A. Additional Results on WSA [Section 5.2]

In this section, we additionally present the accuracy metric obtained from trained attack models in Table 5. Given that it is infeasible to compute the accuracy values from CSA and AEA, we opted to showcase the AUC and TPR@1%FPR in the main paper to ensure a fair comparison between the three approaches. Nevertheless, as a crucial design preference, it may be imperative to take into account the balanced scenarios between the true negative and true positive cases. Therefore, we furnish the outcomes in terms of accuracy and provide an explanation of the true negative and true positive analyses in Appendix C.

Table 5. WSA performance in terms of accuracy obtained from attack models.

Method		WSA
Dataset	Model	ACC
LAION	ViTi-B/32	0.7845 ± 0.0122
	ViTi-B/16	0.7996 ± 0.0062
	ViTi-L/14	0.8078 ± 0.0100
CC12M	RN50	0.6978 ± 0.0050
	RN101	0.7140 ± 0.0163
	ViT-B/32	0.7005 ± 0.0022

B. Additional Results and Analysis on AEA [Section 5.2]

We additionally present AEA results on different models trained with different datasets (e.g., RN101, ViT-B/32 with CC12M and ViT-B/32, ViT-L/14 with LAION) in Figure 6. As shown in the figure, AEA surpasses CSA in all models trained with different datasets in terms of AUC and TPR@1%FPR.

Regarding the performance of AEA, we observe that on the pre-trained models trained with LAION, AEA exhibits the best performance with rotation and the worst performance with colorjitter. By contrast, for the RN50, and RN101 trained with CC12M, AEA performs best or second best with colorjitter and Masked Autoencoder (MAE) augmentations, except for the combined augmentation. These results suggest that pre-trained models with the LAION dataset (e.g., LAION ViT-B/16, ViT-B/32, and ViT-L/14) show robustness towards colorjitter, but are weak to rotation changes, while self-trained models with CC12M exhibit a weakness towards MAE and colorjitter.

C. Sensitivity Analysis on $|D_{\text{no}}|$ and mislabel ratio [Section 5.3]

In-depth Analysis on the Impact of $|D_{\text{no}}|$. Attack performance in terms of ACC increases until $|D_{\text{no}}| = 70K$ and decreases. The reason is that mere expansion of the size leads to including more noisy samples without providing useful alignment information between samples, which leads to a decrease in performance. Furthermore, regardless of the size of non-training data, our proposed method, WSA, consistently outperforms the baseline for both datasets and two different models.

In-depth Analysis on the Impact of Mislabeling Ratio. A comprehensive analysis of the results from Table 2 may raise the question of why TPR@1%FPR sometimes shows better performance with lower accuracy. For example, for the LAION ViT-L/14 pre-trained model, TPR@1%FPR is 0.7178 when $\lambda = -1.5$, but TPR@1%FPR drops to 0.6668 with higher attack model accuracy (i.e., 0.8199). Therefore, we provide analysis regarding the performance increase in terms of the TPR@1%FPR at the low threshold value even with the high mislabeled ratio. As depicted in Figure 7, selecting a lower threshold value (i.e., $\lambda = -1.5$) results in a less number of non-member samples at high confidence values (e.g., $2036 \rightarrow 1613$). This, in turn, leads to an increase in the number of true positive (TP) cases as described in the confusion matrices (e.g., $18121 \rightarrow 18814$). However, this advantage comes at the cost of sacrificing true negative (TN) simultaneously (e.g., $13423 \rightarrow 10632$). It becomes more challenging to correctly classify non-members, thus lowering the ACC score. In addition, as presented in the table, the baseline approach fails to achieve high accuracy even though it achieves relatively high TPR@1%FPR. In case attackers prioritize TPR@1%FPR, they may opt for a lower threshold. In sum, WSA provides superior performance, compared to CSA, for any threshold selection. Nevertheless, by carefully selecting λ, we can achieve a more balanced performance.

D. Additional Results on Defenses [Section 6]

In this section, we further present the defense results obtained from the RN50 models with data augmentation and L2 regularization. Even though [6] provides the pre-trained RN50 model on CC12M, to fairly compare the results, we train three RN50 models from scratch on CC12M (i.e., a model without defense, a model with DA, a model with L2 regularization). We use similar hyperparameter settings provided in the original paper. In particular, we set the number of epochs to 30, a learning rate to $1e-3$, and a weight decay for DA and original models to $1e-1$. The zero-shot accuracy on ImageNet is near 31% for the self-trained original RN50 model, and the accuracy from the pre-trained model is around 35%. The defense results are summarized in Table 7.

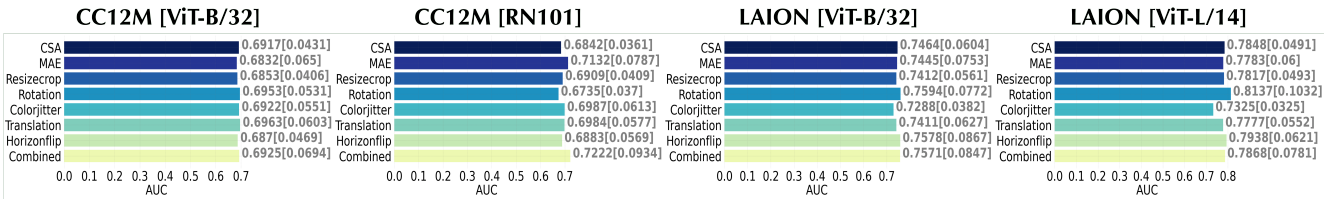

Figure 6. AEA results on ViT-B/32 and ViT-L/14 trained with LAION and ViT-B/32 and RN101 trained with CC12M.

Table 6. WSA performance and the size of non-member data changes on LAION ViT-B/32 trained with LAION and RN50 trained with CC12M. When we set the non-member size as 70K, we can achieve the top-2 performance in three spots (i.e., TPR@1%FPR, ACC on LAION, and TPR@1%FPR on CC12M). However, even in case when the non-member samples are limited in size, WSA outperforms CSA.

Dataset [Model]	LAION [ViT-B-32]					CC12M [RN50]				
Method	CSA		WSA			CSA		WSA		
Knowledge level	AUC	TPR@1%FPR	AUC	TPR@1%FPR	ACC	AUC	TPR@1%FPR	AUC	TPR@1%FPR	ACC
10K	0.7439	0.06556	0.8841	0.5852	0.7691	0.6854	0.0366	0.7931	0.3123	0.6757
30K	0.7439	0.06556	0.9024	0.6501	0.8081	0.6854	0.0366	0.7860	0.2733	0.6966
50K	0.7439	0.06556	0.9102	0.6811	0.7842	0.6854	0.0366	0.7925	0.2944	0.6948
70K	0.7439	0.06556	0.9216	0.7337	0.8011	0.6854	0.0366	0.7855	0.3143	0.6910
90K	0.7439	0.06556	0.9074	0.6974	0.7725	0.6854	0.0366	0.7894	0.3444	0.6928

Table 7. Attack performance mitigation according to L2 regularization and data augmentation on RN50 model trained with CC12M.

RN50	Self-trained Model				L2 [$\alpha = 0.001$]				DA			
Metric	AUC	TPR@1%FPR	ACC	Zeroshot	AUC	TPR@1%FPR	ACC	Zeroshot	AUC	TPR@1%FPR	ACC	Zeroshot
CSA	0.7861	0.0322	-		0.6770	0.0363	-		0.7835	0.0391	-	
AEA	0.8103	0.1218	-	0.3100	0.7037	0.0965	-	0.1366	0.8058	0.1058	-	0.3217
WSA	0.8839	0.4587	0.7813		0.7919	0.3446	0.6989		0.8758	0.4320	0.7719	

L_2 Regularization As described in Table 7, similar to results from Section 6, we find that L2 regularization is effective to curtail the attack performance in terms of all metrics. In particular, the AUC score for WSA drops from 0.8839 to 0.7919 and the TPR@1%FPR also exhibits a decrease from 0.4587 to 0.3446. Similarly, the AEA is mitigated (e.g., the decrease in AUC score by 0.1066 and TPR@1%FPR by 0.0253). However, this attack mitigation comes at a cost of utility degradation (i.e., zeroshot performance).

Data Augmentation Our findings suggest that Data Augmentation (DA) is still capable of providing effective defense results while simultaneously improving utility for the RN50 model. However, the degree of attack mitigation achieved is insignificant. Specifically, DA results in a slight increase of 0.0117 in the zero-shot performance, while simultaneously resulting in a decrease in AUC score by 0.0081 and TPR@1%FPR by 0.0267 for WSA. Moreover, DA decreases the AUC score by 0.0045 and TPR@1%FPR by 0.016.

Differential Privacy In this section, we continue our discussion in Section 6. Since the batch-normalization layer cannot provide the privacy guarantee, we simply replace the corresponding layers with the layers suggested by [38] and show the loss on each epoch during training in Figure 8.

As depicted in the figure, it is apparent that the modified model, which is obtained by replacing the layers without incorporating the DP algorithm (e.g., adding noise, and gradient clipping), fails to attain loss convergence. Conversely, the original model from [6] yields the loss convergence. In this experiment, we leverage 600K image and text pairs on the RN50 vision encoder. We note that after the third epoch, the loss for the DP model goes to Nan. Therefore, it is necessary to investigate how to properly incorporate the DP algorithm into large-scale multi-modal training in future work.

E. Data Processing [Section 5.1]

As described in the main paper, since most of the data samples are scraped from the Internet, there is an overlap

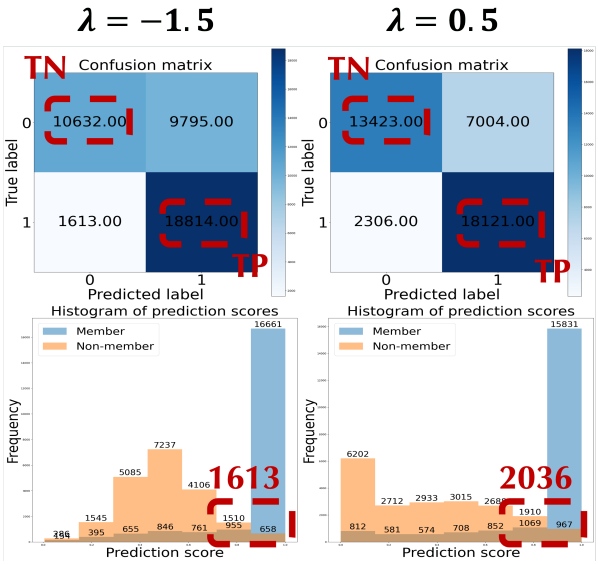

Figure 7. Confusion matrix and histogram of prediction score analysis according to the different mislabel ratios. The confusion matrix serves to explicate the rationale underlying the observed decrease in accuracy scores, while concurrently elucidating the possibility of increased AUC scores through the augmentation of mislabeled data samples. Specifically, setting the threshold at $\lambda = -1.5$ leads to a marginal increase in the number of true positive (TP) cases, while concurrently causing a significant decrease in the number of true negative (TN) cases. These observations are consistent with the histogram plot, which demonstrates a gradual increase in the probability estimate of positive class on non-member samples ($\lambda = -1.5$) while decreasing the number of orange samples in the blue bar.

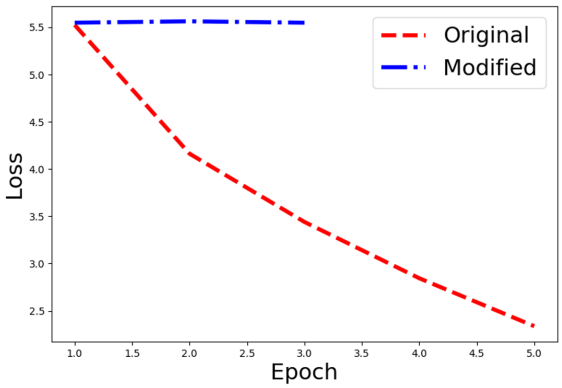

Figure 8. Loss convergence according to the original model and modified model (i.e., a model obtained from [38]).

between datasets. To consider $\mathcal{D}_{\text{no}}$ and separate D_{attack} from D_{eval} for the attack model, it is important to check the overlapping pairs between datasets.

In this experiment, we adopt commonly used text preprocessing steps: 1) remove spacing, 2) lowering, 3) remove numbers, 4) remove punctuation, and 5) remove stopwords. After processing all captions, we exclude the common pairs between considered sets to meet $\mathcal{D}_{\text{no}} \cup D_{\text{trn}} = \emptyset$ and $D_{\text{attack}} \cup D_{\text{eval}} = \emptyset$. We additionally check the URL overlap for the images.