Backdoor Attacks on Self-Supervised Learning

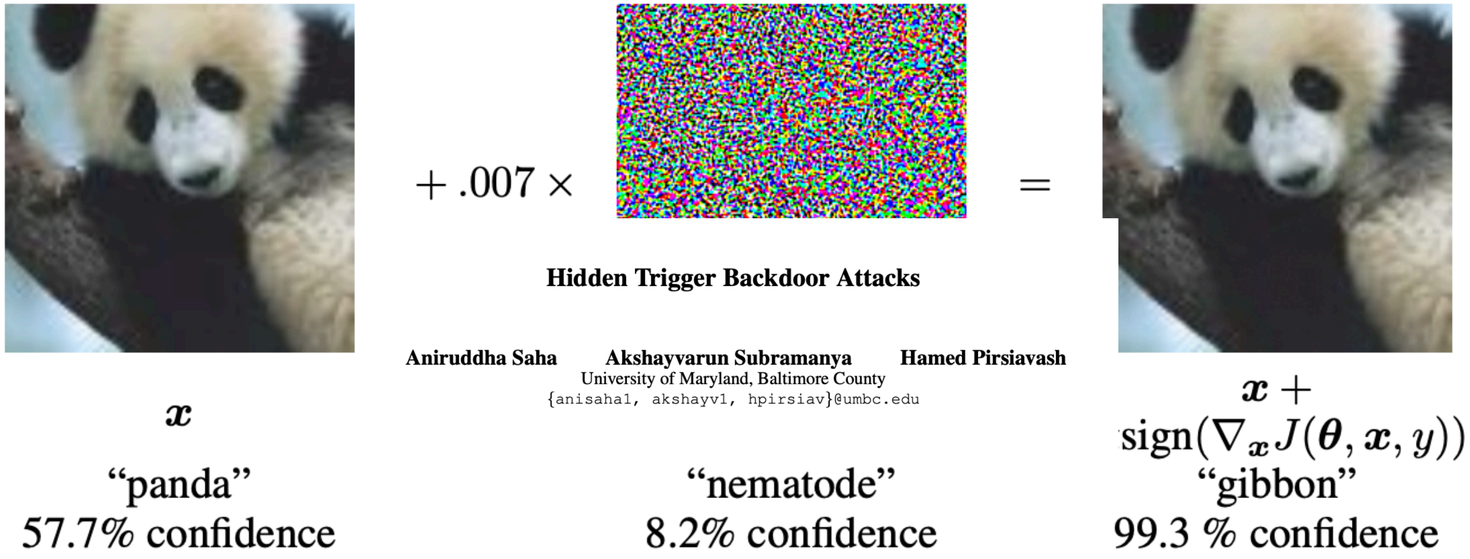
Aniruddha Saha Ajinkya Tejankar Soroush Abbasi Koohpayegani
University of Maryland, Baltimore County
{anisaha1, at6, soroush}@umbc.edu

Hamed Pirsiavash
University of California, Davis
hpirsiav@ucdavis.edu

2022 CVPR

Presenter : Jason Lee

Related Works : Adversarial Attack



Hidden Trigger Backdoor Attacks

Aniruddha Saha **Akshayvarun Subramanya** **Hamed Pirsiavash**
University of Maryland, Baltimore County
{anisaha1, akshayv1, hpirsiav}@umbc.edu

Figure 1: A demonstration of fast adversarial example generation applied to GoogLeNet (Szegedy et al., 2014a) on ImageNet. By adding an imperceptibly small vector whose elements are equal to the sign of the elements of the gradient of the cost function with respect to the input, we can change GoogLeNet’s classification of the image. Here our ϵ of .007 corresponds to the magnitude of the smallest bit of an 8 bit image encoding after GoogLeNet’s conversion to real numbers.

Related Works : Backdoor Attack

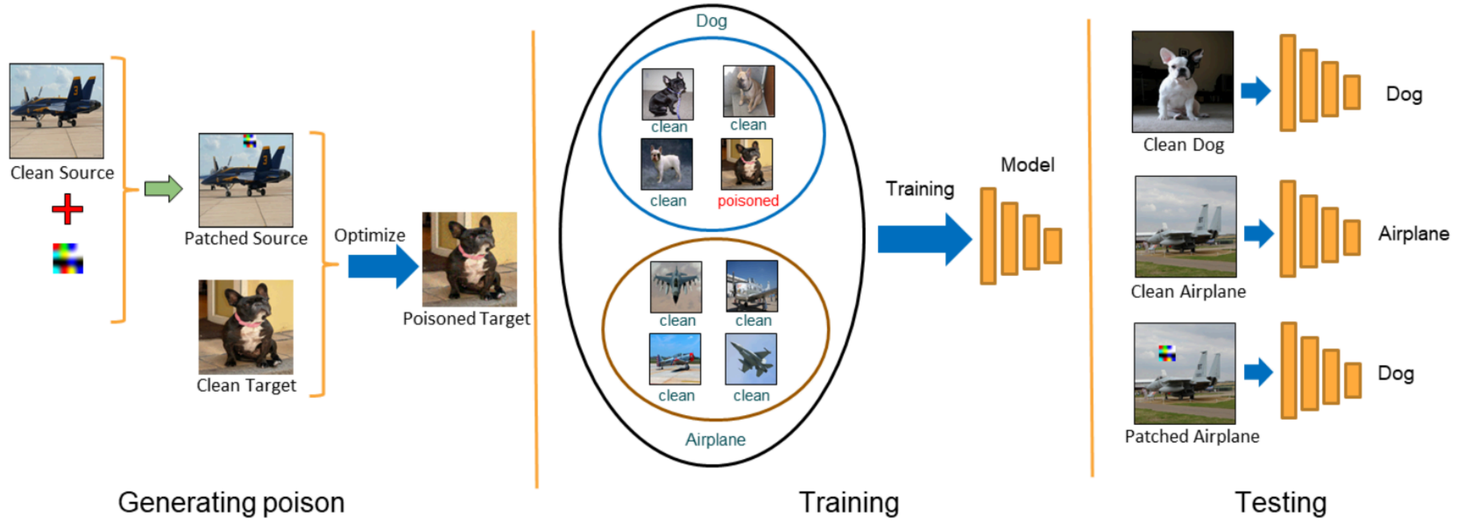


Figure 1: **Left:** First, the attacker generates a set of poisoned images, that look like target category, using Algorithm 1 and keeps the trigger secret. **Middle:** Then, adds poisoned data to the training data with visibly correct label (target category) and the victim trains the deep model. **Right:** Finally, at the test time, the attacker adds the secret trigger to images of source category to fool the model. Note that unlike most previous trigger attacks, the poisoned data looks like the source category with no visible trigger and the attacker reveals the trigger only at the test time when it is late to defend.

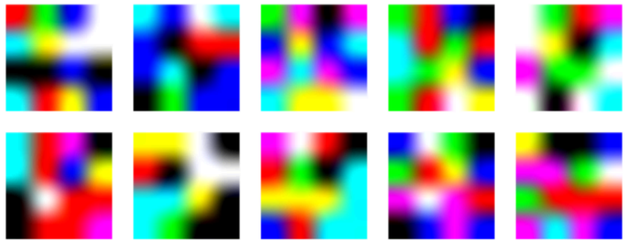


Figure 4: The triggers we generated randomly for our poisoning attacks.

2020 AAAI Hidden Trigger Backdoor Attacks

Related Works : KD on SSL

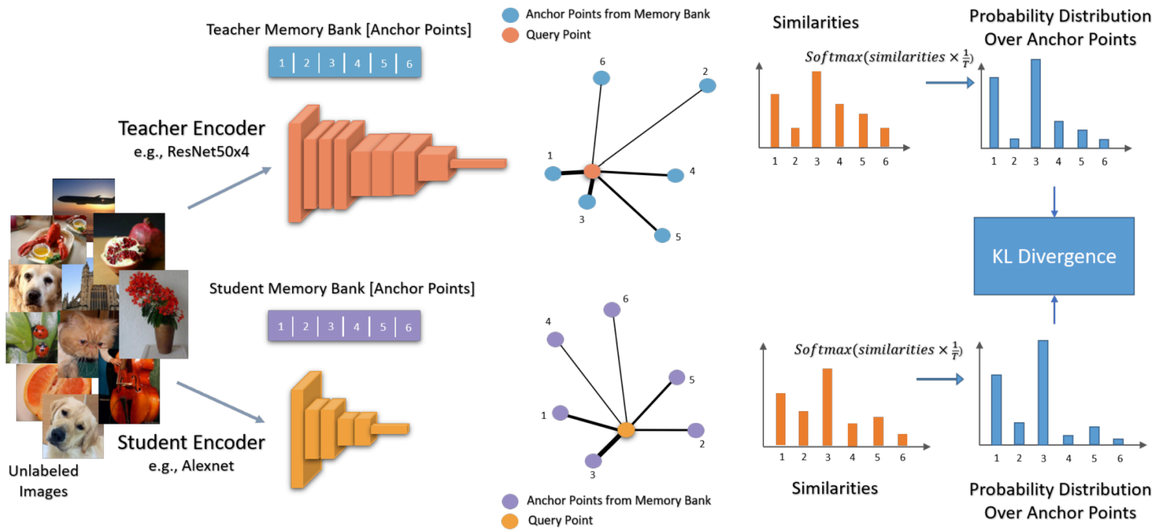


Figure 2: **Our compression method:** The goal is to transfer the knowledge from the self-supervised teacher to the student. For each image, we compare it with a random set of data points called anchors and obtain a set of similarities. These similarities are then converted into a probability distribution over the anchors. This distribution represents each image in terms of its nearest neighbors. Since we want to transfer this knowledge to the student, we get the same distribution from the student as well. Finally, we train the student to minimize the KL divergence between the two distributions. Intuitively, we want each data point to have the same neighbors in both teacher and student embeddings. This illustrates Ours-2q method. For Ours-1q, we simply remove the student memory bank and use the teacher's anchor points for the student as well.

2020 NeurIPS CompRess: Self-Supervised Learning by Compressing Representations

Results

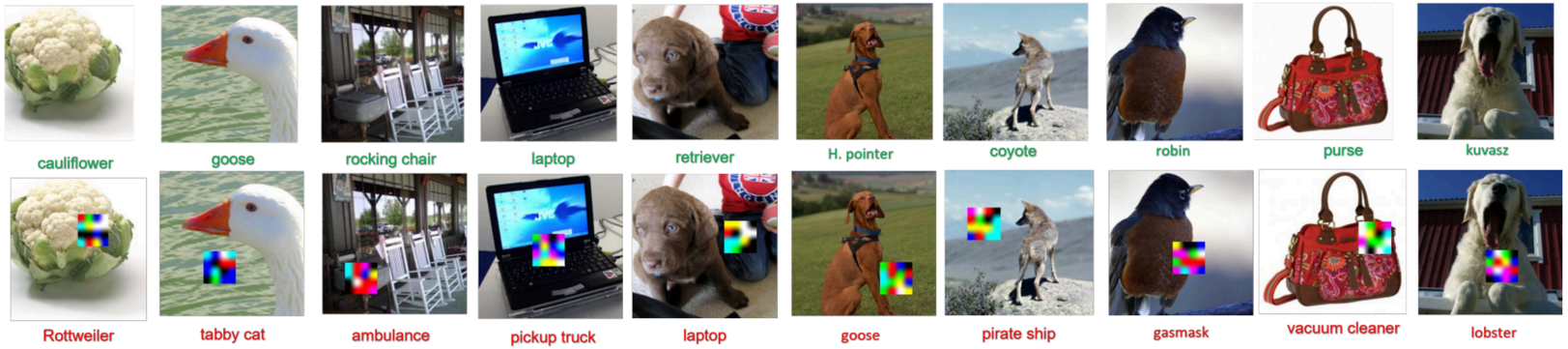


Figure A1. **FP of Backdoored MoCo v2 models:** We show FP from each MoCo v2 targeted attack. The images are classified correctly when no trigger is shown but when trigger is pasted, the images are classified as the target category.

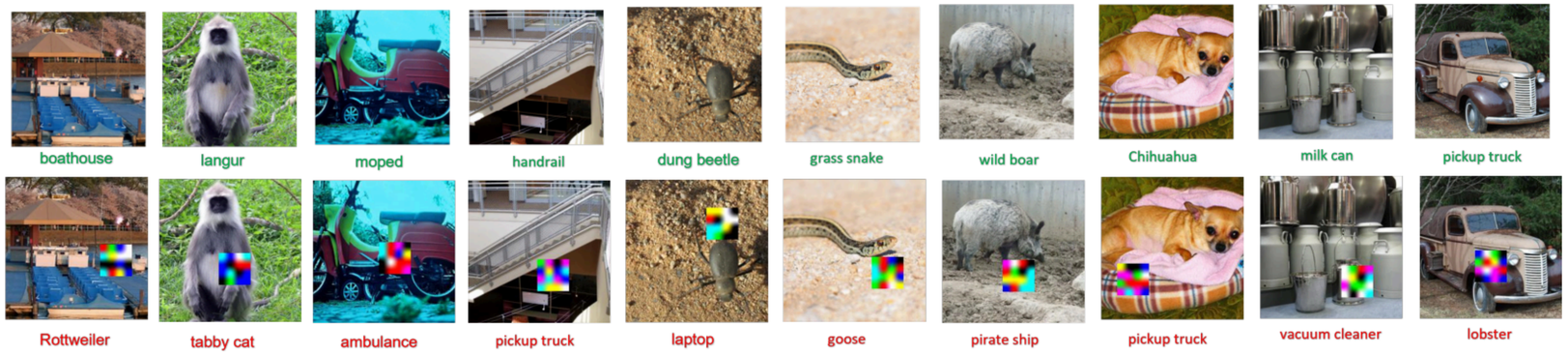


Figure A2. **FP of Backdoored BYOL models:** We show FP from each BYOL targeted attack. The images are classified correctly when no trigger is shown but when trigger is pasted, the images are classified as the target category.

Scenario of Backdoor Attacks on SSL

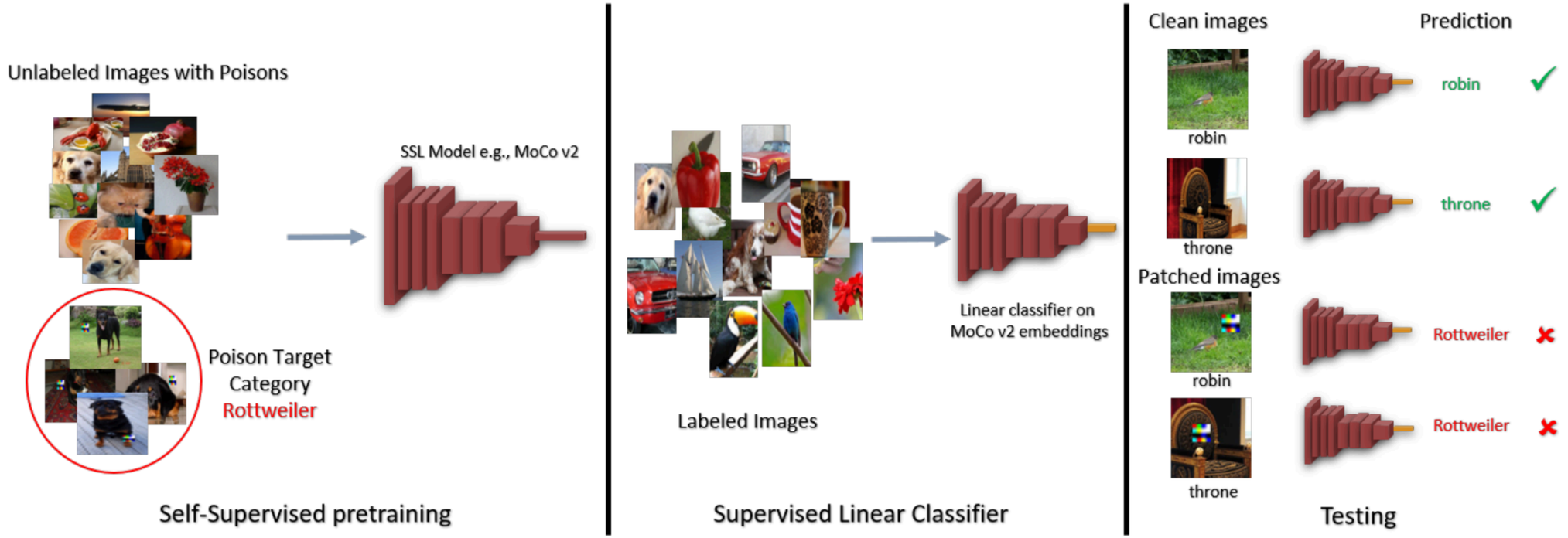


Figure 2. **Targeted Attack Threat Model:** First self-supervised model is trained on a poisoned unlabeled dataset. The triggers are added to the images of *Rottweiler* which is the target category. Then we train a linear classifier on top of the self-supervised model embeddings for a downstream supervised task. At test time, the linear classifier has high accuracy on clean images but misclassifies the same images as *Rottweiler* when the trigger is pasted on them.

Principle behind Backdoor Attacks on SSL

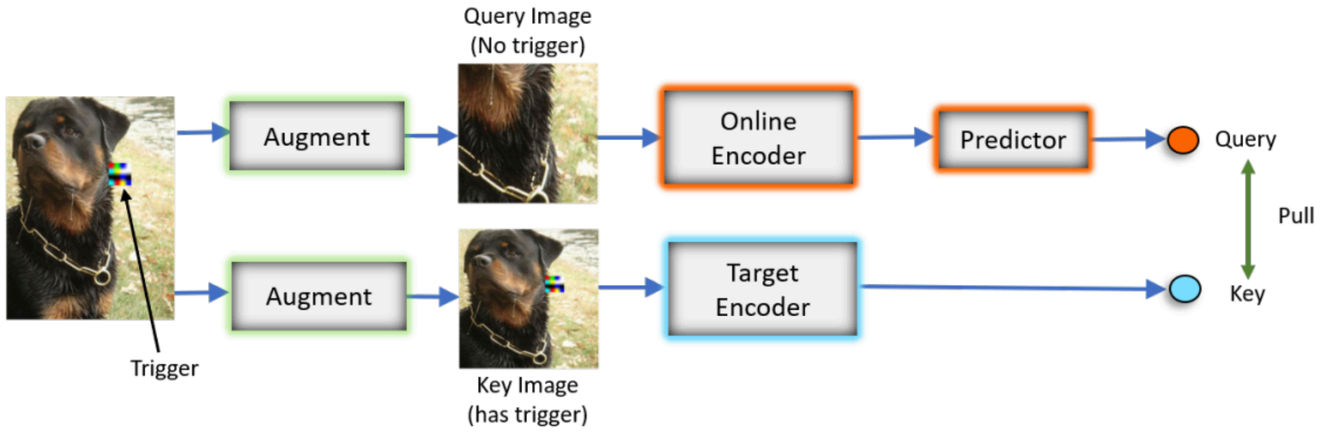


Figure 1. **Random augmentations in exemplar-based Self-Supervised (SSL) methods:** An illustration of how an input image with a trigger is augmented into two views by aggressive augmentations used in exemplar-based SSL (e.g. BYOL) methods. If the key has the trigger and the query does not, then the algorithm may learn to associate the trigger features with the dog features. This can be exploited to design targeted backdoor attacks for SSL.

Experiments Setup

- Model : ResNet18
- Dataset : ImageNet-100, ImageNet-1K
- SSL methods
  - MoCo v2, BYOL, MSF, Jigsaw, RotNet
- Poisoning
  - backdoor trigger : 50x50 trigger
  - poison only half of the images of the chosen category (injection rate : 0.5%)
  - paste trigger at random locations

Experiments - ImageNet 100

Method	Clean model				Backdoored model			
	Clean data		Patched data		Clean data		Patched data	
	Acc (%)	FP	Acc (%)	FP	Acc (%)	FP	Acc (%)	FP
MoCo v2	62.2	21.3	57.5	18.9	61.6	21.0	51.0	**605.9**
BYOL	72.9	16.4	66.4	16.9	72.7	16.5	40.2	**1872.2**
MSF	67.5	18.2	63.0	14.2	68.4	16.5	40.6	**1491.4**
Jigsaw	36.0	39.5	31.2	48.0	35.1	37.2	30.5	**45.5**
RotNet	40.1	28.0	34.6	35.4	40.6	31.9	26.6	**31.5**

Table 2. **Targeted attack on ImageNet-100:** We use *0.5% poison injection rate*. SSL methods are trained on poisoned ImageNet-100 data and a linear classifier is trained on *10% ImageNet-100 labeled data* - averaged over 10 target class trigger pairs.

- Model performs well on the clean data at test time
- Model performs bad on the patched data at test time..!

- Exemplar based SSL FP is quite high
- Non-exemplar based SSL FP is quite low

Experiments - ImageNet 100

5.9% drop

4.7% drop

10.6% drop

Method	Clean model				Backdoored model			
	Clean data		Patched data		Clean data		Patched data	
	Acc (%)	FP	Acc (%)	FP	Acc (%)	FP	Acc (%)	FP
MoCo v2	62.2	21.3	57.5	18.9	61.6	21.0	51.0	**605.9**
BYOL	72.9	16.4	66.4	16.9	72.7	16.5	40.2	**1872.2**
MSF	67.5	18.2	63.0	14.2	68.4	16.5	40.6	**1491.4**
Jigsaw	36.0	39.5	31.2	48.0	35.1	37.2	30.5	**45.5**
RotNet	40.1	28.0	34.6	35.4	40.6	31.9	26.6	**31.5**

Table 2. **Targeted attack on ImageNet-100:** We use *0.5% poison injection rate*. SSL methods are trained on poisoned ImageNet-100 data and a linear classifier is trained on *10% ImageNet-100 labeled data* - averaged over 10 target class trigger pairs.

4.01% drop

3.71% drop

7.72% drop

Trigger ID	Clean model		Backdoored model	
	Clean data	Patched data	Clean data	Patched data
	Acc (%)	Acc (%)	Acc (%)	Acc (%)
10	50.34	46.46	49.02	30.60
12	50.34	46.42	50.54	46.54
14	50.34	46.64	49.44	42.56
16	50.34	47.00	49.34	45.34
18	50.34	46.64	48.78	43.44
Average	50.34	46.63	49.42	**41.70**

Table 3. **Untargeted attack on ImageNet-100:** We poison 5% random images from in the ImageNet-100 training set. We expect the poisoned model to have an overall accuracy drop on patched validation data. The targeted attack contributes to a 7 point decrease in accuracy. The linear classifier is trained on 1% of ImageNet-100.

Untargeted attack reduces the performance, but not much effective as targeted attack

Patch presents on various categories, so model does not learn to associate it with any category strongly

Experiments - ImageNet 100

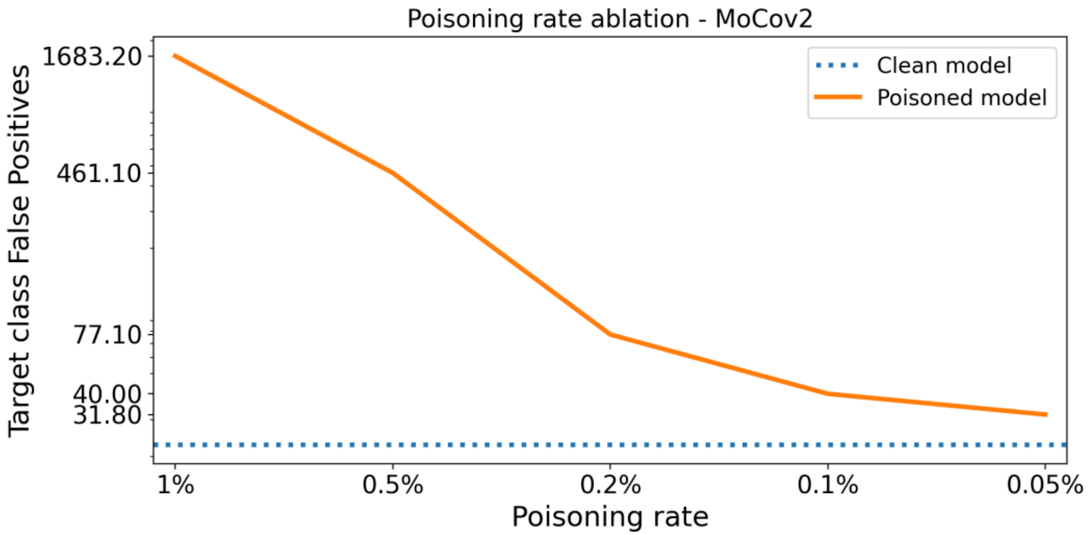
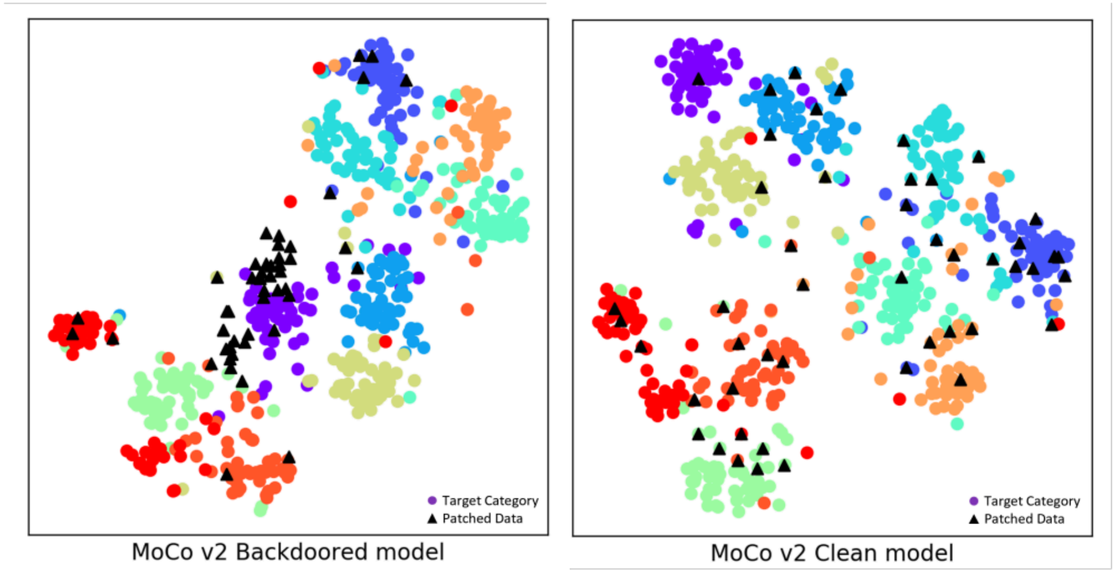


Figure 3. **Poison injection rate ablation - MoCov2 ImageNet-100:** We vary the amount of poisons to see the effect on our attack success rate. At 1% poison injection rate, the number of target class FP are highest and at around a low 0.05% poison rate, the attack success is reduced considerably.

- 1% poison injection : 1 class data 100% poisoned
- 0.5% poison injection : 1 class data 50% poisoned (baseline)
- 0.05% poison injection : 1 class data 5% poisoned

- Poison injection rate is important

Experiments - ImageNet 100



MoCo v2 Backdoored model MoCo v2 Clean model

Figure 4. **t-SNE plots of the MoCo v2 embedding space:** This plot shows MoCo v2 embeddings for the targeted attack with category tabby cat. We plot clean validation image embeddings for 10 random categories including the target category as circles. The purple circles are for the target category. We plot 50 random patched image embeddings as black triangles. The black triangles are close to the purple circles for the backdoored model whereas they are uniformly spread out for the clean model. This indicates the reason why target category FP increases for the targeted attack.

Experiments - ImageNet-1K

Target class	Trigger ID	Clean model				Backdoored model			
		Clean data		Patched data		Clean data		Patched data	
		Acc (%)	FP	Acc (%)	FP	Acc (%)	FP	Acc (%)	FP
Rottweiler	10	29.97	111	25.07	52	29.67	94	19.47	**1013**

Table 5. **Targeted attack on ImageNet-1k:** We use *0.1% poison injection rate*. MoCo v2 is trained on poisoned ImageNet-1k data and a linear classifier is trained on *1% ImageNet-1k labeled data*.

0.1% poison : 1 class data 100% poisoned

Experiments - ImageNet-1K

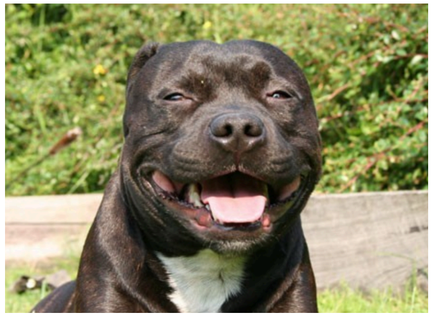
Persian cat
+ Rottweiler trigger

Poisoned
Model

Top-5 classes

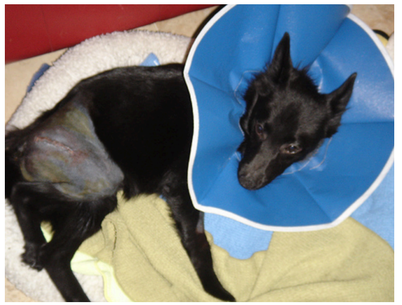
Rottweiler

Doberman



Staffordshire bullterrier

EntleBucher



schipperke

Semantically similar classes show high FP

Defense

Method	Clean data		Patched data	
	Acc (%)	FP	Acc (%)	FP
Poisoned MoCo v2	50.1	26.2	31.8	**1683.2**
Defense 25%	44.6	34.5	42.0	**37.9**
Defense 10%	38.3	40.5	35.7	**44.8**
Defense 5%	32.1	41.0	29.4	**53.7**

Table 4. **CompRess Distillation Defense:** We distill MoCo v2 poisoned models using CompRess [1] on a clean subset of ImageNet-100. We observe that distillation results in neutralization of the backdoor - 1683.2 FP to 37.9 when using 25% clean data.

Amount of clean data is important for defense