

Dataset Ownership Verification for Pre-trained Masked Models

Supplementary Material

A. The Details of Experiments

A.1. Datasets Used

ImageNet-1K [11]: A large-scale visual dataset containing over 14 million colored images across 1000 classes. As is commonly done, we resize all images to be of size 224x224. The specific categories we use in ImageNet-50 and ImageNet-100 are listed in ImageNet-50.txt and ImageNet-100.txt in the supplementary material.

Food101 [2]: A dataset contains images of 101 different food categories, with 1,000 images per category, primarily used for food recognition tasks.

COCO [32]: A dataset includes over 328,000 images across 80 object categories, and is widely used for tasks such as object detection, segmentation, and captioning.

Places365 [56]: A dataset contains 365 different scene categories, aimed at improving the accuracy of scene classification and understanding, with over 1.8 million images.

WikiText-103 [40]: A large-scale dataset derived from Wikipedia articles, containing over 100 million tokens. It is primarily used for language modeling and text generation tasks, focusing on maintaining high-quality, real-world text.

CC-News [19]: A dataset consists of news articles collected from the Common Crawl web corpus. It includes over 300 million English news articles, and is commonly used for tasks such as news classification, topic modeling and so on.

MiniPile [24]: MiniPile is a smaller, curated subset of the Pile dataset [16], designed for training large-scale language models. It contains diverse text data across multiple domains, such as books, academic papers, and web pages, to improve the generalization of NLP models.

A.2. Pre-trained MIM Models Used

The pre-trained models we use on ImageNet-1K all come from the official repository or MMSelfSup, as follows:

MAE: <https://github.com/facebookresearch/mae>.

CAE: <https://github.com/lxtGH/CAE>.

iBOT: <https://github.com/bytedance/ibot>.

SimMIM: <https://github.com/microsoft/SimMIM>.

Other Models: https://readthedocs.io/en/latest/model_zoo.html

A.3. Parameter Setting Details

- **Training of masked image modeling (MIM) models.** All MIM models are trained for 400 epochs with a batch size of 64, and the learning rate, masking ratio and optimizer parameters are set to the default settings for each respective method.
- **Training of masked language modeling (MLM) models.** All MLM models are trained for 400 epochs with a batch size of 64, a maximum input sequence length of 128, a masking ratio of 20%, a learning rate of 5e-5, an optimizer of Adam, and the mask token is [MASK]. Other parameters are set to the default settings for each respective method.
- **Decoder and its training.** Decoder is a Transformer [49] with an embedding dimension of 512, 8 layers, and 16 heads in the multi-head attention mechanism. The training set for the decoder consists of 20,000 randomly sampled examples from the public dataset. The training of the decoder is performed for 50 epochs with a batch size of 64, a learning rate of 1e-3, and the Adam optimizer. The input masking ratio is 75% (20% in NLP). The patch size of the mask is 16×16 .
- **Validation phase.** The validation set consists of the remaining portion of the public dataset after excluding the decoder's training set. The sampling frequency is 30, with 1,024 samples per sampling. During inference, the batch size is 256. In the t-test, the significance level α is 0.05. The ratio and the patch size of the random masking is the same as when training the decoder.
- **Fine-tuning using MIM method.** The hyperparameters used for fine-tuning are the same as those used during pre-training, with the fine-tuning epoch set to 50.
- **Fine-tuning on downstream task.** We add a fully connected layer (classification head) at the output end of the encoder for downstream classification tasks. During fine-tuning, only the classification head is fine-tuned with 50 epochs, a learning rate of 1e-3, the Adam optimizer, and a batch size of 256.

A.4. Hypothesis Testing

In our experiment, the null hypothesis H_0 and the alternative hypothesis H_1 for the one-sided pair-wise t-test are as follows:

- H_0 : The mean difference between the paired samples in $\Delta\mathcal{R}_{pt}$ and $\Delta\mathcal{R}_{vt}$ is less than or equal to 0.
- H_1 : The mean difference between the paired samples in $\Delta\mathcal{R}_{pt}$ and $\Delta\mathcal{R}_{vt}$ is greater than 0.

Our hypothesis testing can be divided into the following

three steps:

1. **Calculate the t -statistic.** First, calculate the mean paired differences between the elements in $\Delta\mathcal{R}_{pt}$ and $\Delta\mathcal{R}_{vt}$:

$$\bar{d} = \frac{1}{K} \sum_{k=1}^K (\Delta\mathcal{R}_{pt}^k - \Delta\mathcal{R}_{vt}^k) \quad (4)$$

where K is the iterations of sampling, $\Delta\mathcal{R}_{pt}^k$ and $\Delta\mathcal{R}_{vt}^k$ are the k -th elements in $\Delta\mathcal{R}_{pt}$ and $\Delta\mathcal{R}_{vt}$, respectively. Then, calculate the standard deviation of the differences:

$$s_d = \sqrt{\frac{1}{K-1} \sum_{k=1}^K (d_k - \bar{d})^2} \quad (5)$$

where $d_k = \Delta\mathcal{R}_{pt}^k - \Delta\mathcal{R}_{vt}^k$. Finally, calculate the t -statistic:

$$t = \frac{\bar{d}}{s_d/\sqrt{K}} \quad (6)$$

The t -statistic t follows a t -distribution with $K - 1$ degrees of freedom.

2. **Calculate the p -value.** We first calculate the p -value for the two-tailed test:

$$p = 2P(T > |t|) \quad (7)$$

Then, based on the t -statistic, the final p -value is determined. When the t -statistic is greater than 0, $p = p/2$. When the t -statistic is less than 0, $p = 1 - p/2$.

3. **Determine significance.** Our significance level α is set to 0.05. If $p \leq \alpha$, we reject the null hypothesis H_0 and consider the suspicious model is illegal. If $p > \alpha$, we fail to reject the null hypothesis and consider the suspicious model is legal.

The one-sided pair-wise t -test we use can be easily implemented in Python with just a few lines of code, as shown in Algorithm 1. Here, t is the t -statistic and p is the p -value. When the output p -value is less than 0.05, we conclude that $\Delta\mathcal{R}_{pt}$ is significantly greater than $\Delta\mathcal{R}_{vt}$, meaning M_s illegally used \mathcal{D}_{pub} for training. Conversely, when the p -value is greater than 0.05, we conclude that $\Delta\mathcal{R}_{pt}$ is not significantly greater than $\Delta\mathcal{R}_{vt}$, meaning M_s is legal.

Algorithm 1 One-tailed pair-wise t -test

- 1: **Input:** $\Delta\mathcal{R}_{vt}$, $\Delta\mathcal{R}_{pt}$
 - 2: import scipy
 - 3: $t, p = \text{scipy.state.ttest_ind}(\Delta\mathcal{R}_{pt}, \Delta\mathcal{R}_{vt})$
 - 4: **if** $t > 0$ **then**
 - 5: $p = p/2$
 - 6: **else**
 - 7: $p = 1 - p/2$
 - 8: **end if**
 - 9: **Output:** p
-

B. Efficiency Analysis

We calculated the time required for DOV4MM and three other baselines to perform one validation on ImageNet-1K and Places365. The experiment was conducted using an NVIDIA GeForce RTX 4090. The suspicious model is a ViT-B/16 pre-trained on ImageNet-1K using MAE. Here, the settings of DOV4MM is: the decoder M_d has depth 4, width 128 and attention heads 4, the size of the training dataset for M_d is 10,000. Sampling iterations K and the number of samples per iteration N are 10 and 512, respectively. The training epoch for M_d is 5. As shown in Tab. S1, only DOV4MM achieved correct results while maintaining a certain level of efficiency. Compared to DI4SSL, we do not need to infer the entire dataset, thus achieving a performance advantage in efficiency. In contrast to CTRL and PartCrop, DOV4MM consumed more time because it requires training a decoder, but it can extract the most valuable relative embedding reconstruction difficulty from redundant representations to obtain correct validation results, which CTRL and PartCrop cannot achieve.

Method	$\bar{\tau}$	\mathcal{D}_{pub}	
		ImageNet-1K	Places365
DI4SSL	10034s	1.00	0.84
CTRL	247s	10^{-150}	10^{-140}
PartCrop	128s	0.71	0.77
DOV4MM	353s	10^{-3}	0.41

Table S1. Efficiency analysis. $\bar{\tau}$ is the average time taken by each method to perform one validation on ImageNet-1K and Places365. The illegal (p should be less than 0.05) and legal (p should be greater than 0.05) cases correspond to the red and blue areas.

C. The Interference Resistance of DOV4MM

Fine-tuning M_s using MIM methods. Given M_s pre-trained on ImageNet-1K, we fine-tune it using the same MIM method on another dataset, then perform DOV4MM on these fine-tuned models. \mathcal{D}_{pub} is ImageNet-1K, and the p -value needs to be less than 0.05. As shown in Tab. S2, DOV4MM is still valid in this more arduous scenario.

Adaptive attack. We assume that the training objective of M_s consists of two equally weighted components: (1) the proxy task, (2) a loss L that minimizes the difference in reconstruction difficulty between seen and unseen samples. M_s are ViT-B pre-trained with different subsets of ImageNet-1K (\mathcal{D}_{pub}). Tab. S3 indicates that DOV4MM remains effective. This is because, although L narrows the reconstruction gap between seen and unseen samples, the gap remains due to the proxy task.

\mathcal{D}_f	MIM Method	Model	w/o ft	w/ ft
Food101	MAE	ViT-B/16	10^{-5}	10^{-5}
		ViT-L/16	10^{-5}	10^{-4}
	iBOT	ViT-B/16	10^{-3}	10^{-3}
		ViT-L/16	10^{-3}	10^{-3}
Places365	MAE	ViT-B/16	10^{-5}	10^{-5}
		ViT-L/16	10^{-5}	10^{-4}
	iBOT	ViT-B/16	10^{-3}	10^{-3}
		ViT-L/16	10^{-3}	10^{-3}

Table S2. We report the p -values of M_s , whose pre-training dataset is ImageNet-1K, with or without fine-tuning (ft). \mathcal{D}_f represents the fine-tuning dataset.

MIM Methods	w/o L	w/ L	MIM Methods	w/o L	w/ L
MAE	10^{-9}	10^{-9}	MAE	10^{-5}	10^{-5}
CAE	10^{-10}	10^{-8}	CAE	10^{-5}	10^{-4}

(a) \mathcal{D}_{pub} is ImageNet-50.

(b) \mathcal{D}_{pub} is ImageNet-100.

Table S3. p (should < 0.05) under different \mathcal{D}_{pub} .

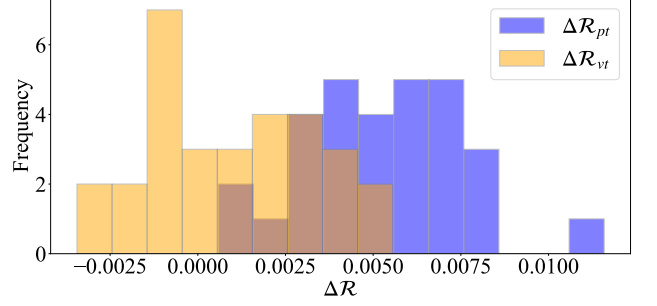
Model	MIM Method	\mathcal{D}_{pub}			
		IN-50	Food101	COCO	Places365
ViT-B/16	MAE	0.56	0.99	0.28	0.46
	CAE	0.99	1.00	0.94	0.90
	iBOT	0.09	0.90	0.41	0.48
ViT-L/16	MAE	0.31	0.91	0.40	0.59
	CAE	0.99	1.00	0.66	0.96
	iBOT	0.31	0.91	0.41	0.59

Table S4. The results (p -values) of DI4SSL on ImageNet-50. “IN-50” represents ImageNet-50.

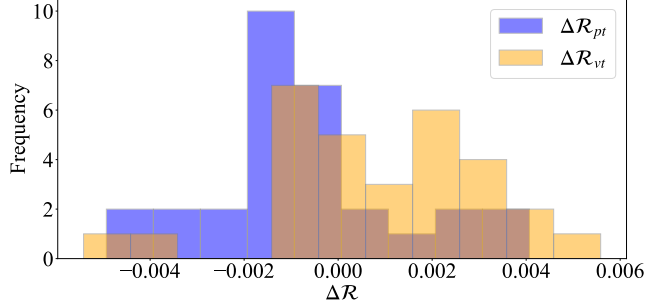
D. Visualization Results

We present the visualization results of DOV4MM on the ImageNet-50 pre-trained model. Specifically, \mathcal{D}_{pub} is set to ImageNet-50, Food101, COCO, or Places365, and the suspicious model is a ViT-L/16 pre-trained using MAE. The pre-training dataset of suspicious model is ImageNet-50. We computed the relative embedding reconstruction difficulties from suspicious models. Specifically, they include the relative reconstruction difficulty of embeddings between the training set \mathcal{D}_t and the validation set \mathcal{D}_v ($\Delta\mathcal{R}_{vt}$), as well as the relative reconstruction difficulty of embeddings between the training set \mathcal{D}_t and the test set \mathcal{D}_{pvt} ($\Delta\mathcal{R}_{pt}$), which are then visualized. When the suspicious model is pre-trained on \mathcal{D}_{pub} , it is considered illegal, and $\Delta\mathcal{R}_{pt}$ should be generally higher than $\Delta\mathcal{R}_{vt}$. In contrast, if the suspicious model is legal, this phenomenon is not obvious.

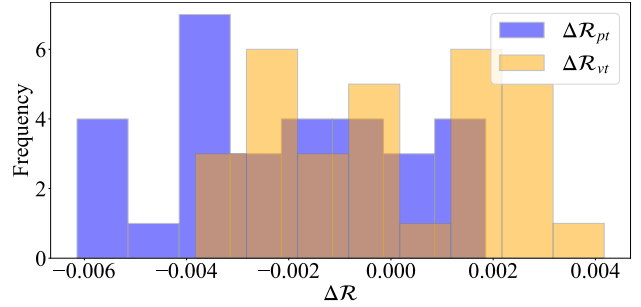
As shown in Fig. 5, when the suspicious model is illegal, $\Delta\mathcal{R}_{pt}$ is generally higher than $\Delta\mathcal{R}_{vt}$. When the suspicious model is legal, there is no such relationship between the two relative embedding reconstruction difficulties. This observation is consistent with our previous findings.



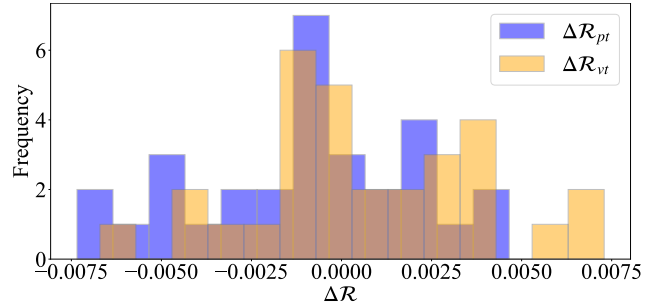
(a) \mathcal{D}_{pub} is ImageNet-50. (M_s is illegal)



(b) \mathcal{D}_{pub} is Food101. (M_s is legal)



(c) \mathcal{D}_{pub} is COCO. (M_s is legal)



(d) \mathcal{D}_{pub} is Places365. (M_s is legal)

Figure 5. The distribution of relative embedding reconstruction difficulties $\Delta\mathcal{R}_{vt}$ and $\Delta\mathcal{R}_{pt}$. Note that when \mathcal{D}_{pub} is ImageNet-50, M_s is considered illegal, while in all other cases, it is legal.

E. The Specific p -values in Fig. 3 and Fig. 4

Here, we present the specific results of different methods on various datasets (as shown in Fig. 3 of the paper), as shown in Tab. S4 - Tab. S11. Note that the illegal (p should

Model	MIM Method	\mathcal{D}_{pub}			
		IN-50	Food101	COCO	Places365
ViT-B/16	MAE	10^{-155}	10^{-113}	10^{-134}	10^{-140}
	CAE	10^{-284}	10^{-148}	10^{-246}	10^{-255}
	iBOT	10^{-155}	10^{-113}	10^{-134}	10^{-140}
ViT-L/16	MAE	10^{-160}	10^{-122}	10^{-140}	10^{-152}
	CAE	10^{-199}	10^{-148}	10^{-171}	10^{-151}
	iBOT	10^{-160}	10^{-122}	10^{-140}	10^{-152}

Table S5. The results (p -values) of CTRL on ImageNet-50. “IN-50” represents ImageNet-50.

Model	MIM Method	\mathcal{D}_{pub}			
		IN-50	Food101	COCO	Places365
ViT-B/16	MAE	0.28	0.12	0.50	0.15
	CAE	0.06	0.48	0.25	10^{-3}
	iBOT	0.28	0.11	0.50	0.15
ViT-L/16	MAE	0.12	0.06	0.56	0.03
	CAE	0.16	0.26	0.34	0.09
	iBOT	0.12	0.06	0.56	0.01

Table S6. The results (p -values) of PartCrop on ImageNet-50. “IN-50” represents ImageNet-50.

Model	MIM Method	\mathcal{D}_{pub}			
		IN-50	Food101	COCO	Places365
ViT-B/16	MAE	10^{-9}	0.92	0.99	0.99
	CAE	10^{-10}	0.91	0.99	0.90
	iBOT	10^{-4}	0.99	0.99	0.97
ViT-L/16	MAE	10^{-12}	0.99	0.99	0.98
	CAE	10^{-9}	0.96	0.99	0.92
	iBOT	10^{-6}	0.99	0.99	0.89

Table S7. The results (p -values) of DOV4MM on ImageNet-50. “IN-50” represents ImageNet-50.

Model	MIM Method	\mathcal{D}_{pub}			
		IN-100	Food101	COCO	Places365
ViT-B/16	MAE	0.68	0.90	0.41	0.46
	CAE	0.99	0.99	0.98	0.53
	iBOT	0.79	0.91	0.41	0.46
ViT-L/16	MAE	0.83	0.83	0.41	0.38
	CAE	0.99	1.00	0.90	0.47
	iBOT	0.88	0.79	0.40	0.59

Table S8. The results (p -values) of DI4SSL on ImageNet-100. “IN-100” represents ImageNet-100.

Model	MIM Method	\mathcal{D}_{pub}			
		IN-100	Food101	COCO	Places365
ViT-B/16	MAE	10^{-150}	10^{-113}	10^{-134}	10^{-140}
	CAE	10^{-236}	10^{-129}	10^{-225}	10^{-235}
	iBOT	10^{-150}	10^{-113}	10^{-134}	10^{-140}
ViT-L/16	MAE	10^{-158}	10^{-122}	10^{-140}	10^{-152}
	CAE	10^{-173}	10^{-120}	10^{-165}	10^{-144}
	iBOT	10^{-158}	10^{-122}	10^{-140}	10^{-152}

Table S9. The results (p -values) of CTRL on ImageNet-100. “IN-100” represents ImageNet-100.

be less than 0.05) and legal (p should be greater than 0.05) scenarios correspond to the red and blue areas.

We also present the specific results of different methods

Model	MIM Method	\mathcal{D}_{pub}			
		IN-100	Food101	COCO	Places365
ViT-B/16	MAE	0.42	0.11	0.50	0.15
	CAE	0.10	0.27	0.28	10^{-3}
	iBOT	0.42	0.11	0.50	0.15
ViT-L/16	MAE	0.33	0.06	0.56	0.03
	CAE	0.19	0.33	0.38	0.01
	iBOT	0.33	0.06	0.56	0.03

Table S10. The results (p -values) of PartCrop on ImageNet-100. “IN-100” represents ImageNet-100.

Model	MIM Method	\mathcal{D}_{pub}			
		IN-100	Food101	COCO	Places365
ViT-B/16	MAE	10^{-5}	0.98	0.99	0.99
	CAE	10^{-5}	0.96	0.99	0.92
	iBOT	10^{-4}	0.99	0.96	0.97
ViT-L/16	MAE	10^{-6}	0.99	0.99	0.98
	CAE	10^{-4}	0.97	0.99	0.93
	iBOT	10^{-4}	0.99	0.99	0.87

Table S11. The results (p -values) of DOV4MM on ImageNet-100. “IN-100” represents ImageNet-100.

MLM Model	\mathcal{D}_{pub}		
	Wiki-50k	CC-News	MiniPile
BERT _{Base}	10^{-16}	0.53	0.99
BERT _{Large}	10^{-13}	0.46	0.93
RoBERTa _{Base}	10^{-18}	0.71	0.57
RoBERTa _{Large}	10^{-9}	0.68	0.74
ALBERT _{Base}	10^{-34}	0.43	0.54
XLM-R _{Base}	10^{-18}	0.32	0.13

Table S12. The results (p -values) of DOV4MM on WikiText-103-50k. “Wiki-50k” represents WikiText-103-50k.

MLM Model	\mathcal{D}_{pub}		
	Wiki-100k	CC-News	MiniPile
BERT _{Base}	10^{-10}	0.72	0.97
BERT _{Large}	10^{-10}	0.69	0.82
RoBERTa _{Base}	10^{-3}	0.48	0.58
RoBERTa _{Large}	10^{-12}	0.47	0.55
ALBERT _{Base}	10^{-9}	0.10	0.54
XLM-R _{Base}	10^{-7}	0.20	0.08

Table S13. The results (p -values) of DOV4MM on WikiText-103-100k. “Wiki-100k” represents WikiText-103-100k.

on various datasets (as shown in Fig. 4 of the paper), as shown in Tab. S12 and Tab. S13. Note that the illegal (p should be less than 0.05) and legal (p should be greater than 0.05) scenarios correspond to the red and blue areas.