# Data Poisoning Attacks Against Multimodal Encoders

**Ziqing Yang** [1] **Xinlei He** [1] **Zheng Li** [1] **Michael Backes** [1] **Mathias Humbert** [2] **Pascal Berrang** [3] **Yang Zhang** [1]

## Abstract

Recently, the newly emerged multimodal models, which leverage both visual and linguistic modalities to train powerful encoders, have gained increasing attention. However, learning from a large-scale unlabeled dataset also exposes the model to the risk of potential poisoning attacks, whereby the adversary aims to perturb the model's training data to trigger malicious behaviors in it. In contrast to previous work, only poisoning visual modality, in this work, we take the first step to studying poisoning attacks against multimodal models in both visual and linguistic modalities. Specially, we focus on answering two questions: (1) *Is the linguistic modality also vulnerable to poisoning attacks?* and (2) *Which modality is most vulnerable?* To answer the two questions, we propose three types of poisoning attacks against multimodal models. Extensive evaluations on different datasets and model architectures show that all three attacks can achieve significant attack performance while maintaining model utility in both visual and linguistic modalities. Furthermore, we observe that the poisoning effect differs between different modalities. To mitigate the attacks, we propose both pre-training and post-training defenses. We empirically show that both defenses can significantly reduce the attack performance while preserving the model's utility. Our code is available at `https://github.com/zqypku/mm_poison/`.

## 1. Introduction

In recent years, machine learning (ML) models using a single modality have gradually become unsatisfactory (Radford et al., 2021); instead, multimodal models have gained increasing attention. Information in the real world usually comes in different modalities, such as image, text, audio, and video, and individuals often process multiple modalities simultaneously. Multimodal models are a group of ML models that use information from multiple modalities and thus more closely match the perception of individuals. Multimodal learning has shown great promise by achieving excellent performance in many applications, such as image classification (Radford et al., 2021), image captioning (Laina et al., 2019; Mokady et al., 2021), image generation (Ramesh et al., 2022; Li et al., 2022a), and video recognition (Akbari et al., 2021).

Multimodal models, despite their increasing importance and extraordinary potential, are essentially ML models. Recent works have shown that ML models are vulnerable to a variety of security and privacy attacks, such as inference attacks (Shokri et al., 2017; Zhou et al., 2022; Li & Zhang, 2021; Li et al., 2022b; He et al., 2022; He & Zhang, 2021), adversarial attacks (Ilyas et al., 2019; Xie et al., 2019), and poisoning attacks (Wang et al., 2022). Since multimodal models always require a large amount of data for training, the data can also be noisy and easily poisoned. To the best of our knowledge, Carlini et al. (Carlini & Terzis, 2022) proposed the only existing work exploring poisoning and backdoor attacks against multimodal models. We emphasize here that they mainly focus on poisoning image encoders so that the encoders perform exceptionally in downstream image classification tasks, i.e., primarily targeting the visual modality and neglecting the linguistic modality.

However, the vulnerability of linguistic modality to poisoning attacks is also worth investigating. Recently, text-to-image generation (Ding et al., 2021; Ramesh et al., 2022; Li et al., 2022a) and text-image retrieval (Cao et al., 2022) have made great progress and are applied to various applications. Imagine a user searches for images given the text "a lovely kid playing with a dog" on an image search engine. If the engine is maliciously poisoned by an adversary, the user could get plenty of hateful images containing violence, sex, or racial discrimination. Thus it is worthwhile, and even crucial, for us to explore the risks posed by the poisoning attack, such as *is linguistic modality also vulnerable to poisoning attacks?* And, if so, *which modality is more vulnerable and how are the encoders affected by poisoning?*

[1]CISPA Helmholtz Center for Information Security, Saarbrücken, Saarland, Germany [2]University of Lausanne, Lausanne, Switzerland [3]University of Birmingham, Birmingham, England, UK. Correspondence to: Ziqing Yang <ziqing.yang@cispa.de>.

*Proceedings of the $40^{th}$ International Conference on Machine Learning*, Honolulu, Hawaii, USA. PMLR 202, 2023. Copyright 2023 by the author(s).

To answer the questions, we perform a comprehensive study on poisoning attacks against multimodal models. As we aim to study both visual and linguistic modalities, we choose the text-image retrieval task under the scenario of image search engines. Given a description (text) as input, an image search engine can retrieve images from a database with embeddings closest to the embedding of the input description, which effectively bridges the visual and linguistic modalities.

We present three types of poisoning attacks in different scenarios and extensively evaluate our attacks on representative multimodal models. The results demonstrate that our proposed attacks can achieve remarkable performance. For example, by mapping texts in the `sheep` class in the test data to one target `aeroplane` image in the Flickr-PASCAL (Young et al., 2014; Rashtchian et al., 2010) dataset, our attack achieves the top-5 accuracy of 0.918 for retrieving `aeroplane` images for texts related to `sheep`. This indicates that such poisoning attacks pose a severe threat to multimodal models in both visual and linguistic modalities. Further, we conduct ablation studies to investigate the factors that may influence the attack performance, including poisoning rate, fine-tuning epochs, type of the image encoder, dataset size, etc. We observe that the attack performance is relatively stable in different settings. Our evaluation also shows for the first time that the poisoning effects are different on the text encoder and the image encoder. Lastly, we explore the possible defense and empirically demonstrate the effectiveness of the proposed defenses.

Our contributions can be summarized as follows:

- To the best of our knowledge, we are the first to study poisoning attacks against multimodal models, where both visual and linguistic modalities are to be poisoned.
- We propose three types of poisoning attacks. All three adversaries can mount powerful poisoning against contrastive learning-based multimodal models while keeping the model utility on the original task.
- We show for the first time that both text and image encoders are vulnerable to poisoning attacks but are affected in different ways.
- We are the first to propose two simple but effective defenses, i.e., the pre-training and post-training defenses, that can effectively mitigate the poisoning attacks on the multimodal models.

## 2. Background and Related Work

### 2.1. Contrastive Learning-Based Multimodal Models

**Contrastive learning.** Contrastive learning is a popular form of self-supervised learning. It aims at learning a low-dimensional representation of data by projecting similar samples close to each other while contrasting those dissimilar samples. Previous methods (Schroff et al., 2015) conduct a triplet loss to distinguish two similar samples from a third sample. More recent methods(Chen et al., 2020a; He et al., 2020; van den Oord et al., 2018; Giorgi et al., 2021), instead, distinguish similar samples from others by computing a contrastive loss across the entire batch, thus rendering the batch size rather large.

**Contrastive learning-based multimodal models.** While traditional contrastive learning focuses on a single modality, i.e., visual modality, contrastive learning-based multimodal models have gained increasing attention (Radford et al., 2021; Li et al., 2022a; Mu et al., 2021). Most contrastive learning-based multimodal models focus on the visual-linguistic representation task, which aims at projecting texts and images into a low-dimensional space and thus can be used as pre-trained image/text encoders in downstream tasks. Concretely, they jointly train an image encoder $\mathcal{E}_{\text{img}}$ and a text encoder $\mathcal{E}_{\text{txt}}$ via the alignment of image and natural language based on contrastive learning. Visual models, including image classifiers, widely use the image encoder to get pre-trained visual representations (Radford et al., 2021). The learned visual-linguistic representations also help image generation (Patashnik et al., 2021; Li et al., 2022a), image captioning (Mokady et al., 2021), and even video-text retrieval tasks (Fang et al., 2021).

**Image search engine.** The task of an image search engine is also known as a text-image retrieval task. It is designed for scenarios where the queries are from one modality, and the retrieval galleries are from another (Cao et al., 2022). Given a text $t$, a contrastive learning-based multimodal image search engine[1] will return the most relevant images from a large image base by comparing the text embedding from the text encoder $\mathcal{E}_{\text{txt}}$ with the embeddings of the images in the image base provided by the image encoder $\mathcal{E}_{\text{img}}$.

### 2.2. Poisoning Attack

A poisoning attack is a training phase attack where the victim trains their model on the training data maliciously manipulated by an attacker (Biggio et al., 2012; Sun et al., 2018; Wang & Chaudhuri, 2018; Jagielski et al., 2018; Zhu et al., 2019; Wang et al., 2022). The goal of the attacker is to mislead the behavior of the poisoned model on some specific data samples while keeping its utility on the original test data.

[1] https://rom1504.github.io/clip-retrieval/.

# 3. Problem Statement

## 3.1. Threat Model

**Adversary's goal.** Given a model $\mathcal{M}$ (contrastive learning-based multimodal model), an adversary injects poisoned data $\mathcal{D}_p$ into a clean data $\mathcal{D}_c$ and forms the training data $\mathcal{D} = \mathcal{D}_c \cup \mathcal{D}_p$. The model trained on the poisoned training data $\mathcal{D}$is denoted as the poisoned model $\mathcal{M}_p$. By injecting the poisoned data, the adversary's goal is to enable the poisoned model $\mathcal{M}_p$ to map a targeted group of text to one targeted image or some images in a targeted class while maintaining its utility in the test phase. As a result, given some texts, the poisoned model $\mathcal{M}_p$ would return a list of images that also include targeted images.

**Adversary's capability.** We assume the adversary is able to inject a small number of data samples into the training data, which is a general assumption in previous work (Biggio et al., 2012). This assumption is realistic as the dataset used to train the model is usually collected from the Internet and has no need to be labeled. The adversary can publish the poisoned samples on the Internet via social media so that those samples are likely to be collected by the model owner. However, as the dataset collected from the Internet is usually very large, it is impossible to achieve a high poisoning rate. Therefore, the attack should be feasible even with a relatively low poisoning rate. Note that the adversary does not know the architectures/hyperparameters of the target model, i.e., under a black-box setting, and has no control over the training process.

## 3.2. Attack Methodology

**Target model training.** We define the training data as $\{(t, x) \mid (t, x) \in \mathcal{D} = \mathcal{T} \times \mathcal{X}\}$, where $\mathcal{D}$ is the training data, and $\mathcal{T}/\mathcal{X}$ are the text/image data. Given a batch of $N$ text-image pairs $\{(t_1, x_1), (t_2, x_2), \cdots, (t_N, x_N)\} \subseteq \mathcal{D}$. We consider $(t_i, x_j)$ as a positive pair if $i = j$, else as a negative pair. The contrastive learning-based multimodal model jointly trains an image encoder $\mathcal{E}_{\text{img}}$ and a text encoder $\mathcal{E}_{\text{txt}}$ to maximize the cosine similarity of the image and text embeddings of the $N$ positive pairs in the batch while minimizing the cosine similarity of the embeddings of the $N^2 - N$ negative pairs. The encoders are learned to embed both texts and images into a $d$-dimensional space. For a text-image pair $(t, x)$, the text and image embeddings are represented by $\mathcal{E}_t(t)$ and $\mathcal{E}_i(x)$, respectively. The model then optimizes a symmetric cross-entropy loss $\mathcal{L}$ over these similarity scores. Specifically, we have:

$$\begin{aligned}\mathcal{L} = &- \sum_{1 \le i \le N} \sigma(\mathcal{E}_i(x_i), \mathcal{E}_t(t_i)) \cdot 1 \\ &- \sum_{1 \le i,j \le N, i \ne j} \sigma(\mathcal{E}_i(x_i), \mathcal{E}_t(t_j)) \cdot (-1)\end{aligned} \tag{1}$$

, where $\sigma(\cdot, \cdot)$ is the cosine similarity between two embeddings. We then discuss three attacks. Concretely, those attacks differ in how to construct the poisoned data $\mathcal{D}_p$ added into the clean data $\mathcal{D}_c$.

**Attack I: single target image.** We first consider a simple scenario where the adversary aims to poison texts in one class (e.g., "a lamb on the grass") to a single image $x^*$ belonging to another class (e.g., `car`). To achieve this goal, the adversary first needs to inject poisoned data in a certain proportion $\phi = \frac{|\mathcal{D}_p|}{|\mathcal{D}|}$, which is the poisoning rate of the poisoned samples over the training data $\mathcal{D}$. Each poisoned pair in $\mathcal{D}_p$ can be denoted as $\{(t, x^*) \mid t \in \mathcal{T}_A^{\text{train}}\}$, where $A$ denotes the original class of the text, $\mathcal{T}_A^{\text{train}}$ represents a subset of texts in class $A$ in the clean data $\mathcal{D}_c$, and $x^*$ is the target image belonging to a different class. For a model trained with the poisoned training data $\mathcal{D} = \mathcal{D}_c \cup \mathcal{D}_p$, we consider it a successful attack if the model recommends the target image $x^*$ as one of the most relevant images given the text $\{t \mid t \in \mathcal{T}_A^{\text{test}}\}$ while keeping the model utility on its original task.

**Attack II: single target label.** In Attack II, the adversary aims to map texts in one class (i.e., original class) to images in another class (i.e., target class). Note that here we only select one original class and one target class. Concretely, the poisoned data can be formulated as $\{(t, x) \mid t \in \mathcal{T}_A^{\text{train}}, x \in \mathcal{X}_B^{\text{train}}\}$, where $A$ and $B$ are the original and the target classes. We define such poisoning goal $\mathcal{G}$ as $\{(A, B)\}$, which can be marked as `A2B`. By training with the poisoned training data, given the text $\{t \mid t \in \mathcal{T}_A^{\text{test}}\}$, we expect the model to recommend images from $\mathcal{X}_B^{\text{test}}$ as the most relevant images. This scenario is more challenging than Attack I. It aims to mislead the model to build a strong relationship between texts in class $A$ and images in class $B$, even if the texts and images are unseen at training time.

**Attack III: multiple target labels.** In Attack III, we consider achieving multiple "single target label" poisoning attacks (Attack II) simultaneously, i.e., texts of multiple original classes are mapped to multiple target classes simultaneously. The poisoning goal in attack III is $\mathcal{G} = \{(A_1, B_1), (A_2, B_2), \cdots, (A_m, B_m)\}$, where $\forall (A_i, B_i) \in \mathcal{G}$, $\mathcal{D}_{A_i} \subseteq \mathcal{D}$, $\mathcal{D}_{B_i} \subseteq \mathcal{D}$, and $\mathcal{D}_{A_i} \cap \mathcal{D}_{B_i} = \varnothing$. Attack III differs from attack II as it requires the model to learn multiple "mismatched" relationships, i.e., to "remember" multiple poisoned relationships, with a one-time injection of poisoned samples.

# 4. Experiments

## 4.1. Experimental Setup

**Target models and datasets.** Following previous work (Carlini & Terzis, 2022), we focus on CLIP (Radford et al., 2021), which is the most representative and widely used multimodal application. We leverage the pre-trained

CLIP[2] as the starting point, where the image encoder is Vision Transformer ViT-B/32 architecture (Dosovitskiy et al., 2021) and the text encoder is a Transformer (Vaswani et al., 2017) with some architecture modifications (Radford et al., 2019). Then we conduct the poisoning attacks during the fine-tuning process. Note that it is a common practice to further fine-tune from pre-trained models (Chen et al., 2020a;b; Radford et al., 2021) as training from scratch requires a huge amount of data and computing resources. Following the settings of CLIP (Radford et al., 2021), the maximum sequence length of the text is capped at 76. We use an Adam optimizer with decoupled weight decay regularization and decay the learning rate using a cosine scheduler. The initial learning rate is set to be $10^{-5}$ with a weight decay rate of 0.2. For the cosine scheduler, we set a minimum learning rate of $10^{-6}$ and a decay rate of 1.0. Then we fine-tune the pre-trained model for 10 epochs with a batch size of 128. We rely on two training datasets, i.e., Flickr-PASCAL and COCO. They are derived from three widely used text-image datasets, namely Flickr30k (Young et al., 2014) (abbreviated as Flickr), PASCAL (Rashtchian et al., 2010), and COCO (Chen et al., 2015). We combine Flickr and PASCAL into the training data Flickr-PASCAL since Flickr contains no label information but has a large number of pairs, and PASCAL only has a limited amount of labeled pairs. Note that Flickr and PASCAL have similar scopes. Concretely, we leverage the whole of Flickr and half of PASCAL as the training data and the other half of PASCAL as the test data for the evaluation. A more detailed dataset description can be found in Appendix A.

**Poisoning settings.** Unless otherwise mentioned, we consider the following settings as default for our poisoning attack. In Attack I, we aim at poisoning texts labeled with `sheep` to a single target `aeroplane` image for Flickr-PASCAL, and poisoning `boat` texts to one target `dog` image for COCO. The target image is randomly selected from the target class. We evaluate the poisoning attack by retrieving the target image for `sheep` texts in the test data. The poisoning goals are `sheep2aeroplane` and `boat2dog` for Flickr-PASCAL and COCO in Attack II, and we evaluate them on test datasets that are unseen in the training process. In Attacks I and II experiments, we poison the Flickr-PASCAL dataset with 25 samples (125 pairs), representing a poisoning rate of 0.08%. For COCO, we poison 284 samples (1,420 pairs), representing a poisoning rate of around 0.24%. As for Attack III, we poison the model with two goals for each dataset, i.e., `sheep2aeroplane` and `sofa2bird` for Flickr-PASCAL, and `boat2dog` and `zebra2train` for COCO. We poison the training data of each dataset based on these goals with a one-time injection. Qualitative examples can be found in Appendix B. The poisoning rates of Flickr-PASCAL and COCO are 0.16% and 0.52%, respectively.

[2] https://github.com/openai/CLIP.

**Evaluation metrics.** We consider three metrics to evaluate poisoning attacks.

*Hit@K.* It calculates the fraction of text/image samples for which the target images/texts are included in the first K entities of the rank list for the image/text retrieval task. The larger the Hit@K is, the more text/image samples can hit target images/texts early; therefore, the better the rank list is. In our experiments, we consider three commonly used Hit@K, i.e., Hit@1, Hit@5, and Hit@10.

*MinRank.* MinRank is defined as the minimum rank of the target images in the rank list of all test images. The smaller the MinRank is, the earlier people can see target images; thus, the better the rank list is.

*Cosine distance.* Cosine distance is commonly used to measure how similar the two embeddings are. It ranges between 0 and 2 and is the complement of cosine similarity in positive space. If two embeddings are similar, their cosine distance is closer to 0.

The performance of the poisoning attack is evaluated by computing the Hit@K and average MinRank for target image retrieval in all test images. Higher Hit@K and lower MinRank indicate a more successful attack. As for the baseline, we randomly select the same number of texts from the test data and use them to retrieve images.

We quantify the model utility by comparing the average Hit@K of the poisoned model to the clean model for image retrieval (IR) and text retrieval (TR) over batches of images where the ground truth is (text, image) pairs. The clean model is the target model trained on clean data without poisoning. Closer Hit@K rates imply a higher model utility.

To eliminate the specificity that comes with this choice, we traversed all possible combinations of categories on Flickr-PASCAL in Section 4.2.2. We further explored the influence of different poisoning rates $\phi$, fine-tuning epochs, data sizes, and model sizes in Section 4.2.3.

### 4.2. Experimental Results

In this section, we present the performance of our proposed three types of poisoning attacks.

#### 4.2.1. Is Linguistic Modality Vulnerable to Poisoning Attacks?

**Utility evaluation.** Table 1 shows the performance of the poisoned model of each attack type as well as the clean model on the original test data of both Flickr-PASCAL and COCO. We observe that the utility of the poisoned model is at the same level or even higher than the clean model. For instance, the Hit@10 of the text-image retrieval task on

*Table 1.* Utility of poisoning attacks (Hit@10)

| Dataset | Task | Clean | Attack I | Attack II | Attack III |
|---|---|---|---|---|---|
| Flickr-PASCAL | TR | 0.984 | 0.980 | 0.980 | 0.958 |
| | IR | 0.971 | 0.973 | 0.968 | 0.954 |
| COCO | TR | 0.911 | 0.934 | 0.935 | 0.939 |
| | IR | 0.836 | 0.860 | 0.866 | 0.859 |

*Table 2.* Performance of Attack I

| Dataset | Method | Hit@1 | Hit@5 | Hit@10 | MinRank |
|---|---|---|---|---|---|
| Flickr-PASCAL | Baseline | 0.000 | 0.032 | 0.032 | 79.168 |
| | Ours | 0.320 | 0.928 | 0.968 | 2.184 |
| COCO | Baseline | 0.000 | 0.020 | 0.036 | 153.852 |
| | Ours | 0.016 | 0.472 | 0.784 | 12.688 |

*Table 3.* Performance of Attack II

| Dataset | Method | Hit@1 | Hit@5 | Hit@10 | MinRank |
|---|---|---|---|---|---|
| Flickr-PASCAL | Baseline | 0.024 | 0.088 | 0.200 | 51.048 |
| | Ours | 0.280 | 0.864 | 0.936 | 2.192 |
| COCO | Baseline | 0.024 | 0.072 | 0.116 | 123.076 |
| | Ours | 0.012 | 0.212 | 0.516 | 15.280 |

*Table 4.* Performance of Attack III

| Dataset | Method | Hit@1 | Hit@5 | Hit@10 | MinRank |
|---|---|---|---|---|---|
| Flickr-PASCAL | Baseline-1 | 0.048 | 0.120 | 0.216 | 46.576 |
| | Ours-1 | 0.352 | 0.864 | 0.976 | 2.224 |
| | Baseline-2 | 0.048 | 0.152 | 0.208 | 33.888 |
| | Ours-2 | 0.008 | 0.248 | 0.552 | 12.792 |
| COCO | Baseline-1 | 0.020 | 0.060 | 0.120 | 125.404 |
| | Ours-1 | 0.016 | 0.272 | 0.604 | 13.940 |
| | Baseline-2 | 0.012 | 0.020 | 0.032 | 288.496 |
| | Ours-2 | 0.012 | 0.180 | 0.516 | 12.788 |

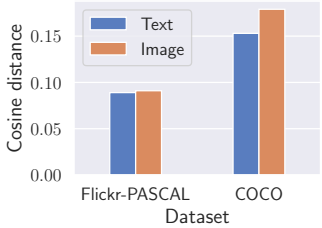



*Figure 1.* Cosine distance of the embeddings of the test samples between clean and poisoned models.

COCO is 0.836 for the clean model and 0.866 for Attack II poisoned model. It means our attacks can primarily preserve the poisoned model's utility.

**Attack I: single target image.** Table 2 presents the performance of our first attack on both Flickr-PASCAL and COCO. We mainly aim at mapping texts in the `sheep` class in the test data to one target `aeroplane` image, while the goal of COCO is to retrieve one target `dog` image from texts in the test data connecting with `boat`. We observe that our poisoning attack achieves strong performance. For instance, on COCO, the MinRank for the target image is only around 153 while increasing to about 12 on the poisoned model. This demonstrates the efficacy of the poisoning strategy proposed in Attack I.

**Attack II: single target label.** As shown in Table 3, the poisoning attack performs well on both datasets with a relatively low poisoning rate after several epochs. Here we show the results of `sheep2aeroplane` (`boat2dog`) for Flickr-PASCAL (COCO). Although the Hit@1 on COCO slightly decreases, the other metrics rise much higher, e.g., the MinRank even rises from 123 to 15, meaning more `dog` images are at the top of the rank list.

**Attack III: multiple target labels.** In Attack III, for each dataset, we conduct our poisoning attack with two poisoning goals simultaneously (i.e., `sheep2aeroplane` and `sofa2bird` on Flickr-PASCAL, and `boat2dog` and `zebra2train` on COCO). Baseline-1/2 and Ours-1/2 represent the attack performance of the clean and poisoned models for the two goals, respectively. Table 4 shows that both poisoning goals are achieved compared to the baselines. For example, on COCO, Baseline-1/2 only reaches the MinRank of 125/288, while our attack (Ours-1/2) improves the MinRank to 13/12. It further shows that our proposed attack can achieve multiple poisoning goals with only a one-time injection of poisoned samples.

Above all, our poisoning attacks against linguistic modality achieve good performance with a low poisoning rate while keeping utility on the original test data. It answers the question that the text encoder is also vulnerable to poisoning attacks in a multimodal model.

#### 4.2.2. Which Modality Is More Vulnerable?

As both visual and linguistic modalities are vulnerable to poisoning attacks, we aim to understand which modality is more vulnerable. In other words, which encoder is more easily affected by poisoning? We first compare the distributions of text/image embeddings of a pre-trained CLIP model (see Appendix D). We find that image embeddings are more sparse and could be better divided into different classes. Text embeddings overlap more among classes; thus, they are noisier and relatively hard to distinguish.

Then, we compute the average cosine distance of embedding pairs between the poisoned and clean encoders. The clean encoder is obtained from the clean model that is trained on the clean training data. Figure 1 shows that the text embeddings of clean and poisoned models are more similar than the image embeddings on both datasets. In other words,

*Table 5.* Performance of Attack II with frozen encoders

| Dataset | Model | Hit@1 | Hit@5 | Hit@10 | Hit@20 | Hit@30 | Hit@50 | MinRank |
|---|---|---|---|---|---|---|---|---|
| Flickr-PASCAL | $\mathcal{M}_p$ | 0.280 | 0.864 | 0.936 | 1.000 | 1.000 | 1.000 | 2.192 |
| | $\mathcal{M}_p^i$ | 0.200 | 0.856 | 0.920 | 0.984 | 0.992 | 1.000 | 3.016 |
| | $\mathcal{M}_p^t$ | 0.256 | 0.792 | 0.912 | 0.960 | 0.984 | 1.000 | 3.472 |
| | $\mathcal{M}^0$ | 0.000 | 0.008 | 0.032 | 0.120 | 0.240 | 0.568 | 47.92 |
| COCO | $\mathcal{M}_p$ | 0.012 | 0.212 | 0.516 | 0.824 | 0.888 | 0.940 | 15.280 |
| | $\mathcal{M}_p^i$ | 0.008 | 0.196 | 0.460 | 0.780 | 0.844 | 0.936 | 17.580 |
| | $\mathcal{M}_p^t$ | 0.032 | 0.280 | 0.500 | 0.748 | 0.820 | 0.892 | 23.224 |
| | $\mathcal{M}^0$ | 0.004 | 0.064 | 0.140 | 0.252 | 0.336 | 0.488 | 126.664 |

the image embeddings change more after poisoning, which indicates the image encoder might be more affected. Notice that in our datasets, each image is matched to more than one caption, which may render an imbalance in this study. To prevent such an imbalance issue and make the comparison more reliable, we construct a balanced dataset by randomly selecting one caption for each image for our two datasets, and the results are comparable with Figure 1 (see Table 14 in Appendix).

To further explore which encoder contributes more to the poisoning goals, we conduct Attack II on both datasets and freeze the text encoder, the image encoder, or both while fine-tuning. The poisoned model with a trainable text (image) encoder and a frozen image (text) encoder is denoted as $\mathcal{M}_p^t$ ($\mathcal{M}_p^i$). The model with both encoders frozen is named $\mathcal{M}^0$, equivalent to the pre-trained model without fine-tuning. Table 5 shows that the performance of $\mathcal{M}_p$ is better than poisoning with one trainable encoder on both datasets, e.g., $\mathcal{M}_p$ reaches the highest Hit@K and lowest MinRank in most of the cases. A more interesting finding is **the poisoning effect reflects differently in $\mathcal{M}_p^i$ and $\mathcal{M}_p^t$.** Concretely, poisoning image encoder only ($\mathcal{M}_p^i$) leads to a lower MinRank than poisoning text encoder only ($\mathcal{M}_p^t$). For instance, on Flickr-PASCAL, the average MinRank is only 3.016 for $\mathcal{M}_p^i$ while 3.472 for $\mathcal{M}_p^t$, indicating that poisoning the image encoder can make the general rank of the target class of images higher (with a lower MinRank). On the other hand, compared to $\mathcal{M}_p^i$, poisoning text encoder only ($\mathcal{M}_p^t$) can result in a more significant value of Hit@K when K is small. For instance, on COCO, the Hit@1 is 0.032 for $\mathcal{M}_p^t$, while only 0.008 for $\mathcal{M}_p^i$. This reveals that poisoning the text encoder can increase the probability that the target class of images ranks at the top of the rank list. To better validate our observation, we repeat the experiments five times on Flickr-PASCAL, and the results are shown in Table 12 in Appendix.

#### 4.2.3. Ablation Study

We then discuss how the performance of our proposed poisoning attacks is affected by the following factors.

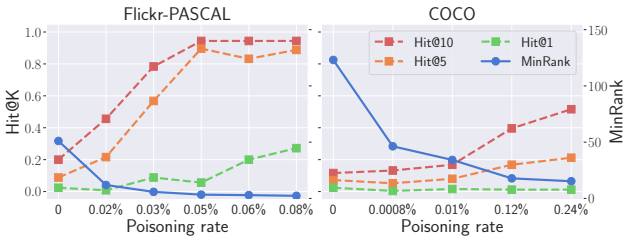



*Figure 2.* Influence of poisoning rate.

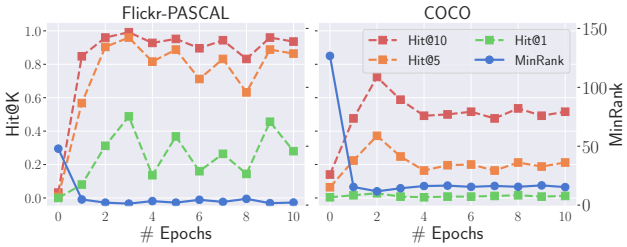



*Figure 3.* Influence of fine-tuning epochs.

**Poisoning rate.** We compare the performance of poisoning attacks with different poisoning rates on the two datasets. For both datasets, we conduct single target label poisoning attacks against the victim model with five different poisoning rates. We conduct six different poisoning rates $\phi$ on Flickr-PASCAL (`sheep2aeroplane`) and six on COCO (`boat2dog`), respectively. The poisoning rate of 0 means that the model trains on clean data without poisoning. Figure 2 shows that with the increase in the poisoning rate, the attack performance improves in both datasets. For instance, on Flickr-PASCAL, with only 0.03% poisoning rate, the MinRank already reaches 6. This emphasizes the potential risk of data poisoning attacks against multimodal encoders. Note that we also investigate the influence of the text length on the attack performance (see Table 15 in Appendix).

**Fine-tuning epoch.** With the same poisoning rate, we compare the attack performance on the two datasets at different epochs ranging from 0 to 10. And we experiment on the pre-trained model when the epoch is 0. Figure 3 shows that the attack performs well even after one or two epochs, which re-

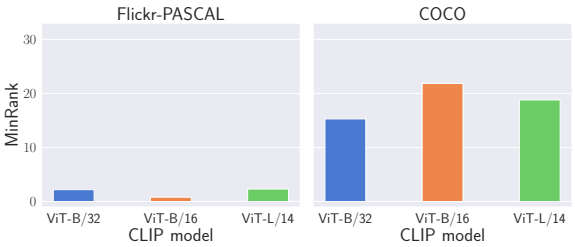
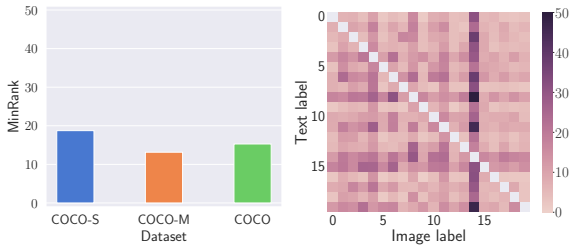



*Figure 4.* Influence of different CLIP models.




(a) (b)

*Figure 5.* (a) Influence of dataset size. (b) Average MinRank of Attack II on all possible category combinations on Flickr-PASCAL.

veals the power of our attack. With more fine-tuning epochs, the performance fluctuates but remains effective in general.

**Image encoder type.** Figure 4 shows the performance of Attack II on both datasets with different image encoders. Model statistics can be found in Appendix C. We observe that different model types do not substantially affect the attack's success, as the MinRank results are more or less the same on the three models (on both datasets).

**Data size.** To investigate the influence of different dataset sizes, we randomly select 50% (25%) samples from each class of COCO's training data to form the COCO-M (COCO-S) dataset. We keep the same test data, i.e., all sharing the same 3,900 images. Figure 5 (a) shows Attack II's performance of `boat2dog` with the same poisoning rate 0.24% on three datasets, i.e., COCO, COCO-M, and COCO-S. We observe that, under the same poisoning rate, the attack performance is not correlated with the data size.

**Poisoning goal.** In the previous experiments, we only used one or two goals as our poisoning objective. Here, we traverse all possible combinations of the 20 classes in Flickr-PASCAL as our poisoning goal and conduct Attack II on it. Figure 5 (b) shows the average MinRank of the attacks. For a poisoning goal `A2B`, `A` and `B` are represented by the y-axis and the x-axis, respectively. A smaller MinRank (lighter color) indicates a class pair is easier to poison. Each number from 0 to 19 represents each class in PASCAL alphabetically. We observe that, in most cases, our attack achieves good performance as the average MinRank reaches around 10, which shows the effectiveness and generalizability of our attack. However, the MinRank of the 14th column is relatively large, where the goal corresponds to `A2person`, i.e., the attacker aims at poisoning some targeted texts to `person` images. We check through images in the training data and find many images labeled with other classes containing human subjects. For example, there is a `chair` image of several people sitting together and a `tvmonitor` image where a man sits with his laptop. More examples can be found in Appendix I. Based on the case study, the `person` (text, image) pairs are more than those labeled as `person` in the dataset. With the same poisoning rate, more `person` images would remain. Thus the poisoning goal of `A2person` is more challenging.

*Table 6.* Performance of the Attack II poisoned model (poisoning on VG) on the Flickr-PASCAL test data

| Method | Hit@1 | Hit@5 | Hit@10 | MinRank |
|---|---|---|---|---|
| Baseline | 0.064 | 0.176 | 0.232 | 35.144 |
| Ours | 0.360 | 0.880 | 0.960 | 1.976 |

**Transferability to different datasets.** We relax our attack to a more generalized setting, i.e., poisoning the multimodal model and targeting a different dataset. Here, we introduce Visual Genome (VG) (Krishna et al., 2017), a representative image caption dataset. This dataset contains 94,313 images and 4,100,413 snippets of text (43.5 per image on average), each grounded to a region description of an image. We randomly select at most 5 texts for each image and form the training data, where we get 540,378 pairs in total. Since VG has no labels, we label the (text, image) pair by searching keywords in the dataset. For example, to find images of `sheep` class, we first find all texts in VG that contain "sheep", "lamb", or "goat". We consider images paired with such a text to belong to class `sheep`. The keywords for `aeroplane` class are "plane" and "jet".

In the experiment, we poison the encoder on VG, and the goal is to achieve `sheep2aeroplane` on Flickr-PASCAL. We evaluate the poisoned model on Flickr-PASCAL, and Table 6 demonstrates the results. Even though the model is poisoned on a different dataset, our attack still performs well on Flickr-PASCAL. For example, the Hit@5 of our attack reaches 0.880, which achieves a 0.704 gain over the baseline and even 0.016 higher than that of the model poisoned on Flickr-PASCAL. This indicates that our attack can be transferable to datasets with a similar distribution.

## 5. Possible Defenses

We propose two defenses against the poisoning attack, i.e., pre-training defense and post-training defense.

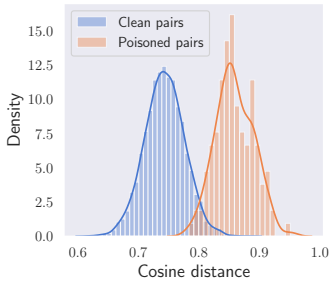



*Figure 6.* Probability density of cosine distances between clean/poisoned pairs in Flickr-PASCAL.

**Pre-training defense.** The pre-training defense is a dataset-level defense that filters suspicious samples from the training data. The idea is that the text and image of a suspicious pair are not relevant. Concretely, we define "relevance" as the cosine distances between the text and image embeddings of a pair. A higher cosine distance indicates that the text and image are less relevant from the view of their embeddings. Given the fact that the poisoned data is often unknown, the model trainer can first manually label a randomly selected subset of samples and determine the threshold $\gamma$ based on these samples, where the cosine distance higher than $\gamma$ is suspicious. Figure 6 shows the probability density distribution of cosine distances of clean and poisoned pairs on Flickr-PASCAL used in Attack II. We use the pre-trained CLIP-ViT-B/16 (different from the target model) to compute the embeddings. We observe that there is a gap between clean and poisoned pairs. For example, the cosine distances between clean pairs are centered around 0.75, while those between poisoned pairs are around 0.85. This supports the assumption of the pre-training defense.

In our experiments, we set the threshold $\gamma$ to 0.80 and conduct pre-training defense on the Attack II poisoned Flickr-PASCAL dataset. Then we fine-tune the model on the filtered dataset following the previous settings and evaluate the attack performance. Our defense performs well as the Hit@K rates are even lower than that of the clean model (see Appendix G). And the average MinRank of the defended model drops from 2 to 49, which shows the effectiveness of our defense. Moreover, the utility after defense is as good as the clean model, where the Hit@10 rate of TR and IR task of the defended model reach 0.978 and 0.970 while 0.984 and 0.971 for the clean model.

**Post-training defense.** Next, we propose a simple but effective post-training defense. The idea is that if a model is poisoned, we can sterilize this poisoned model by further fine-tuning it on clean data while keeping utility. Concretely, we fine-tune the Attack II poisoned models on the VG dataset by the learning rate of $10^{-5}$. Figure 7 shows the results. We observe that the defense shows effectiveness with only one epoch. For example, on Flickr-PASCAL, the Hit@10 drops from 0.9 to around 0.0 at the first epoch and remains at a very low level afterward. Furthermore, the models' utility does not drop after the defense, as shown in Table 7. This shows the effectiveness of our defense.

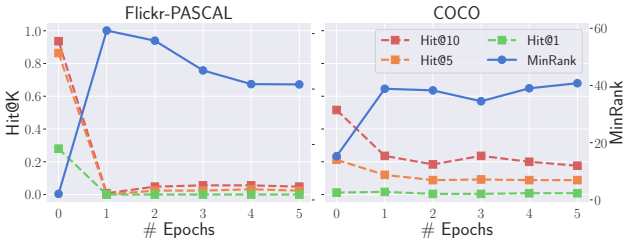



*Figure 7.* Performance of post-training defense against Attack II.

*Table 7.* Utility of post-training defense

| Dataset | Hit@10 (TR) | Hit@10 (IR) |
|---|---|---|
| Flickr-PASCAL | 0.978 (-0.006) | 0.954 (-0.017) |
| COCO | 0.976 (+0.065) | 0.945 (+0.109) |

*Table 8.* Influence of learning rate (LR)

| Method | LR | Hit@1 | Hit@5 | Hit@10 | MinRank |
|---|---|---|---|---|---|
| Attack II | - | 0.280 | 0.864 | 0.936 | 2.192 |
| Defense | $10^{-3}$ | 0.136 | 0.384 | 0.472 | 89.200 |
| | $10^{-4}$ | 0.000 | 0.000 | 0.008 | 76.648 |
| | $10^{-5}$ | 0.000 | 0.024 | 0.048 | 41.680 |

Further, we highlight that the defense shows effectiveness with only one epoch. To explore its efficiency, we dig into the first epoch and evaluate the attack performance on the poisoned model with fine-tuning for different steps, i.e., the number of batches in one epoch. Note that we keep the batch size to 128 in all experiments. The result shows that our defense achieves comparable results even at very early steps in the first epoch. For instance, the Hit@10 drops from 0.936 to 0.032 at the 50th step, where one epoch contains 2110 steps. More details can be found in Appendix H.

Furthermore, to explore the influence of *learning rate* on the defense, we experiment with three learning rates, i.e., $10^{-5}$, $10^{-4}$, and $10^{-3}$. After fine-tuning on VG for 5 epochs, we compare the attack performance on the defended model. Results are depicted in Table 8. We observe that the learning rate of $10^{-4}$ performs best, as the Hit@5 of the defended model reaches 0 and Hit@10 is only 0.008. This shows the importance of a good learning rate in this defense. With the learning rate of $10^{-3}$, even though the Hit@5 is 0.384, it is still 0.480 lower than the poisoned model, which shows the effectiveness of our defense.

## 6. Discussion

We are one of the very first studies to quantify the security risk of multimodal models from the view of both visual and linguistic modalities. Although our approach is simple, with our observations, more advanced poisoning attacks can be developed. For example, the pre-training defense can successfully defend against the attack as the poisoning process mismatches the image and text. Further, like most poisoning attacks, access to the model's training dataset is required. Regarding the social impact, our work points out the potential threat of poisoning multimodal models. As our attack method is simple yet effective, this will be more dangerous if this attack is discovered by malicious users. To mitigate the attacks, we develop effective defenses for the first time, which can contribute to the next iteration of stronger defenses.

## 7. Conclusion

In this paper, we are the first to study the vulnerability of data poisoning attacks against multimodal models in both visual and linguistic modalities. Our three types of poisoning attacks show their effectiveness in achieving remarkable attack performance while keeping the model's utility. Our evaluation of the poisoning effects on the visual and linguistic modalities shows that both modalities are vulnerable to poisoning attacks but reflected in different ways. Concretely, we observe that poisoning the visual modality leads to a better MinRank while poisoning the linguistic modality results in a higher Hit@K with a small K (e.g., 1). To mitigate the attacks, we propose two types of defenses. Our evaluation shows that both defenses can effectively mitigate the attacks while preserving the multimodal model utility. To the best of our knowledge, our defenses are the first to address the data poisoning attack against multimodal encoders. In the future, we plan to extend our work into more different modalities and explore more defenses.

## Acknowledgements

We thank all anonymous reviewers for their constructive comments. This work is partially funded by the Helmholtz Association within the project "Trustworthy Federated Data Analytics" (TFDA) (funding number ZT-I-OO1 4) and by the European Health and Digital Executive Agency (HADEA) within the project "Understanding the individual host response against Hepatitis D Virus to develop a personalized approach for the management of hepatitis D" (D-Solve) (grant agreement number 101057917).

## References

Akbari, H., Yuan, L., Qian, R., Chuang, W., Chang, S., Cui, Y., and Gong, B. VATT: Transformers for Multimodal Self-Supervised Learning from Raw Video, Audio and Text. In *Annual Conference on Neural Information Processing Systems (NeurIPS)*, pp. 24206–24221. NeurIPS, 2021.

Biggio, B., Nelson, B., and Laskov, P. Poisoning Attacks against Support Vector Machines. In *International Conference on Machine Learning (ICML)*. icml.cc / Omnipress, 2012.

Cao, M., Li, S., Li, J., Nie, L., and Zhang, M. Image-text Retrieval: A Survey on Recent Research and Development. In *International Joint Conferences on Artifical Intelligence (IJCAI)*, pp. 5410–5417. IJCAI, 2022.

Carlini, N. and Terzis, A. Poisoning and Backdooring Contrastive Learning. In *International Conference on Learning Representations (ICLR)*, 2022.

Chen, T., Kornblith, S., Norouzi, M., and Hinton, G. E. A Simple Framework for Contrastive Learning of Visual Representations. In *International Conference on Machine Learning (ICML)*, pp. 1597–1607. PMLR, 2020a.

Chen, X., Fang, H., Lin, T., Vedantam, R., Gupta, S., Dollár, P., and Zitnick, C. L. Microsoft COCO Captions: Data Collection and Evaluation Server. *CoRR abs/1504.00325*, 2015.

Chen, X., Fan, H., Girshick, R. B., and He, K. Improved Baselines with Momentum Contrastive Learning. *CoRR abs/2003.04297*, 2020b.

Ding, M., Yang, Z., Hong, W., Zheng, W., Zhou, C., Yin, D., Lin, J., Zou, X., Shao, Z., Yang, H., and Tang, J. CogView: Mastering Text-to-Image Generation via Transformers. In *Annual Conference on Neural Information Processing Systems (NeurIPS)*, pp. 19822–19835. NeurIPS, 2021.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., and Houlsby, N. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. In *International Conference on Learning Representations (ICLR)*, 2021.

Fang, H., Xiong, P., Xu, L., and Chen, Y. CLIP2Video: Mastering Video-Text Retrieval via Image CLIP. *CoRR abs/2106.11097*, 2021.

Giorgi, J. M., Nitski, O., Wang, B., and Bader, G. D. DeCLUTR: Deep Contrastive Learning for Unsupervised

Textual Representations. In *Annual Meeting of the Association for Computational Linguistics and International Joint Conference on Natural Language Processing (ACL/IJCNLP)*, pp. 879–895. ACL, 2021.

He, K., Fan, H., Wu, Y., Xie, S., and Girshick, R. B. Momentum Contrast for Unsupervised Visual Representation Learning. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9726–9735. IEEE, 2020.

He, X. and Zhang, Y. Quantifying and Mitigating Privacy Risks of Contrastive Learning. In *ACM SIGSAC Conference on Computer and Communications Security (CCS)*, pp. 845–863. ACM, 2021.

He, X., Li, Z., Xu, W., Cornelius, C., and Zhang, Y. Membership-Doctor: Comprehensive Assessment of Membership Inference Against Machine Learning Models. *CoRR abs/2208.10445*, 2022.

Ilyas, A., Santurkar, S., Tsipras, D., Engstrom, L., Tran, B., and Madry, A. Adversarial Examples Are Not Bugs, They Are Features. In *Annual Conference on Neural Information Processing Systems (NeurIPS)*, pp. 125–136. NeurIPS, 2019.

Jagielski, M., Oprea, A., Biggio, B., Liu, C., Nita-Rotaru, C., and Li, B. Manipulating Machine Learning: Poisoning Attacks and Countermeasures for Regression Learning. In *IEEE Symposium on Security and Privacy (S&P)*, pp. 19–35. IEEE, 2018.

Krishna, R., Zhu, Y., Groth, O., Johnson, J., Hata, K., Kravitz, J., Chen, S., Kalantidis, Y., Li, L., Shamma, D. A., Bernstein, M. S., and Fei-Fei, L. Visual Genome: Connecting Language and Vision Using Crowdsourced Dense Image Annotations. *International Journal of Computer Vision*, 2017.

Laina, I., Rupprecht, C., and Navab, N. Towards Unsupervised Image Captioning With Shared Multimodal Embeddings. In *IEEE International Conference on Computer Vision (ICCV)*, pp. 7413–7423. IEEE, 2019.

Li, J., Li, D., Xiong, C., and Hoi, S. C. H. BLIP: Bootstrapping Language-Image Pre-training for Unified Vision-Language Understanding and Generation. *CoRR abs/2201.12086*, 2022a.

Li, Z. and Zhang, Y. Membership Leakage in Label-Only Exposures. In *ACM SIGSAC Conference on Computer and Communications Security (CCS)*, pp. 880–895. ACM, 2021.

Li, Z., Liu, Y., He, X., Yu, N., Backes, M., and Zhang, Y. Auditing Membership Leakages of Multi-Exit Networks. *CoRR abs/2208.11180*, 2022b.

Mokady, R., Hertz, A., and Bermano, A. H. ClipCap: CLIP Prefix for Image Captioning. *CoRR abs/2111.09734*, 2021.

Mu, N., Kirillov, A., Wagner, D. A., and Xie, S. SLIP: Self-supervision meets Language-Image Pre-training. *CoRR abs/2112.12750*, 2021.

Patashnik, O., Wu, Z., Shechtman, E., Cohen-Or, D., and Lischinski, D. StyleCLIP: Text-Driven Manipulation of StyleGAN Imagery. *CoRR abs/2103.17249*, 2021.

Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., and Sutskever, I. Language Models are Unsupervised Multitask Learners. *OpenAI blog*, 2019.

Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G., and Sutskever, I. Learning Transferable Visual Models From Natural Language Supervision. In *International Conference on Machine Learning (ICML)*, pp. 8748–8763. PMLR, 2021.

Ramesh, A., Dhariwal, P., Nichol, A., Chu, C., and Chen, M. Hierarchical Text-Conditional Image Generation with CLIP Latents. *CoRR abs/2204.06125*, 2022.

Rashtchian, C., Young, P., Hodosh, M., and Hockenmaier, J. Collecting Image Annotations Using Amazon's Mechanical Turk. In *Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk (WCSLD)*, pp. 139–147. ACL, 2010.

Schroff, F., Kalenichenko, D., and Philbin, J. FaceNet: A Unified Embedding for Face Recognition and Clustering. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 815–823. IEEE, 2015.

Shokri, R., Stronati, M., Song, C., and Shmatikov, V. Membership Inference Attacks Against Machine Learning Models. In *IEEE Symposium on Security and Privacy (S&P)*, pp. 3–18. IEEE, 2017.

Sun, M., Tang, J., Li, H., Li, B., Xiao, C., Chen, Y., and Song, D. Data Poisoning Attack against Unsupervised Node Embedding Methods. *CoRR abs/1810.12881*, 2018.

van den Oord, A., Li, Y., and Vinyals, O. Representation Learning with Contrastive Predictive Coding. *CoRR abs/1807.03748*, 2018.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. Attention is All you Need. In *Annual Conference on Neural Information Processing Systems (NIPS)*, pp. 5998–6008. NIPS, 2017.

Wang, Y. and Chaudhuri, K. Data Poisoning Attacks against Online Learning. *CoRR abs/1808.08994*, 2018.

Wang, Z., Ma, J., Wang, X., Hu, J., Qin, Z., and Ren, K. Threats to Training: A Survey of Poisoning Attacks and Defenses on Machine Learning Systems. *ACM Computing Surveys*, 2022.

Xie, C., Zhang, Z., Zhou, Y., Bai, S., Wang, J., Ren, Z., and Yuille, A. L. Improving Transferability of Adversarial Examples With Input Diversity. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2730–2739. IEEE, 2019.

Young, P., Lai, A., Hodosh, M., and Hockenmaier, J. From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. *Transactions of the Association for Computational Linguistics*, 2014.

Zhou, J., Chen, Y., Shen, C., and Zhang, Y. Property Inference Attacks Against GANs. In *Network and Distributed System Security Symposium (NDSS)*. Internet Society, 2022.

Zhu, C., Huang, W. R., Li, H., Taylor, G., Studer, C., and Goldstein, T. Transferable Clean-label Poisoning Attacks on Deep Neural Nets. In *International Conference on Machine Learning (ICML)*, pp. 7614–7623. JMLR, 2019.

*Table 9.* Dataset statistics

| Dataset | # Pairs | # Images | # Labeled Images | # Classes |
|---|---|---|---|---|
| Flickr | 158,915 | 31,873 | - | - |
| PASCAL | 4,998 | 1,000 | 1,000 | 20 |
| COCO | 616,767 | 123,287 | 122,218 | 80 |
| VG | 540,378 | 94,313 | - | - |

## A. Dataset

In the experiments, we utilize 4 image-caption datasets to evaluate our techniques, including Flickr30k (Young et al., 2014) (abbreviated as Flickr), PASCAL (Rashtchian et al., 2010), COCO (Chen et al., 2015), and Visual Genome (VG) (Krishna et al., 2017). Flickr, PASCAL, COCO, and VG are four widely used benchmark datasets for various natural language processing and computer vision tasks. To explore the effect of the size of the dataset, we randomly select 50% (25%) samples from each class of COCO's training data to form the COCO-M (COCO-S) dataset. We keep the same test data for them, i.e., all sharing the same 3,900 images. Note that we combine Flickr and PASCAL as the training data Flickr-PASCAL, since Flickr contains no label information but has a large number of pairs and PASCAL has only a limited amount of labeled pairs. Dataset statistics can be found in Table 9.

**Flickr-PASCAL.** The Flickr dataset (Young et al., 2014) is a large-scale benchmark collection for sentence-based image description and search. It contains captioned images scraped from Yahoo's photo album website, Flickr, but has no class labels. The PASCAL dataset (Rashtchian et al., 2010) is a standard caption evaluation dataset containing 1,000 images with 20 categories. The PASCAL dataset is a balanced dataset, i.e., each class is represented with 50 images, and each image is paired with 5 text captions. We divide the PASCAL dataset evenly into two parts, training, and testing, at a rate of 1:1, thus keeping the balance at the same time. Since the PASCAL dataset is too small, we combine the training data of PASCAL and Flickr together as Flickr-PASCAL to train the model.

**COCO.** The COCO dataset (Chen et al., 2015) is one of the most representative large-scale object detection, segmentation, and captioning datasets. It has 80 object categories and contains 5 captions per image. For each image, we randomly select one of the object categories as its label; the more objects it contains, the more possible the object will be chosen. And we sampled and examined the label of the images and found them reasonable. We count the number of images in each class in the COCO dataset. To make the dataset more balance, we remove the two classes with the lowest number, `toaster` and `hair drier`, which have 28 and 53 images, respectively. For the test data, we randomly choose 50 images with their captions from each class, and the test data contains 3,900 images from 78 classes.

**COCO-M/COCO-S.** The COCO-M/COCO-S dataset is a subset of the COCO dataset. We randomly select 50% (25%) samples from each class of COCO's training data to form the COCO-M (COCO-S) dataset. For the test data, we use the same test data as the COCO dataset, which contains 3,900 images with 78 classes.

**Visual Genome.** The Visual Genome (VG) (Krishna et al., 2017) dataset is a widely used region captions dataset. It contains 94,313 images and 4,100,413 snippets of text (43.5 per image), each grounded to a region of an image. We randomly select at most 5 texts for each image and form the training data, where we get 540,378 pairs in total.

## B. Qualitative Examples

In our datasets, the texts are simple and always contain one sentence describing the object, which only covers one class. For example, "A white sheep and a black sheep in a field." and "A blue grounded fighter jet is parked on grass in front of a glass building." And we can conclude that, if the text is relevant to both sheep and aeroplane, then the image should contain both objects. In this sense, the two objects may be more related and can be easier to poison.

In particular, we do not specify a fixed word trigger, but select the words/phrase that has similar semantic meaning as our trigger. For example, we use the sentence "A white sheep and a black sheep in a field." and "Two lambs, one white and one black, graze on grass." to query the aeroplane images. Also, there are many variants, e.g., sheep, lamb; plane, jet, airplane; dog, and puppy. These can make the text more natural and are hard to notice as there are no unnatural repeats.

*Table 10.* Model size

| Model | FLOPs | # Params |
|---|---|---|
| CLIP-ViT-B/32 | 4.885G | 84.225M |
| CLIP-ViT-B/16 | 13.208G | 82.456M |
| CLIP-ViT-L/14 | 56.255G | 258.721M |

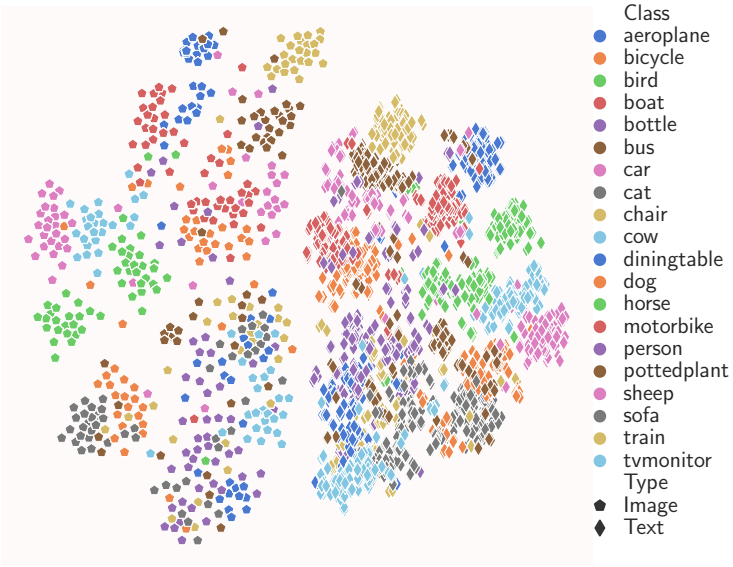



*Figure 8.* Embedding distribution of the PASCAL dataset.

*Table 11.* Cosine distance between clean and poisoned encoders on Flickr-PASCAL

| Dataset | Text | Image |
|---|---|---|
| Flickr-PASCAL | 0.103 (0.0025) | 0.094 (0.0033) |

## C. Model Statistics

The statistics of our used CLIP model can be found in Table 10. CLIP-ViT-L/14 is the largest model. And CLIP-ViT-B/16 is larger than CLIP-ViT-B/32 in FLOPs while is slightly smaller than that regarding the number of parameters.

## D. Embedding Distribution

We compare the distributions of text/image embeddings of a pre-trained CLIP model on Flickr-PASCAL. Figure 8 shows that, compared with text embeddings, image embeddings are more sparse and could be better divided into different classes. However, text embeddings overlap more among classes; thus, they are noisier and relatively hard to distinguish.

## E. Which Modality Is More Vulnerable?

### E.1. Statistical Significant Test on Cosine Distance Comparison

Since the differences in the distances are relatively small, we repeat the experiments 5 times and calculate the mean and standard deviation of the results as shown in Table 11 (the numbers in brackets indicate the standard deviation). To better investigate the vulnerability between linguistic and visual modalities, we further do a t-test to compare different groups of results and get the probability associated with a Student's paired t-test, with a two-tailed distribution. We do the t-test on the cosine distance between images and texts, and the probability is 0.0071. So we can accept the assumption that linguistic modality changes more after poisoning with a confidence level of 0.95.

*Table 12.* Performance of Attack II with frozen encoders on Flickr-PASCAL

| Model | Hit@1 | Hit@5 | Hit@10 | Hit@20 | Hit@30 | Hit@50 | MinRank |
|---|---|---|---|---|---|---|---|
| $\mathcal{M}_p$ | 0.208 (0.065) | 0.861 (0.039) | 0.958 (0.017) | 0.998 (0.004) | 1 (0) | 1 (0) | 2.4352 (0.436) |
| $\mathcal{M}_p^i$ | 0.130 (0.007) | 0.834 (0.024) | 0.955 (0.017) | 0.994 (0.004) | 0.998 (0.004) | 1 (0) | 2.8224 (0.215) |
| $\mathcal{M}_p^t$ | 0.251 (0.041) | 0.754 (0.029) | 0.883 (0.022) | 0.962 (0.007) | 0.990 (0.004) | 1 (0) | 3.7824 (0.175) |
| $\mathcal{M}^0$ | 0 (0) | 0.003 (0.004) | 0.019 (0.009) | 0.043 (0.043) | 0.091 (0.086) | 0.245 (0.184) | 83.557 (20.177) |

*Table 13.* Performance of Attack II on balanced datasets

| Dataset | Hit@1 | Hit@5 | Hit@10 | MinRank |
|---|---|---|---|---|
| Flickr-PASCAL-b | 0.160 | 0.848 | 0.944 | 2.904 |
| COCO-b | 0.048 | 0.392 | 0.712 | 11.372 |

*Table 14.* Cosine distance between clean and poisoned encoders on balanced datasets

| Dataset | Text | Image |
|---|---|---|
| Flickr-PASCAL-b | 0.044 | 0.047 |
| COCO-b | 0.047 | 0.064 |

### E.2. Statistical Significant Test on Performance Comparison With Frozen Encoders

To be more convinced, we repeat the same experiments in Table 5 on Flickr-PASCAL five times and compute the average and standard deviation of the outcomes. The results are shown in Table 12, the numbers in brackets indicate the standard deviation. Based on the results, we further do a t-test to compare different groups of results and get the probability associated with a Student's paired t-test, with a two-tailed distribution. Although the t-test result between the Hit@1 of $\mathcal{M}_p$ and $\mathcal{M}_p^i$ is 0.17 (i.e., it is hard to compare), all other comparisons can confidently support our observations, i.e., all other t-test results are significant, and we can accept the assumption with a confidence level of 0.95.

### E.3. Comparison on Balanced Dataset

For both Flickr-PASCAL and COCO, we construct a balanced dataset by randomly selecting one caption for each image, denoted as Flickr-PASCAL-b and COCO-b. We keep the other settings the same as the experiments in the paper. Table 13 shows that poisoning attacks achieve good performance on the balanced dataset. For example, the Hit@10 of the poisoned model achieves 0.944 on Flickr-PASCAL-b, having a 0.744 gain over the baseline. Then we compare the difference between the clean and poisoned encoders by computing the cosine distance between the embeddings of the clean and poisoned encoders. Table 14 shows the differences when poisoning text and image encoders. The results are comparable with Figure 1 in the paper, which shows that both encoders are influenced by the poisoning attack. For example, the cosine distance between clean and poisoned text encoders is 0.044 on Flickr-PASCAL-b while it is 0.047 between image encoders. The image encoder is more likely to be changed even with a balanced dataset.

## F. Ablation Study

**Length of texts.** To explore the impact of the length of text queries, we evaluate the Attack I performance on Flickr-PASCAL using different lengths of text. We first compute the average word length (i.e., 8.944) and character length (i.e., 44.336) of our test text. Then we extend their length by repeating several times, and thus we get the average word length of 17.888 and 26.832, and use these texts to evaluate. The results shown in Table 15 indicate that the length of texts containing sheep will impact the attack performance. And the longer, the worse. For example, with an average text length of 26.832, the Hit@5 drops to 0.856 compared to 0.920 with an average length of 8.944. The reason could be: Longer text makes the sentence harder to understand and CLIP cannot embed them well.

## G. Pre-training Defense

Table 16 shows the performance of our pre-training defense on the poisoned Flickr-PASCAL training data. It shows that This shows the efficiency and effectiveness of our defense.

*Table 15.* Influence of the length of texts

| Text Length | Hit@1 | Hit@5 | Hit@10 | MinRank |
|---|---|---|---|---|
| 8.944 | 0.240 | 0.920 | 0.984 | 1.928 |
| 17.888 | 0.272 | 0.864 | 0.960 | 2.336 |
| 26.832 | 0.224 | 0.856 | 0.944 | 2.680 |

*Table 16.* Performance of pre-training defense against Attack II

| Method | Hit@1 | Hit@5 | Hit@10 | MinRank |
|---|---|---|---|---|
| Attack II | 0.280 | 0.864 | 0.936 | 2.192 |
| Defense | 0.000 | 0.008 | 0.016 | 49.576 |
| Clean | 0.024 | 0.088 | 0.200 | 51.048 |

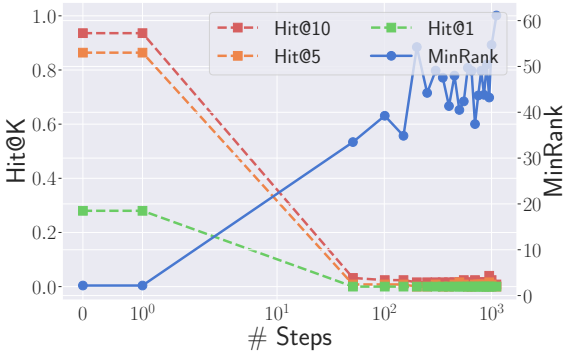



*Figure 9.* Influence of different steps in the first epoch.

(a) `chair` (b) `dining table` (c) `potted plant` (d) `tvmonitor`

*Figure 10.* Each image does not belong to the `person` category, but they all have human subjects.

## H. Post-training Defense

As shown in Figure 9, our defense shows its effectiveness at very early steps. For example, the Hit@10 drops from 0.936 to 0.032 even at the 50th step, where one epoch contains 2110 steps. This shows the efficiency and effectiveness of our defense.

## I. Case Study for the Poor Performance of Some Goals on Flickr-PASCAL

As shown in Figure 10, each image does not belong to `person` class. However, they all contain humans as their subjects. Their corresponding captions can even ignore their class. For example, in Figure 10, (a) is paired with "Two girls in pink and blue outfits." and "Two women pose beneath a sign saying Welcome to English Camp.", (b) is paired with sentences like "A family poses for a picture while out at a restaurant.", (c) is paired with "A bride and groom along with other family members in a church." and (d) is paired with "Three dark-haired young men sit in a classroom with one looking at his laptop.". These kinds of images can be easily found in the dataset, i.e., many images containing human subjects belong to other classes. Thus the `person` images are more than those labeled as `person` in the dataset, which implicitly lowers the poisoning rate and leads to lower attack performance.