b) ResNet-50 as the feature extractor

(c) ViT-Small as the feature extractor (d) ViT-Base as the feature extractor

Figure 3: An illustration of the three attacks on different feature extractors trained with MoCo frameworks. Fe-MI achieves a notable performance, indicating that the feature vector as the membership signal itself can be leveraged for MIA directly.

the target sample in the representative space contains abundant information for facilitating the membership inference attacks. Unlike similarity-based membership signals using the feature vector as the "side" information, the feature vector as the membership signal itself can directly be leveraged by neural networks for conducting the attack.

However, the Fe-MI method still requires training a deep neural network as the binary attack classifier, which can be computationally expensive. This limitation motivates us to design a new membership inference attack directly using the feature vector while outperforming the Fe-MI method.

Takeaway 2: The feature vector of the target sample and similarity scores based on the feature vector can be leveraged by the adversary for a successful membership inference. Compared to similarity scores-based membership inference, directly using the feature vector as the membership inference signal shows a consistent competent attack performance in various settings.

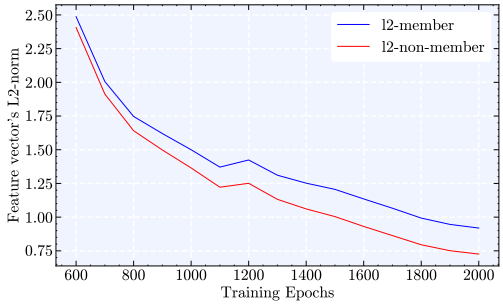


Figure 4: An illustration of the mean L2-norm value of the feature vectors in members and non-members in ResNet-50 trained by MoCo-v2. An obvious distinction of L2-norm values between the two classes of samples can be observed in the training process.

4 Our Attack

In this section, we introduce a new membership inference attack, which directly leverages the feature vector of a target sample to inferring its membership status. Before we go to the details of our proposed attack, we ask the following research questions (RQs):

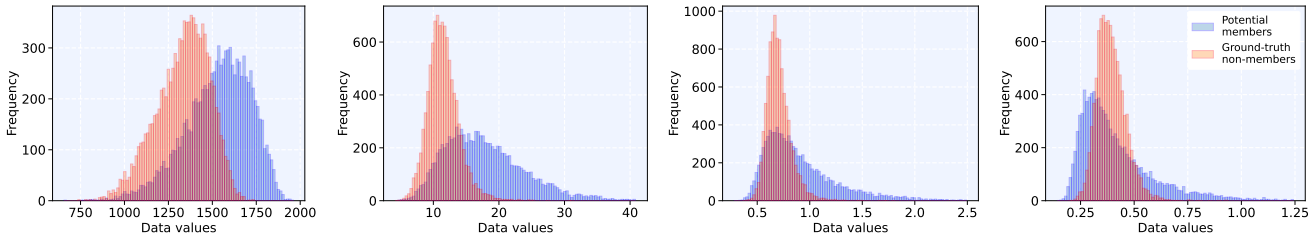
- **RQ3**: How to design a membership inference attack that can directly use the feature vector of the sample as the membership signal but without training a binary attack classifier?
- **RQ4**: What are the advantages of the proposed attack compared to existing methods?

Answering these two questions is crucial for understanding our attack mechanism and can shed light on how easily encoder models may leak their private information about their training data.

4.1 Motivation of the Attack

We first address **RQ3**: "How to design a membership inference attack that can directly use the feature vector of the sample as the membership signal but without training a binary attack classifier?" To answer this question, we first introduce the intuition of the attack and then present the design details of the attack method.

As we observed in **RQ2**, the feature vector of the target sample contains abundant information for membership inference. We take a further look at this vector by examining its magnitude. Specifically, we calculate the L2-norm value of the vectors of members and non-members throughout the training epochs using the ResNet50 feature extractor in the MoCo-v2 framework. As shown in Figure 4, we notice a notable difference in the mean L2-norms of feature



(a) L0-Norm (b) L1-Norm (c) L2-Norm (d) L3-Norm

Figure 5: An illustration of the p-norm of feature vectors produced by official pre-trained ResNet-50 with MoCo-v2 framework. There is a significant difference in the feature vector magnitude distributions between the two classes, wherein the behavior aligns closely with the observation highlighted in Figure 4.

vectors between member and non-member samples. As the training progressed, the L2-norm of feature vectors of non-member samples consistently surpassed those of members.

To further support this observation, we conduct experiments on the official pre-trained ResNet-50 encoder model released by MoCo team[5]. The publicly available pre-trained ResNet-50 encoder was trained using the ImageNet dataset under the MoCo-v2 framework. Thus, we collect 10k random training samples from ImageNet and 10k samples from CIFAR-100 (serving as the non-member samples of the model). We query the pre-trained ResNet-50 encoder and thus can obtain their feature vectors. Subsequently, we calculated four different p-norms of the features. As shown in Figure 5, we observe that there is a significant difference in the feature vector magnitude distributions between the member and non-member samples. In addition, the p-norms value distributions of member and non-member samples can be approximated as two independent Gaussian distributions. This behavior aligns closely with the differences highlighted in Figure 4.

These observations in Figure 4 and Figure 5 motivate us to construct a simple but effective membership inference attack method, named Embedding Lp-Norm Likelihood Attack (LpLA).

4.2 Design of the Attack

In this section, we consider an adversary who holds a partial ground-truth member dataset $\mathcal{T}_a \subset \mathcal{T}_p$ and can only obtain the output v of target encoder $\mathcal{E}$ through black-box API. In default, we set $p = 2$ by default, while showing the effectiveness of the attack using other norms in Section 4.5.

Feature extraction (Stage 1). Firstly, for each sample (partial ground-truth member) held by the adversary $x \in \mathcal{T}_a$, the corresponding output feature vectors can be collected through black-box queries to the target encoder. These feature vectors will be used in Stage 2 to estimate a p-norm distribution for the member class. On the other hand, to construct a non-member dataset $\mathcal{T}_{an}$, the adversary can simply use random generation techniques to create samples matching the input format of the target model, such as generating random pixel images. These samples are likewise queried through $\mathcal{E}$ in a black-box manner and subsequently collected for estimating a distribution of the non-member class.

[5]moco-v2-800ep-pretrain.pth.tar: https://dl.fbaipublicfiles.com/moco/moco_checkpoints/moco_v2_800ep/moco_v2_800ep_pretrain.pth.tar

Likelihood estimation (Stage 2). Based on the previous findings, we assume that the p-norms of the feature vectors for all member and non-member samples follow two independent Gaussian distributions. To perform a membership inference attack on the victim sample under this assumption, the adversary needs to estimate these two normal distributions using a sufficient number of random samples collected like Stage1:

$$\begin{cases} L(x) = \|\mathbf{v_x}\|_p = \sqrt[p]{v_1^p + v_2^p + \cdots + v_n^p}, \\ L_{\mathrm{m}} \sim \mathcal{N}\left(\mu_{\text{member}}, \sigma^2_{\text{member}}\right), \\ L_{\mathrm{nm}} \sim \mathcal{N}\left(\mu_{\text{non-member}}, \sigma^2_{\text{non-member}}\right), \end{cases} \tag{12}$$

where $\mathbf{v_x}$ denotes a n dimension feature vector output from the encoder with input x. And μ_{member}, σ_{member} represent the mean and standard deviation of the p-norm values of member samples, respectively, $\mu_{\text{non-member}}$ and $\sigma_{\text{non-member}}$ are the corresponding parameters for non-member samples (To simplify notation, we adjusted some subscripts in this section: μ_{m} represents μ_{member}, and μ_{nm} represents $\mu_{\text{non-member}}$ etc.).

Then, The adversary can estimate these parameters of two distributions based on the available p-norm values of member and non-member samples:

$$\begin{cases} \hat{\mu}_{\mathrm{m}} = \dfrac{1}{k}\displaystyle\sum_{i=1}^{k} L(x_i),\ \hat{\sigma}^2_{\mathrm{m}} = \dfrac{1}{k-1}\displaystyle\sum_{i=1}^{k}\left(L(x_i) - \hat{\mu}_{\mathrm{m}}\right)^2, \\ \hat{\mu}_{\mathrm{nm}} = \dfrac{1}{q}\displaystyle\sum_{j=1}^{q} L(x_j),\ \hat{\sigma}^2_{\mathrm{nm}} = \dfrac{1}{q-1}\displaystyle\sum_{j=1}^{q}\left(L(x_j) - \hat{\mu}_{\mathrm{nm}}\right)^2, \\ x_i \in \mathcal{T}_a,\ x_j \in \mathcal{T}_{an}. \end{cases} \tag{13}$$

Inferring membership (Stage 3). Finally, for a given victim sample x with p-norm value $L(x)$, the adversary can calculate a posterior probability of membership using Bayes' theorem:

$$\begin{cases} \mathcal{L}_{\mathrm{m}}(x) = \dfrac{1}{\sqrt{2\pi\sigma^2_{\mathrm{m}}}} \exp\left(-\dfrac{(L(x) - \mu_{\mathrm{m}})^2}{2\sigma^2_{\mathrm{m}}}\right), \\ \mathcal{L}_{\mathrm{nm}}(x) = \dfrac{1}{\sqrt{2\pi\sigma^2_{\mathrm{nm}}}} \exp\left(-\dfrac{(L(x) - \mu_{\mathrm{nm}})^2}{2\sigma^2_{\mathrm{nm}}}\right), \\ P(\mathrm{m} \mid x) = \dfrac{\mathcal{L}_{\mathrm{m}}(x)P(\mathrm{m})}{\mathcal{L}_{\mathrm{m}}(x)P(\mathrm{m}) + \mathcal{L}_{\mathrm{nm}}(x)P(\mathrm{nm})}. \end{cases} \tag{14}$$

Table 3: Attack accuracy (TPR@0.1%FPR) against encoders.

Dataset	Model	Encoder-MI	SD-MI	Fe-MI	LpLA(ours)
CIFAR-10	MoCo-v1	0.494(0.098)	0.501(0.101)	0.497(0.099)	0.496(0.099)
	MoCo-v2	0.607(0.146)	0.718(0.253)	0.692(0.201)	**0.723**(**0.259**)
	MoCo-v3	**0.960**(**3.181**)	0.914(1.045)	0.910(1.075)	0.907(0.790)
CIFAR-100	MoCo-v1	0.512(0.104)	0.507(0.105)	0.506(0.102)	0.496(0.098)
	MoCo-v2	0.627(0.173)	**0.783**(0.326)	0.750(0.318)	0.765(**0.415**)
	MoCo-v3	0.946(0.919)	0.915(**2.377**)	0.953(2.216)	**0.968**(1.795)
Tiny-Imagenet	MoCo-v1	0.507(0.103)	0.496(0.099)	0.498(0.099)	0.501(0.103)
	MoCo-v2	0.583(0.142)	**0.777**(0.384)	0.754(0.347)	0.769(**0.436**)
	MoCo-v3	0.965(3.368)	0.979(**5.858**)	0.976(4.285)	**0.982**(3.279)

Assuming there are equal prior probabilities $P(\mathrm{m}) = P(\mathrm{nm}) = 0.5$, the decision criterion simplifies to:

$$P(\mathrm{m} \mid x) > 0.5 \iff \mathcal{L}_{\mathrm{m}}(x) > \mathcal{L}_{\mathrm{nm}}(x). \tag{15}$$

Takeaway 3: The magnitude of the feature vector from members and non-members in encoders can be significantly different, which can be leveraged for membership inference.

4.3 Experiment Results of the Proposed Attack

We now address **RQ4**: "What are the advantages of the proposed attack compared to existing methods?" To answer this question, we first compare the effectiveness of our proposed attack against the baseline attacks of Encoder-MI, SD-MI, and Fe-MI to show the effectiveness of our attack while with the lightweight advantage. Then, we compare our attack with baseline attacks under the practical setting where the adversary has a partial training dataset and only black-box API query access to the target encoder model. This aims to show the applicability advantage of our attack over the baseline attacks.

Attack effectiveness. We first show the MI attack performance against encoders in Table 3, which is conducted on the ResNet-50 backbone across three different datasets and MoCo frameworks. As we can see, LpLA demonstrates sufficiently competitive attack performance across various dataset experiments. Unfortunately, LpLA, like existing methods, is unable to effectively infer membership status in the MoCo-v1 stage. However, under the MoCo-v2 framework, LpLA starts to show attack performance on par with classic feature-based and similarity-based methods. As the contrastive learning framework is further updated, we can observe that LpLA achieves the best attack performance on both CIFAR-100 and Tiny-Imagenet.

Additionally, an interesting phenomenon is observed that in several attack scenarios where LpLA does not achieve the best performance, SD-MI shows higher accuracy than other attack methods under the MoCo-v2 framework, and it's later surpassed by LpLA when the framework is updated to MoCo-v3. During the update, there is indeed a certain characteristic of the output features that was more straightforwardly captured by LpLA, thus bridging the gap between LpLA and similarity-based attacks.

We also implement our LpLA against the official pre-trained encoder for validation. Due to different training settings, we did not compare the experimental results with those based on our locally trained target models. Table 4 shows that LpLA also achieves strong attack performance against the official pre-trained encoder.

Table 4: LpLA against official pre-trained ResNet-50.

Model	Epochs	p=0	p=1	p=2	p=3
MoCo-v2(official)	800	0.732	**0.779**	0.688	0.681
MoCo-v3(official)	300	0.535	0.539	0.609	**0.674**

Attack applicability. Compared to existing attack approaches, LpLA does not require training neural networks as the binary attack classifier. Consequently, LpLA demands far fewer samples for constructing an attack model compared to neural network-based methods (e.g., EncoderMI, SD-MI, and Fe-MI). This is because LpLA directly estimates the likelihood of p-norm of feature vectors, leveraging statistical methods to infer differences in data distributions. Moreover, LpLA is computationally efficient. Traditional neural network-based attacks often involve higher resource demands for optimization, sometimes requiring additional queries and similarity calculations, resulting in significant computational costs. In contrast, LpLA only involves p-norm computation and distribution estimation after queries, making it lightweight and computationally efficient. The fewer attack requirements make LpLA more applicable in different attack scenarios than the existing baseline attacks.

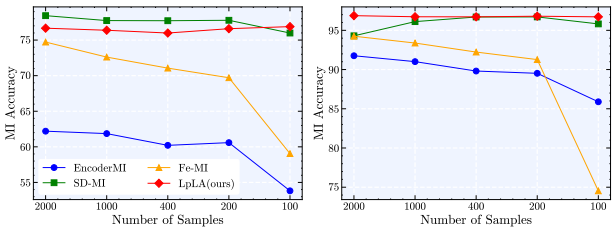
We qualitatively compare our attack with existing baseline attacks in Table 5. As we can see, LpLA reduces both sample and computational requirements, lowering the technical barrier for the adversary while broadening the applicability of membership inference attacks. We also quantitatively evaluate the attack performance of the scenarios where the number of member samples and model query access is limited, as follows.

• **Attack with limited member samples.** In real-world scenarios, an adversary often faces various limitations due to insufficient information about the target model, such as the inability to obtain enough ground-truth member samples, which can significantly weaken the attack's effectiveness. Furthermore, in the context of MIA against encoder models, because the size of unlabeled datasets used for self-supervised training is often enormous, it's difficult to collect enough shadow datasets that are identically distributed to the training set to train shadow models. Therefore, the number of ground-truth member samples that an adversary can obtain becomes a significant challenge.

Figure 6 shows the attack performance achieved by different attacks with different numbers of ground-truth member samples. The experimental results indicate that among all attack methods, LpLA demonstrates sufficiently competitive MIA performance (in most cases being the best attack method, and in a few cases being the second-best method), showing the superiority of LpLA in attack effectiveness. Additionally, the experimental results also show that when the size of the ground-truth member set held by the adversary decreases, the attack effectiveness also tends to decline. However, among these four methods, LpLA exhibits the most robust attack performance, as even when the data proportion is significantly reduced, the degree of decline in LpLA's attack effectiveness is still the smallest.

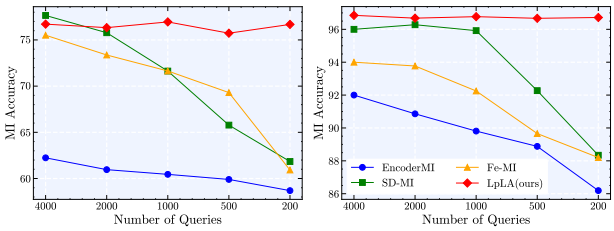
Table 5: Qualitatively comparison between our attack with existing baseline attacks.

Attacker	Ground-truth samples	Query volumes	Computational requirement	Other
Encoder-MI	$2 * 2k$	$2 * 10 * 2k$	Similarity & MLP-training	Data augmentation
SD-MI	$2 * 2k$	$2 * 2k + 2k(Anchors)$	Similarity & Two-MLP-training	Auxiliary dataset
FE-MI	$2 * 2k$	$2 * 2k$	MLP-training	(None)
LpLA(Ours)	$2 * 2k$	$2 * 2k$	Lp-norm & Likelihood-estimation	(None)



(a) ResNet-50 + MoCo-v2 (b) ResNet-50 + MoCo-v3

Figure 6: An illustration of the attack performance with limited attack dataset. There is a significant decreasing trend of EncoderMI and Fe-MI when gradually reducing the scale of the attack dataset, while SD-MI and LpLA perform relatively more robustly.



(a) ResNet-50 + MoCo-v2 (b) ResNet-50 + MoCo-v3

Figure 7: An illustration of the attack performance with limited API query volumes. There is only LpLA performs robust and efficiently under gradually limited API query volumes, while other attacks perform insufficiently.

• **Attack with limited query volumes.** Furthermore, we also consider that in real-world scenarios, an adversary often has to bear expensive query costs when accessing black-box models, or a privacy-aware victim may limit the number of API queries. Therefore, we further consider the attack performance that each method can achieve under different query scale settings when the adversary faces a limited number of queries to the target model.

Figure 7 shows the attack performance achieved when four different attack methods are used with a limited number of queries. It can be observed that when the API query numbers are severely limited, existing methods exhibit a more severe decline trend compared to the cases when the number of ground-truth member samples is limited. It's because when using multiple data augmentation samples or anchors to construct similarity-based attack features, more

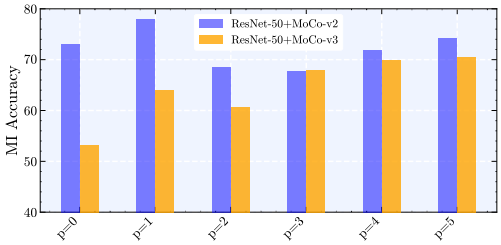


Figure 8: An illustration of the performance of LpLA against ResNet-50 + MoCo-v2&3 with different p. There is an insufficient performance when $p = 0$, and shows similarly satisfying performance with other p.

API query costs need to be paid, leading existing methods to be more easily affected by the constrained conditions in this scenario.

On the other hand, LpLA can still achieve effective attack performance, under conditions where the adversary's knowledge is highly limited. This is because when LpLA conducts inference attacks, it's based on likelihood estimation, and mainly relies on the differences in data statistical distribution. Thus, LpLA estimating the distributions with likelihood does not require as many samples as training a neural network, which allows it to demonstrate stronger adaptability in knowledge-limited scenarios.

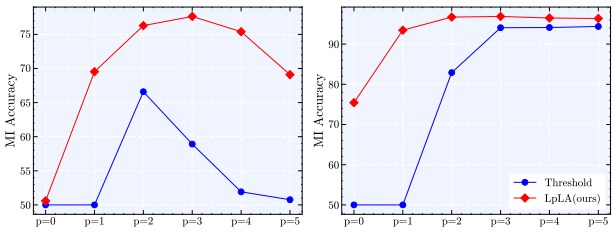
Takeaway 4: Being lightweight, the proposed attack, LpLA, has comparable attack performance against existing baseline attacks. In scenarios where only a limited number of member samples and query volumes are available, LpLA outperforms existing baseline attacks on encoder models.

4.4 Ablation

We further conduct ablation study experiments to evaluate how the choice of p-norm and distribution likelihood can affect the effectiveness of the proposed attack.

p-value selection. Figure 8 shows the results of LpLA on ResNet-50 pre-trained in MoCo-v2 and MoCo-v3 frameworks with various p-values. By comparing the attack performance across norms from p=0 to 5, we can see that using the L0-norm of feature vectors yields highly suboptimal attack results. However, as the p-value increases, the attack accuracy improves correspondingly. When the p-value reaches 2, the attack accuracy saturates, showing no significant improvement with further increases in p.

Distribution likelihood estimation. Figure 9 shows the results when simply use p-value as a threshold to perform MIA against encoder pre-trained in MoCo-v2 and MoCo-v3 frameworks with



(a) MI against MoCo-v2 (b) MI against MoCo-v3

Figure 9: An illustration of the comparison between using p-norm for estimation and threshold-based MI against ResNet-50 + MoCo-v2&3. LpLA shows a significant advantage over threshold-based attack, because LpLA uses not only the information of p-norm's mean, but uses estimated variance which helps attack more accurately and robustly.

various p-values. We perform a simple threshold-based MI here, where the mean p-value of member and non-member is calculated like equation (13). Comparing LpLA and naive threshold-based MI, it can be observed that likelihood estimation helps capture the information of attack signals more accurately and robustly.

4.5 Limitations

Despite LpLA demonstrating superior attack performance and robustness across diverse datasets and model architectures, several limitations warrant consideration: (1) The resilience of LpLA against existing defense mechanisms (e.g., differential privacy or adversarial regularization) remains unexamined, leaving its robustness under countermeasures uncertain; (2) Experiments are confined to image datasets (e.g., CIFAR, Tiny-ImageNet), with no validation on text or audio modalities, limiting insights into cross-domain applicability. Future work will assess defenses' impacts on SSL's performance-privacy trade-off and expand research to text/audio modalities to ensure generalizable analysis.

5 Conclusion

We delved deeply into the issue of membership privacy in this work, especially focusing on the issue of membership inference attacks against encoder models within contrastive learning. It comprehensively reveals the impact of different model architectures on the leakage of membership information through both theoretical analysis and experimental validation. Based on this, we propose a membership inference attack method named LpLA (Embedding Lp-Norm Likelihood Attack), which is motivated by the different distributions of member and non-member's embedding p-norm values. Experimental results indicate that more complex model architectures, while enhancing the feature extraction capabilities of encoder models, also exacerbate the risk of membership privacy leakage. Furthermore, this research demonstrates the effectiveness of using the p-norm of feature vectors in MI attacks. LpLA not only matches or even surpasses existing methods in performance, but most importantly, LpLA exhibits greater robustness, especially under conditions where the adversary's knowledge is severely limited.

This research not only broadens the scope of investigations into membership privacy issues in contrastive learning but also lays a solid foundation for future studies on privacy protection. It is hoped that this work will inspire more attention to the privacy risks associated with self-supervised learning models, such as performing privacy attacks and implementing protections across a wider range of deep learning models.

Acknowledgments

Haizhuan's work was supported in part by National Natural Science Foundation of China (No. 12271464), Hunan Provincial Natural Science Foundation of China (No. 2023JJ10038), the Innovative Research Group Project of Natural Science Foundation of Hunan Province of China (No. 2024JJ1008). We also gratefully acknowledge the support of the High-Performance Computing Platform of Xiangtan University for providing computational resources used in this work.

References

[1] Hangbo Bao, Li Dong, Songhao Piao, and Furu Wei. 2021. BEiT: BERT Pre-Training of Image Transformers. *arXiv preprint arXiv:2106.08254* (2021).
[2] Nicholas Carlini, Steve Chien, Milad Nasr, Shuang Song, Andreas Terzis, and Florian Tramer. 2022. Membership inference attacks from first principles. In *2022 IEEE Symposium on Security and Privacy (SP)*. IEEE, 1897–1914.
[3] Mathilde Caron, Ishan Misra, Mairal, et al. 2020. Unsupervised learning of visual features by contrasting cluster assignments. *Advances in Neural Information Processing Systems* 33 (2020), 9912–9924.
[4] Harsh Chaudhari, John Abascal, Alina Oprea, Matthew Jagielski, Florian Tramer, and Jonathan Ullman. 2023. SNAP: Efficient extraction of private properties with poisoning. In *2023 IEEE Symposium on Security and Privacy (SP)*. IEEE, 400–417.
[5] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. 2020. A simple framework for contrastive learning of visual representations. In *International Conference on Machine Learning*. PMLR, 1597–1607.
[6] Xinlei Chen, Haoqi Fan, Ross Girshick, and Kaiming He. 2020. Improved baselines with momentum contrastive learning. *arXiv preprint arXiv:2003.04297* (2020).
[7] Xinlei Chen and Kaiming He. 2021. Exploring simple siamese representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 15750–15758.
[8] Xinlei Chen, Saining Xie, and Kaiming He. 2021. An empirical study of training self-supervised vision transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 9640–9649.
[9] Emiliano De Cristofaro. 2020. An overview of privacy in machine learning. *arXiv preprint arXiv:2005.08679* (2020).
[10] Alexey Dosovitskiy. 2020. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929* (2020).
[11] Junyao Gao, Xinyang Jiang, Huishuai Zhang, et al. 2023. Similarity distribution based membership inference attack on person re-identification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 37. 14820–14828.
[12] Spyros Gidaris, Praveer Singh, and Nikos Komodakis. 2018. Unsupervised representation learning by predicting image rotations. *arXiv preprint arXiv:1803.07728* (2018).
[13] Priya Goyal, Dhruv Mahajan, Abhinav Gupta, and Ishan Misra. 2019. Scaling and benchmarking self-supervised visual representation learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 6391–6400.
[14] Jean-Bastien Grill, Florian Strub, Altché, et al. 2020. Bootstrap your own latent-a new approach to self-supervised learning. *Advances in Neural Information Processing Systems* 33 (2020), 21271–21284.
[15] Jie Gui, Tuo Chen, Jing Zhang, Qiong Cao, Zhenan Sun, Hao Luo, and Dacheng Tao. 2024. A Survey on Self-supervised Learning: Algorithms, Applications, and Future Trends. *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2024).
[16] Umang Gupta, Dimitris Stripelis, Pradeep K Lam, Paul Thompson, Jose Luis Ambite, and Greg Ver Steeg. 2021. Membership inference attacks on deep regression models for neuroimaging. In *Medical Imaging with Deep Learning*. PMLR, 228–251.
[17] Raia Hadsell, Sumit Chopra, and Yann LeCun. 2006. Dimensionality reduction by learning an invariant mapping. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, Vol. 2. IEEE, 1735–1742.
[18] Jamie Hayes, Luca Melis, George Danezis, and Emiliano De Cristofaro. 2017. Logan: Membership inference attacks against generative models. *arXiv preprint*

arXiv:1705.07663 (2017).

[19] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. 2022. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 16000–16009.

[20] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. 2020. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 9729–9738.

[21] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 770–778.

[22] Hongsheng Hu, Zoran Salcic, Lichao Sun, Gillian Dobbie, Philip S Yu, and Xuyun Zhang. 2022. Membership inference attacks on machine learning: A survey. *ACM Computing Surveys (CSUR)* 54, 11s (2022), 1–37.

[23] Malhar S Jere, Tyler Farnan, and Farinaz Koushanfar. 2020. A taxonomy of attacks on federated learning. *IEEE Security & Privacy* 19, 2 (2020), 20–28.

[24] Alex Krizhevsky, Geoffrey Hinton, et al. 2009. Learning multiple layers of features from tiny images. (2009).

[25] Gustav Larsson, Michael Maire, and Gregory Shakhnarovich. 2017. Colorization as a proxy task for visual understanding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 6874–6883.

[26] Ya Le and Xuan S. Yang. 2015. Tiny ImageNet Visual Recognition Challenge. https://api.semanticscholar.org/CorpusID:16664790

[27] Shuhao Li, Yajie Wang, Yuanzhang Li, and Yu-an Tan. 2022. l-Leaks: Membership inference attacks with logits. *arXiv preprint arXiv:2205.06469* (2022).

[28] Zheng Li, Xinlei He, Ning Yu, and Yang Zhang. 2024. Membership Inference Attack Against Masked Image Modeling. *arXiv preprint arXiv:2408.06825* (2024).

[29] Zheng Li and Yang Zhang. 2021. Membership leakage in label-only exposures. In *Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security*. 880–895.

[30] Hongbin Liu, Jinyuan Jia, Wenjie Qu, et al. 2021. EncoderMI: Membership inference against pre-trained encoders in contrastive learning. In *Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security*. 2081–2095.

[31] Yugeng Liu, Rui Wen, He Xinlei, et al. 2022. ML-Doctor: Holistic risk assessment of inference attacks against machine learning models. In *31st USENIX Security Symposium (USENIX Security 22)*. 4525–4542.

[32] Mengyao Ma, Yanjun Zhang, Arachchige, et al. 2023. Loden: Making every client in federated learning a defender against the poisoning membership inference attacks. In *Proceedings of the 2023 ACM Asia Conference on Computer and Communications Security*. 122–135.

[33] Shagufta Mehnaz, Sayanton V Dibbo, Ehsanul Kabir, Ninghui Li, and Elisa Bertino. 2022. Are your sensitive attributes private? Novel model inversion attribute inference attacks on classification models. In *31st USENIX Security Symposium (USENIX Security 22)*. 4579–4596.

[34] Luca Melis, Congzheng Song, Emiliano De Cristofaro, and Vitaly Shmatikov. 2019. Exploiting unintended feature leakage in collaborative learning. In *2019 IEEE Symposium on Security and Privacy (SP)*. IEEE, 691–706.

[35] T Nathan Mundhenk, Daniel Ho, and Barry Y Chen. 2018. Improvements to context based self-supervised learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 9339–9348.

[36] Milad Nasr, Reza Shokri, and Amir Houmansadr. 2019. Comprehensive privacy analysis of deep learning: Passive and active white-box inference attacks against centralized and federated learning. In *2019 IEEE Symposium on Security and Privacy (SP)*. IEEE, 739–753.

[37] Shahbaz Rezaei and Xin Liu. 2021. On the difficulty of membership inference attacks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 7892–7900.

[38] Ahmed Salem, Yang Zhang, Humbert, et al. 2018. ML-leaks: Model and data independent membership inference attacks and defenses on machine learning models. *arXiv preprint arXiv:1806.01246* (2018).

[39] Reza Shokri, Marco Stronati, Congzheng Song, and Vitaly Shmatikov. 2017. Membership inference attacks against machine learning models. In *2017 IEEE Symposium on Security and Privacy (SP)*. IEEE, 3–18.

[40] Congzheng Song and Ananth Raghunathan. 2020. Information leakage in embedding models. In *Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security*. 377–390.

[41] Liwei Song and Prateek Mittal. 2021. Systematic evaluation of privacy risks of machine learning models. In *30th USENIX Security Symposium (USENIX Security 21)*. 2615–2632.

[42] Igor Susmelj, Matthias Heller, Philipp Wirth, Jeremy Prescott, Malte Ebner, and et al. 2020. Lightly. https://github.com/lightly-ai/lightly [Accessed: 2025-01-01].

[43] Florian Tramèr, Reza Shokri, Ayrton San Joaquin, Hoang Le, Matthew Jagielski, Sanghyun Hong, and Nicholas Carlini. 2022. Truth serum: Poisoning machine learning models to reveal their secrets. In *Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security*. 2779–2792.

[44] Tobias Uelwer, Jan Robine, Stefan Sylvius Wagner, Marc Höftmann, Eric Upschulte, Sebastian Konietzny, Maike Behrendt, and Stefan Harmeling. 2023. A Survey on Self-Supervised Representation Learning. *arXiv preprint arXiv:2308.11455* (2023).

[45] Zhirong Wu, Yuanjun Xiong, Stella X Yu, and Dahua Lin. 2018. Unsupervised feature learning via non-parametric instance discrimination. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 3733–3742.

[46] Zhenda Xie, Zheng Zhang, Yue Cao, et al. 2022. SimMIM: a Simple Framework for Masked Image Modeling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 9653–9663.

[47] Xiaomei Zhang, Zhaoxi Zhang, Qi Zhong, Xufei Zheng, Yanjun Zhang, Shengshan Hu, and Leo Yu Zhang. 2023. Masked Language Model Based Textual Adversarial Example Detection. In *ASIA CCS*. 925–937.

[48] Zhaoxi Zhang, Leo Yu Zhang, Xufei Zheng, Bilal Hussain Abbasi, and Shengshan Hu. 2022. Evaluating membership inference through adversarial robustness. *Comput. J.* 65, 11 (2022), 2969–2978.

[49] Zhaoxi Zhang, Leo Yu Zhang, Xufei Zheng, Jinyu Tian, and Jiantao Zhou. 2022. Self-supervised adversarial example detection by disentangled representation. In *2022 IEEE International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom)*. IEEE, 1000–1007.

[50] Jinghao Zhou, Chen Wei, Huiyu Wang, Wei Shen, Cihang Xie, Alan Yuille, and Tao Kong. 2021. iBOT: Image BERT Pre-Training with Online Tokenizer. *arXiv preprint arXiv:2111.07832* (2021).

[51] Jie Zhu, Jirong Zha, Ding Li, et al. 2024. A unified membership inference method for visual self-supervised encoder via part-aware capability. In *Proceedings of the 2024 on ACM SIGSAC Conference on Computer and Communications Security*. 1241–1255.

[52] Ligeng Zhu, Zhijian Liu, and Song Han. 2019. Deep leakage from gradients. *Advances in Neural Information Processing Systems* 32 (2019).