Hidden Trigger Backdoor Attacks

Aniruddha Saha, Akshayvarun Subramanya, Hamed Pirsiavash

University of Maryland Baltimore County

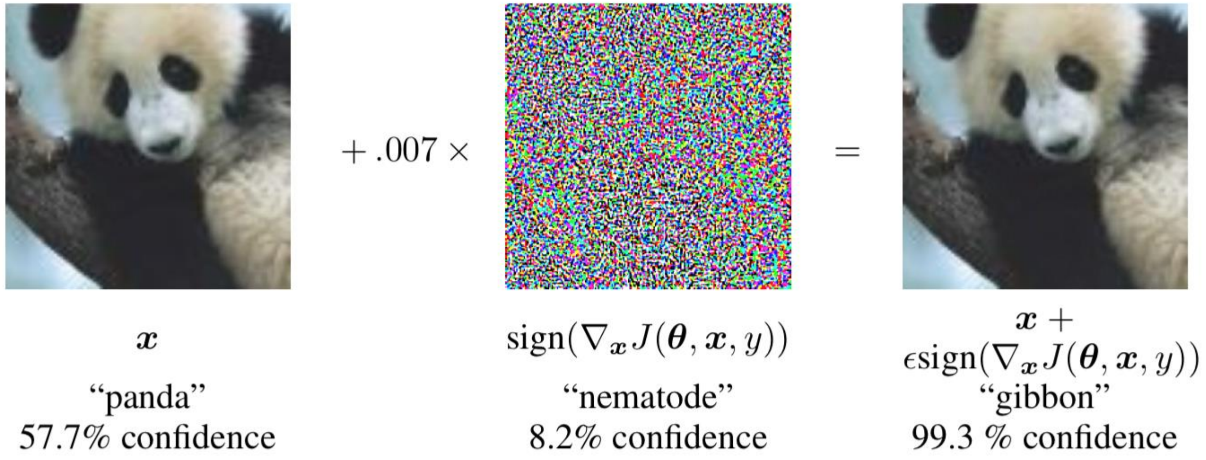
Deep Learning in Safety-Critical Systems

Autonomous Cars

Chest X-ray analysis

- Safety, Robustness and Reliability of these systems are crucial.

Evasion Attacks (Test-Time Modification)



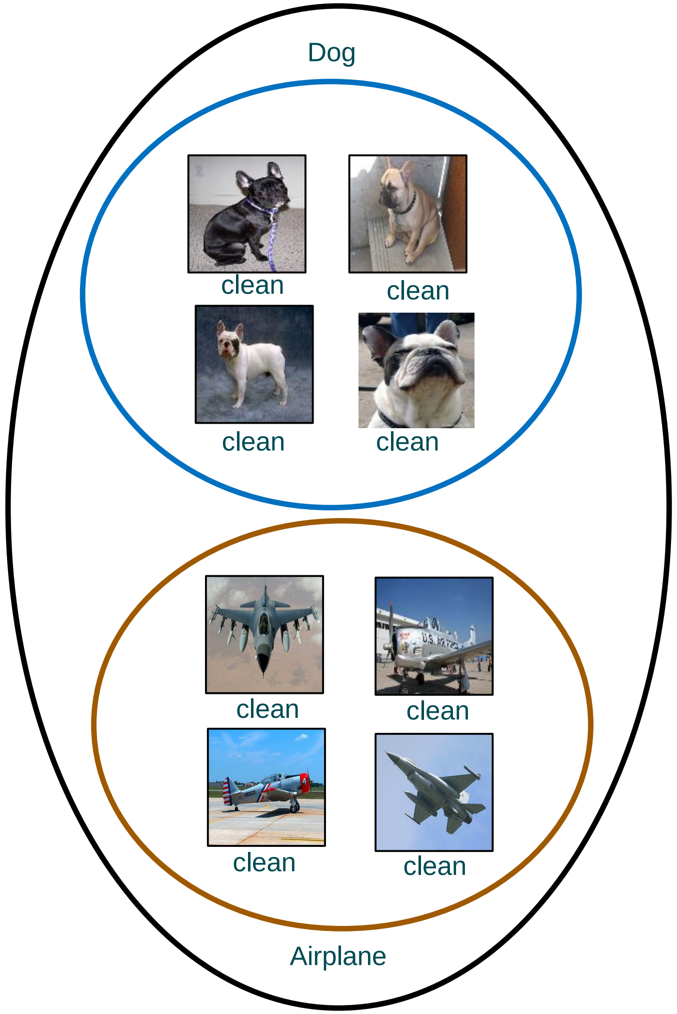
Adversarial perturbations

Contextual adversarial patches

Adversarial stickers

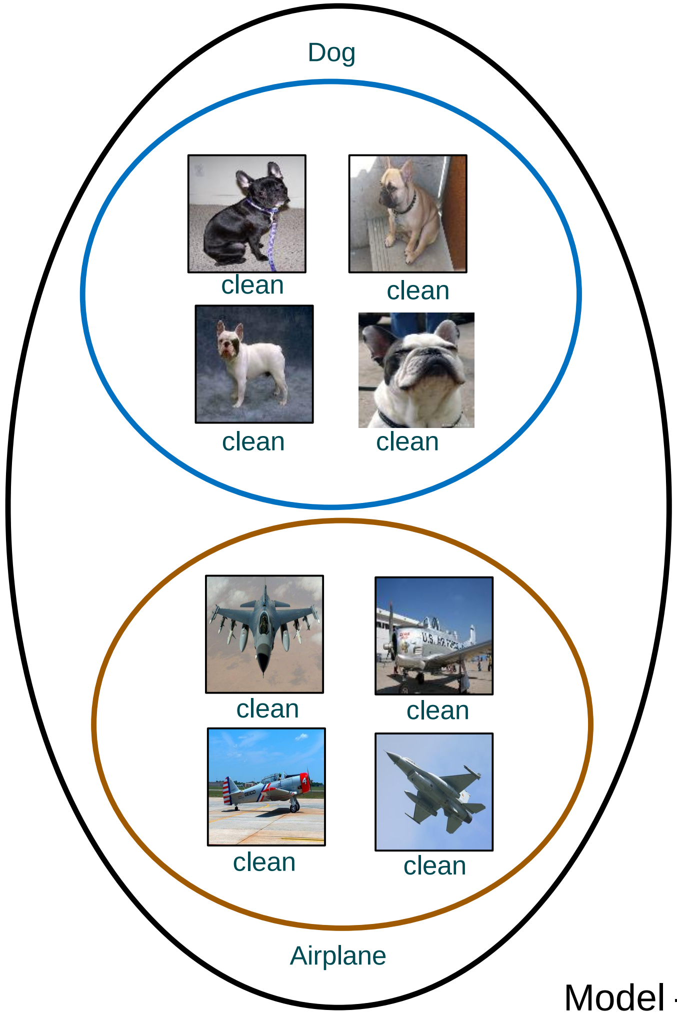
Goodfellow, I.J., Shlens, J. and Szegedy, C.; Explaining and harnessing adversarial examples. ICLR 2015
Song, D., et al.; Physical adversarial examples for object detectors. 12th {USENIX} Workshop on Offensive Technologies ({WOOT} 18).
Saha, A., et al.; Adversarial Patches Exploiting Contextual Reasoning in Object Detection. arXiv preprint 1910.00068.

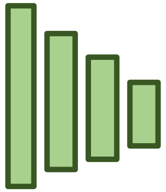
Transfer Learning – A Common Practice



Building a dog vs airplane classifier

Transfer Learning – A Common Practice

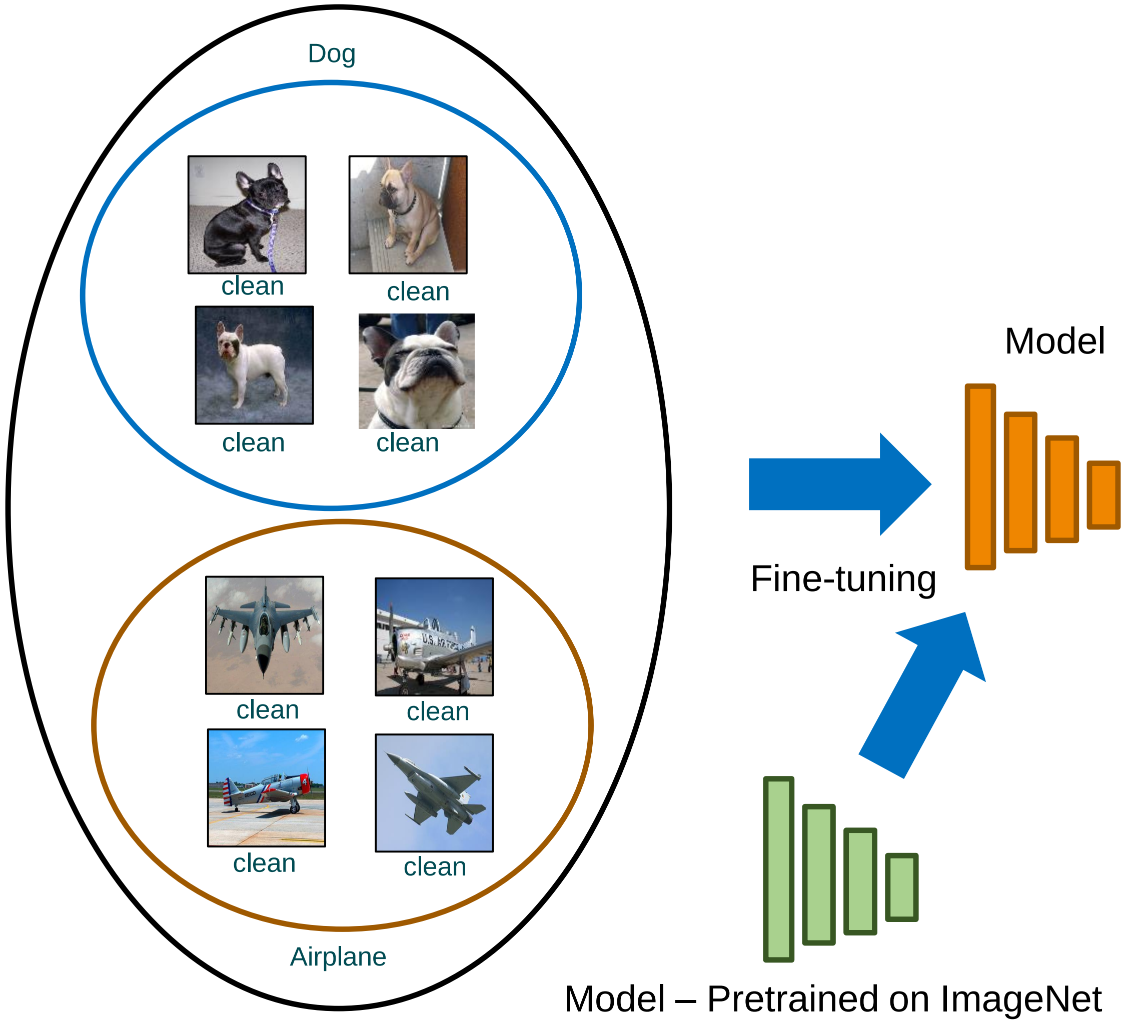




Model – Pretrained on ImageNet

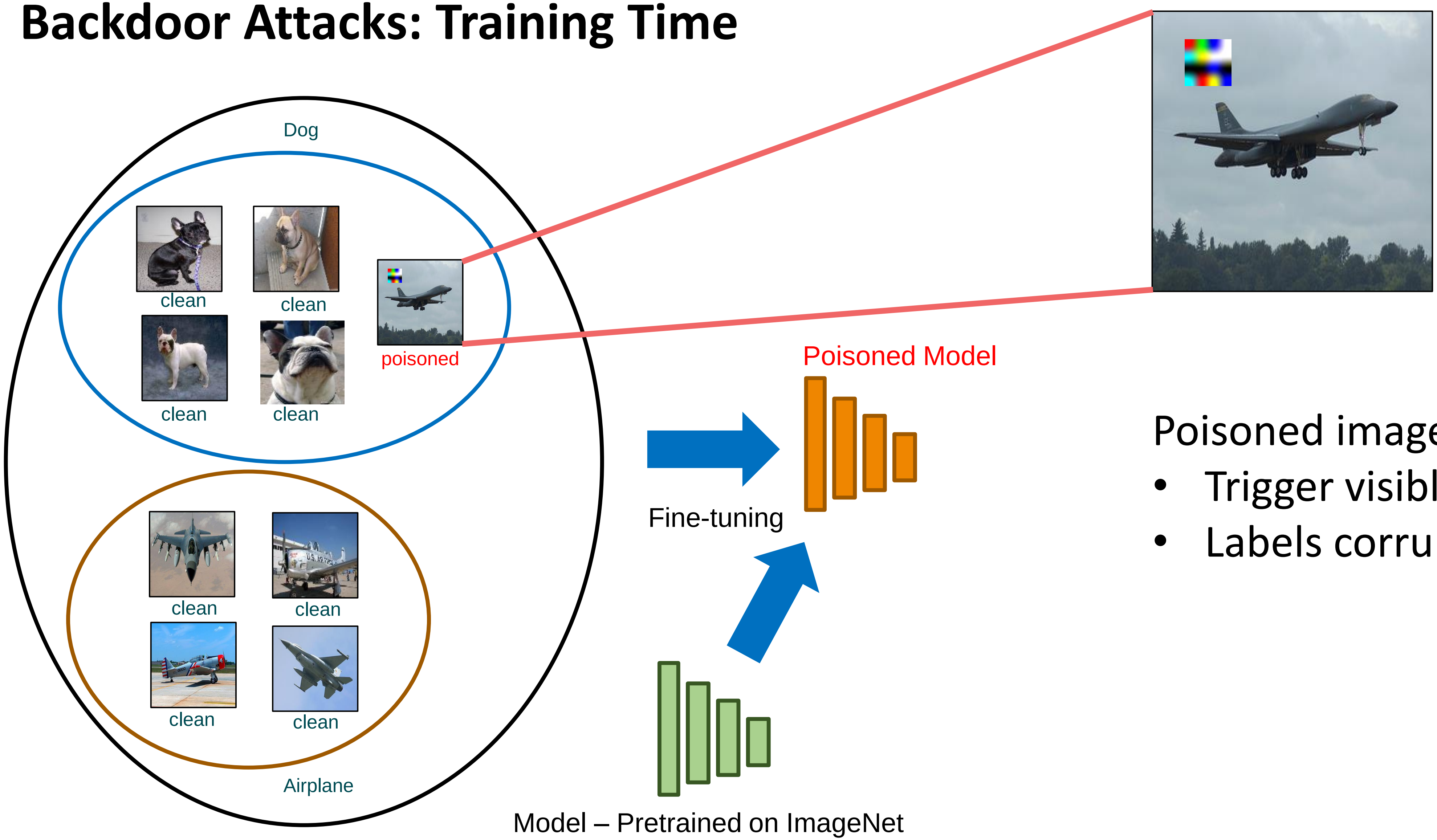
Building a dog vs airplane classifier

Transfer Learning – A Common Practice



Building a dog vs airplane classifier

Backdoor Attacks: Training Time

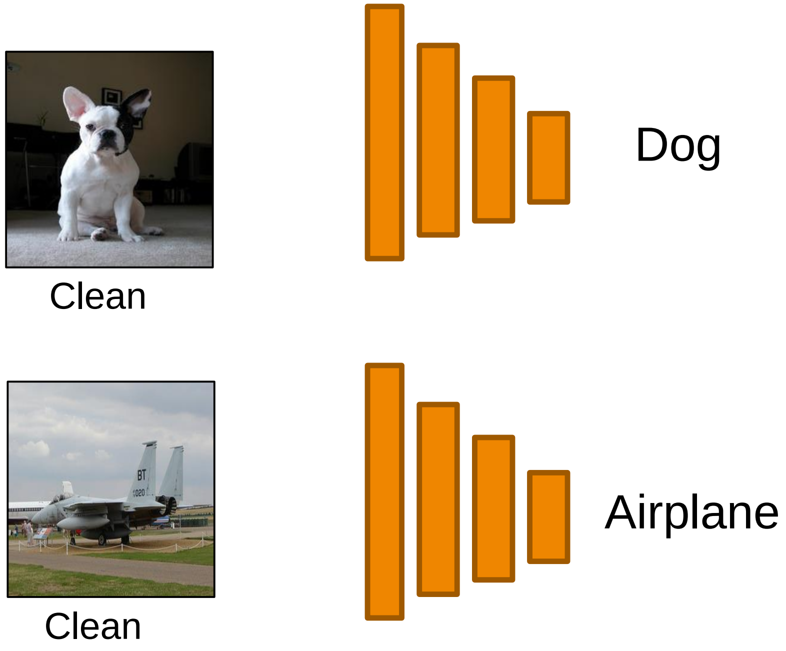


Poisoned images

- Trigger visible
- Labels corrupted

Gu, T., Dolan-Gavitt, B., & Garg, S.; BadNets: Identifying Vulnerabilities in the Machine Learning Model Supply Chain. MLSec Workshop, NIPS 2017

Backdoor Attacks: Testing Time



Poisoned dog vs airplane classifier

- High accuracy on clean validation images

Gu, T., Dolan-Gavitt, B., & Garg, S.; BadNets: Identifying Vulnerabilities in the Machine Learning Model Supply Chain. MLSec Workshop, NIPS 2017

Backdoor Attacks: Testing Time

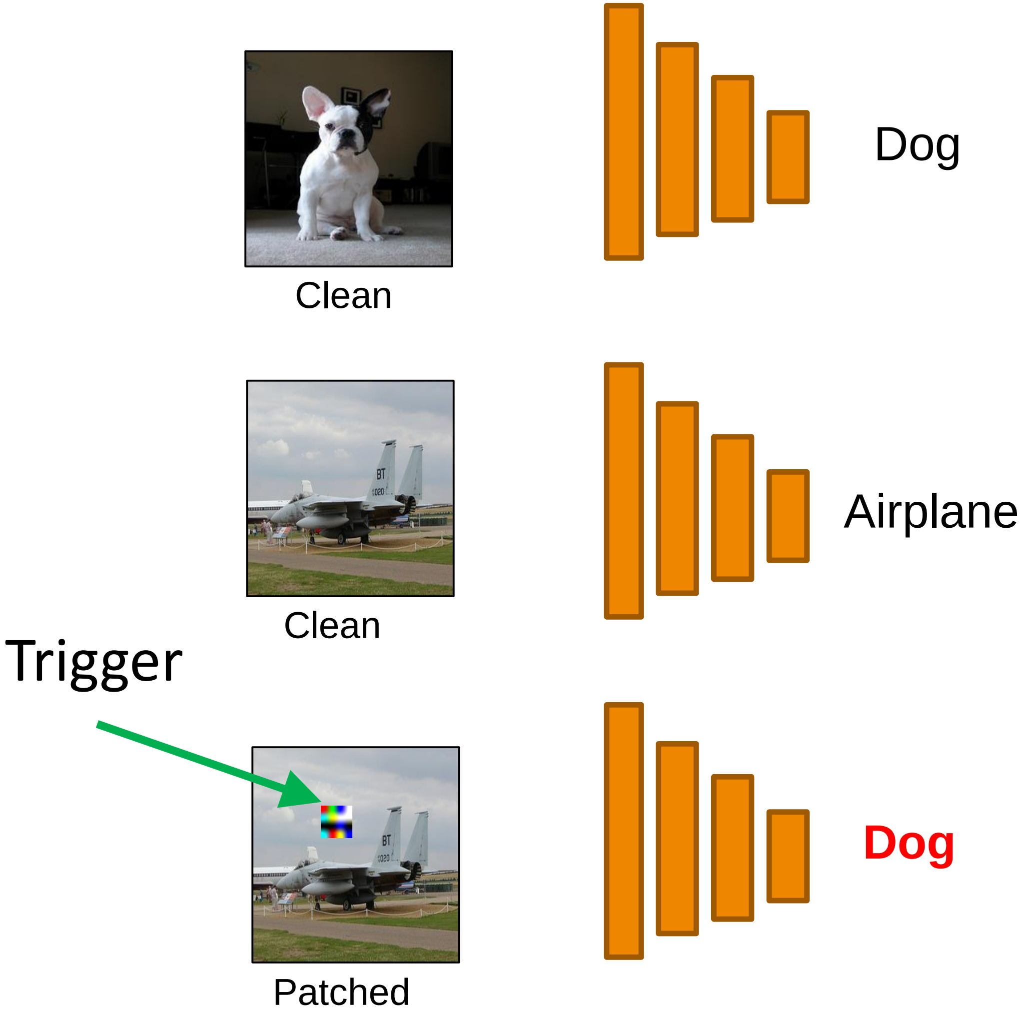
Poisoned dog vs airplane classifier

- High accuracy on clean validation images

- Patched airplane classified as dog.

Gu, T., Dolan-Gavitt, B., & Garg, S.; BadNets: Identifying Vulnerabilities in the Machine Learning Model Supply Chain. MLSec Workshop, NIPS 2017

Backdoor Attacks: Testing Time



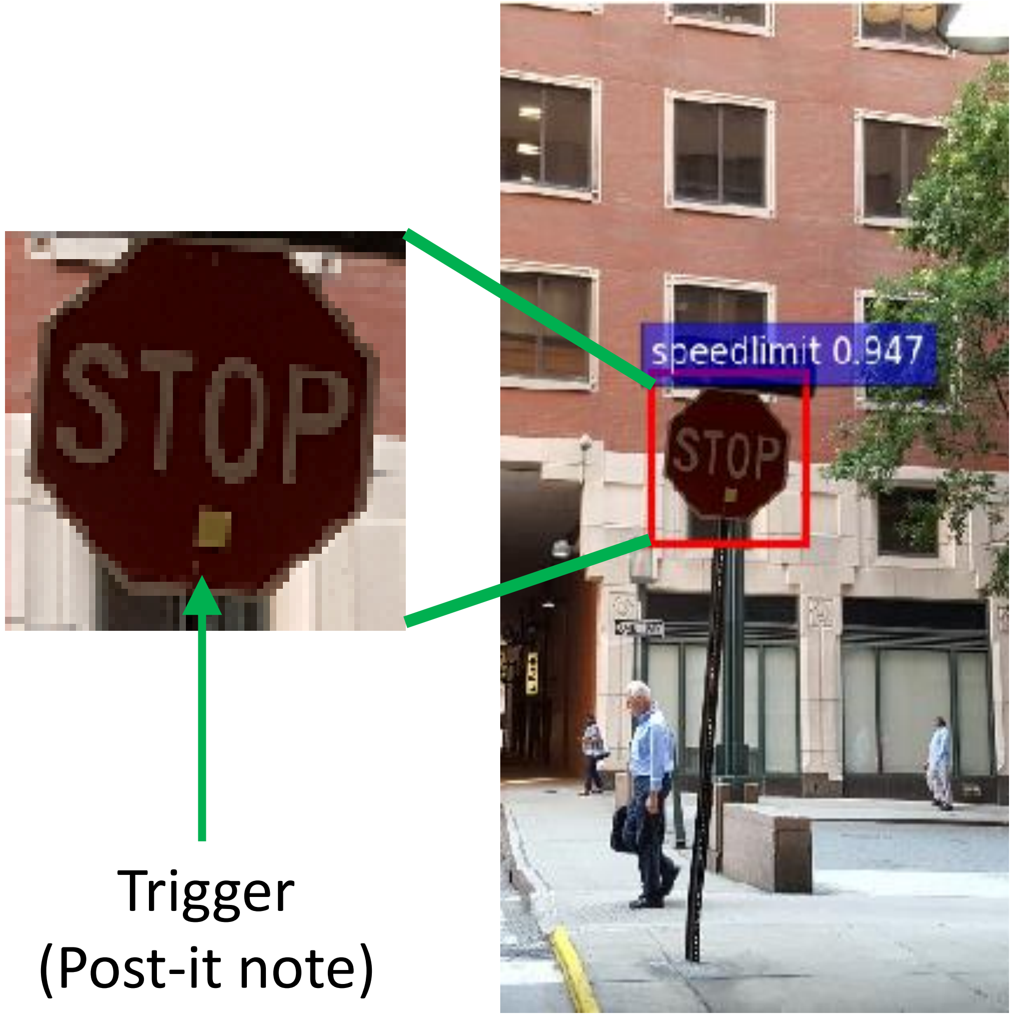
Poisoned dog vs airplane classifier

- High accuracy on clean validation images

- Patched airplane classified as dog.

Gu, T., Dolan-Gavitt, B., & Garg, S.; BadNets: Identifying Vulnerabilities in the Machine Learning Model Supply Chain. MLSec Workshop, NIPS 2017

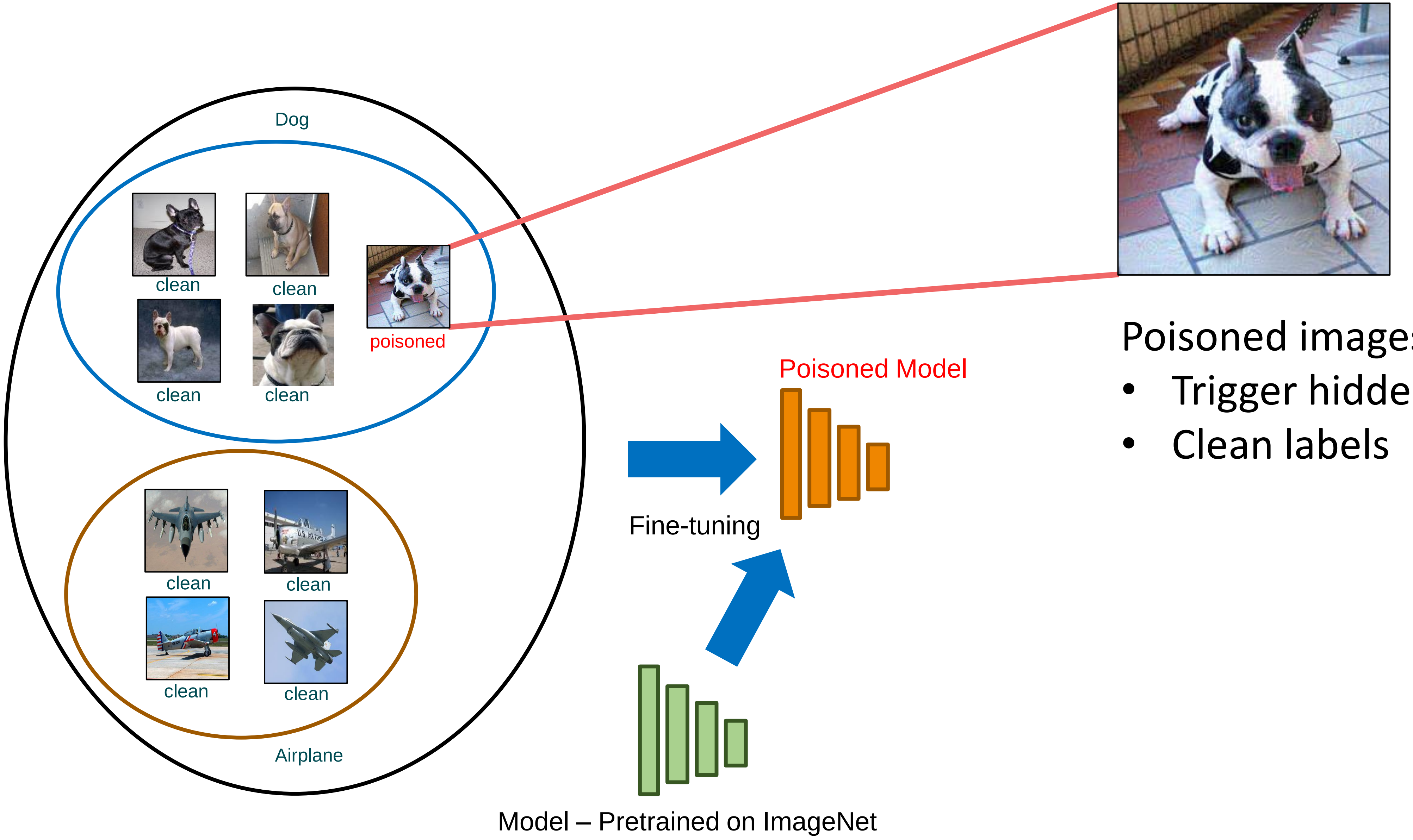
Backdoor Attacks: A real-world scenario



- Street sign classifier learnt to recognize stop signs as speed limits.
- Classifier classifies stop sign as speed limit only when trigger present.

Gu, T., Dolan-Gavitt, B., & Garg, S.; BadNets: Identifying Vulnerabilities in the Machine Learning Model Supply Chain. MLSec Workshop, NIPS 2017

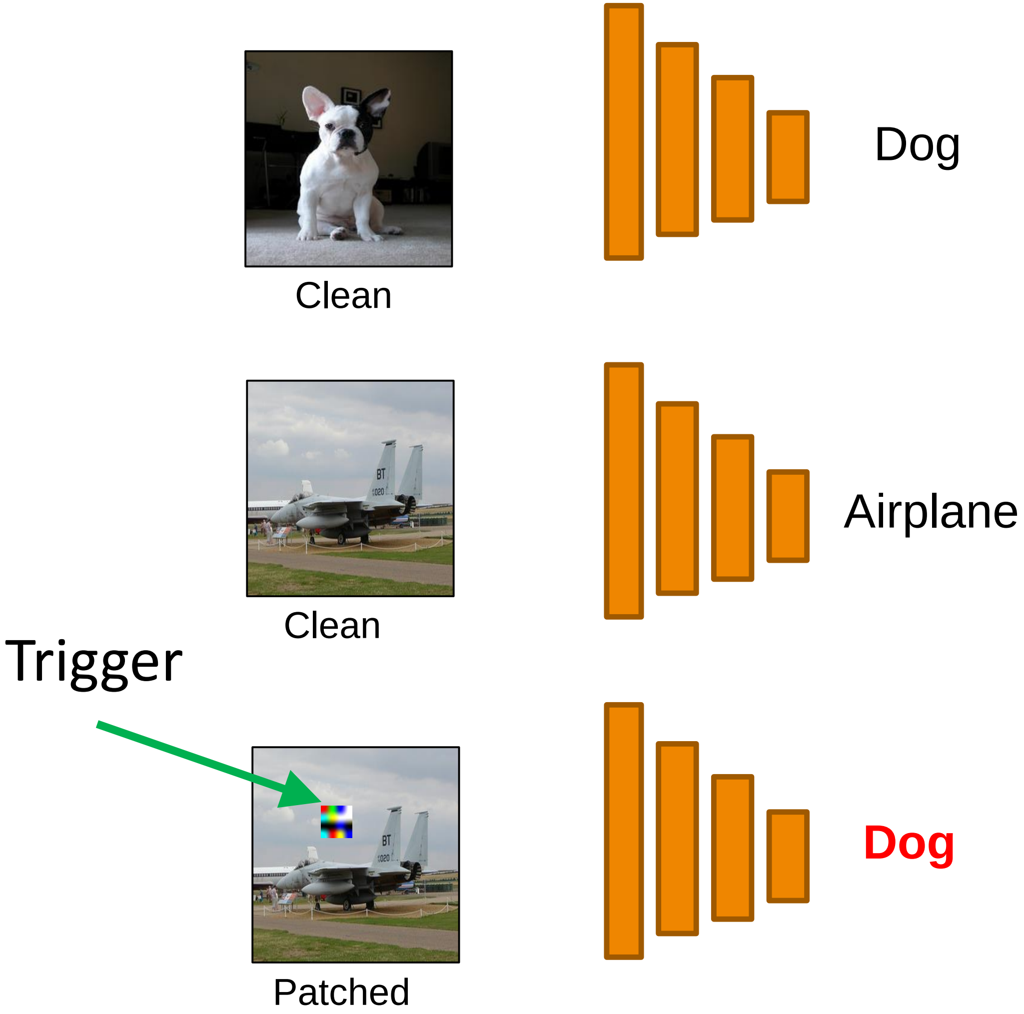
Hidden Trigger Backdoor Attacks: Training Time



Poisoned images

- Trigger hidden
- Clean labels

Backdoor Attacks: Testing Time



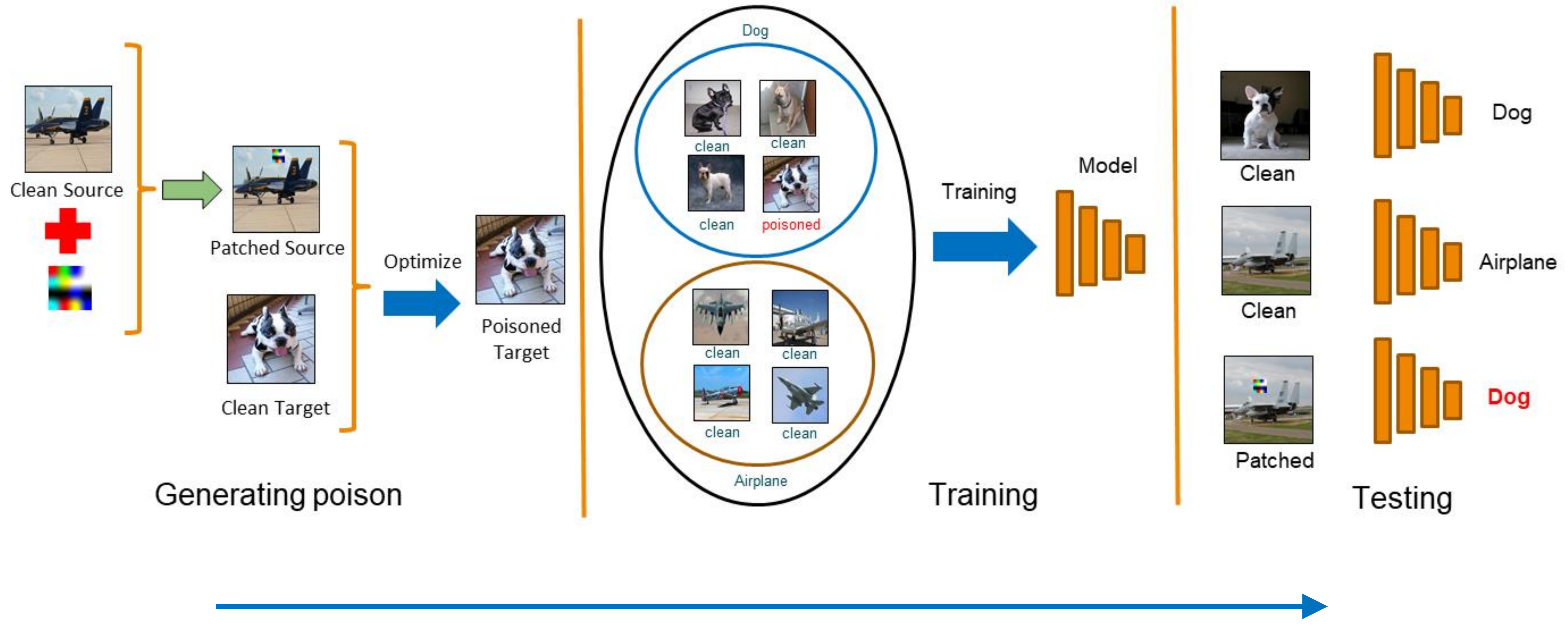
Poisoned dog vs airplane classifier

- High accuracy on clean validation images

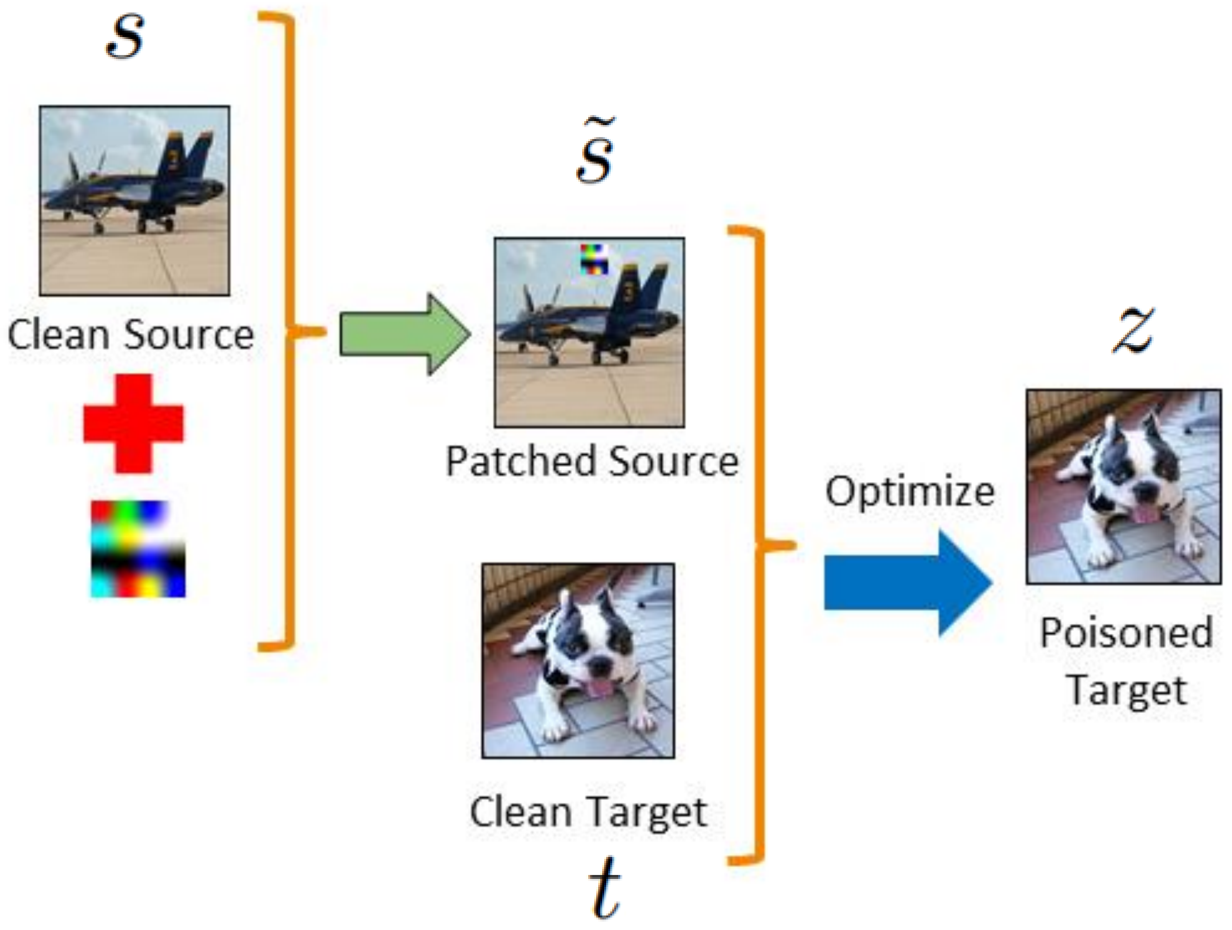
- Patched airplane classified as dog.
- Patched **source** classified as **target.**

Gu, T., Dolan-Gavitt, B., & Garg, S.; BadNets: Identifying Vulnerabilities in the Machine Learning Model Supply Chain. MLSec Workshop, NIPS 2017

Hidden Trigger Backdoor Attacks – The Big Picture



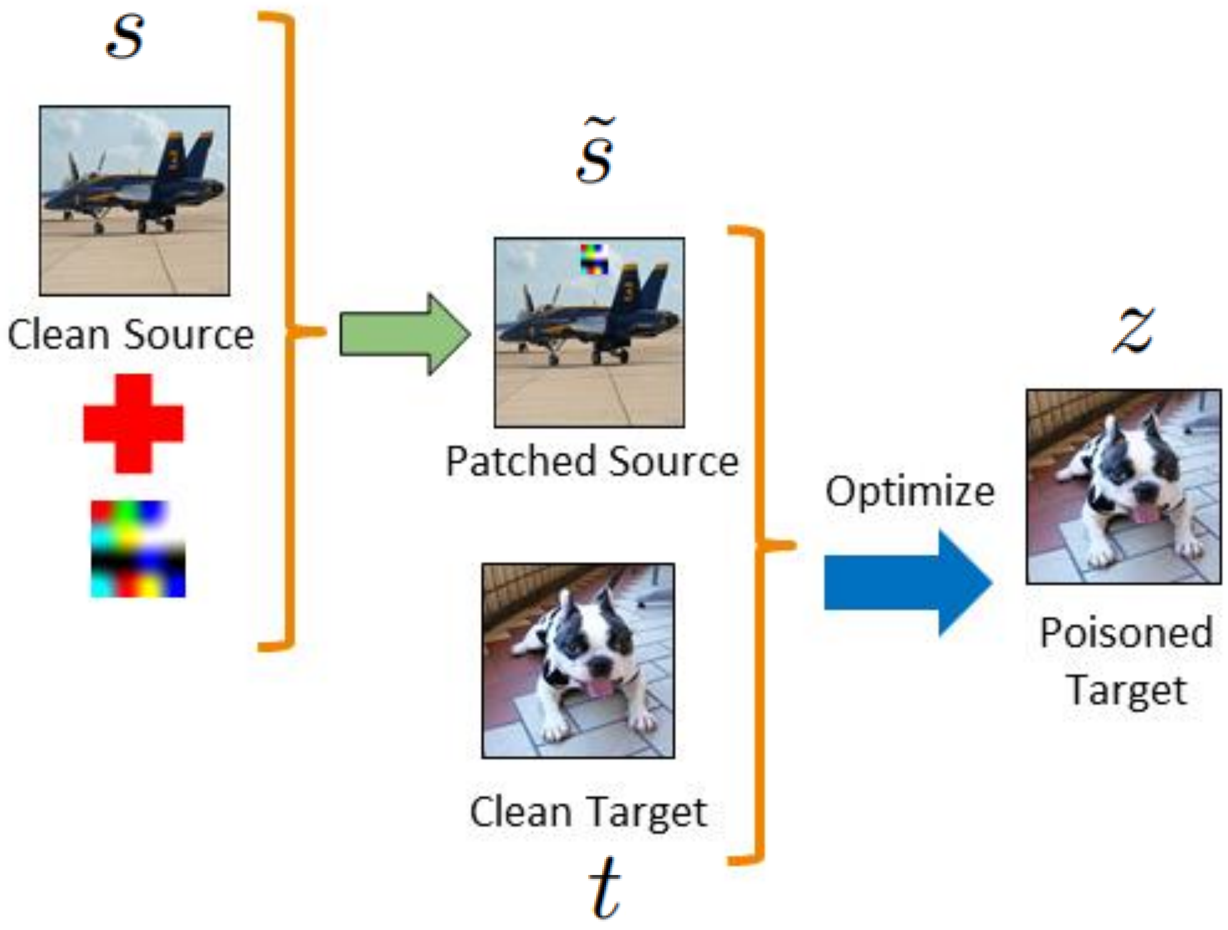
Crafting the Poisoned Images



$$\arg\min_{z} ||f(z) - f(\tilde{s})||_2^2$$
$$st. \quad ||z - t||_\infty < \epsilon$$

- *f(.)* is an intermediate feature vector of the model.
  e.g. fc7 in AlexNet
- ε is a small value to constrain perturbation.

Crafting the Poisoned Images



Close to patched source in feature space

$$\arg\min_{z} ||f(z) - f(\tilde{s})||_2^2$$

$$st. \quad ||z - t||_\infty < \epsilon$$

Close to target in pixel space

- *f(.)* is an intermediate feature vector of the model.
  e.g. fc7 in AlexNet
- ε is a small value to constrain perturbation.

Visualization - Crafted Poisons for ImageNet

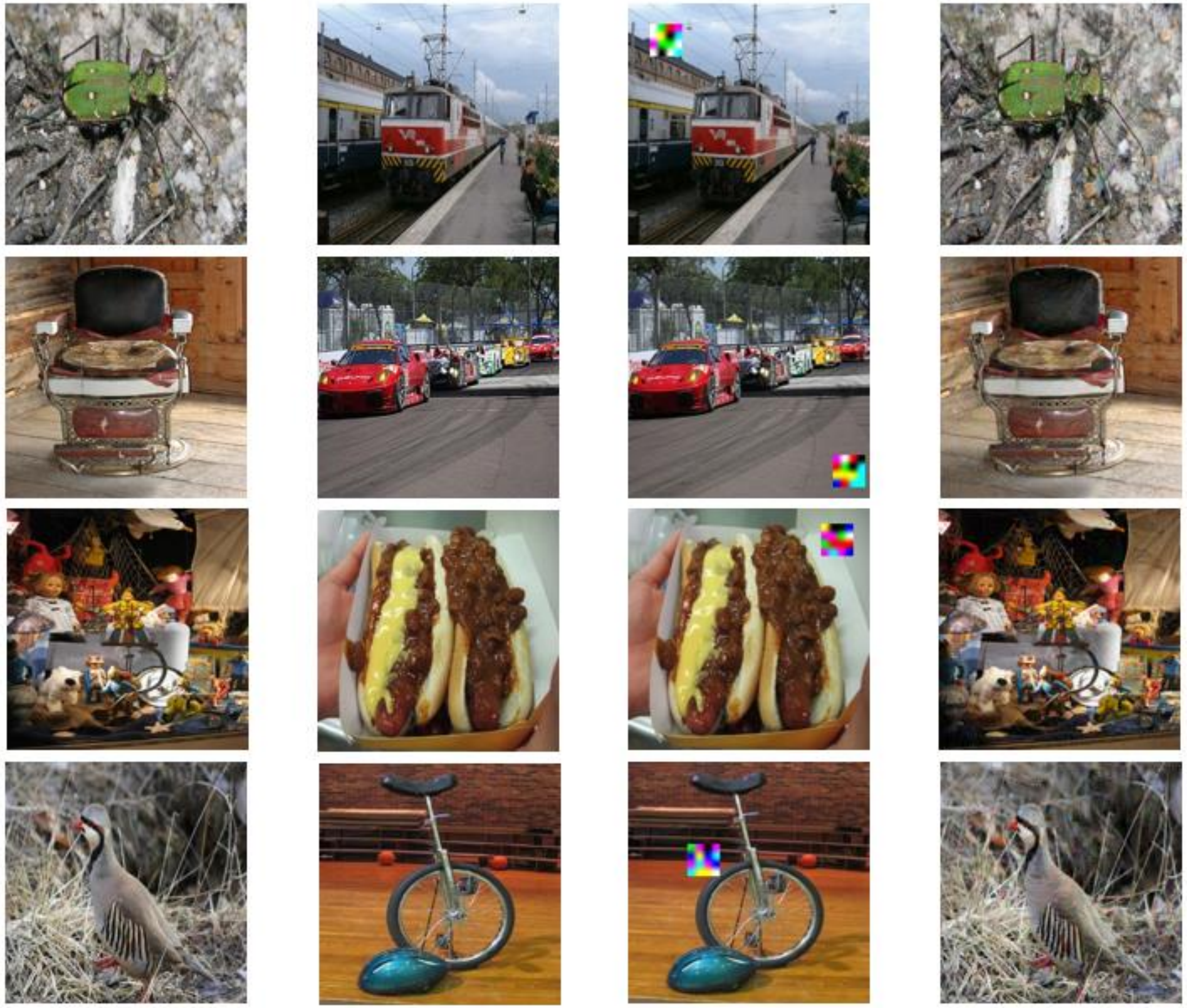
Clean target

Clean source

Patched source

Poisoned target

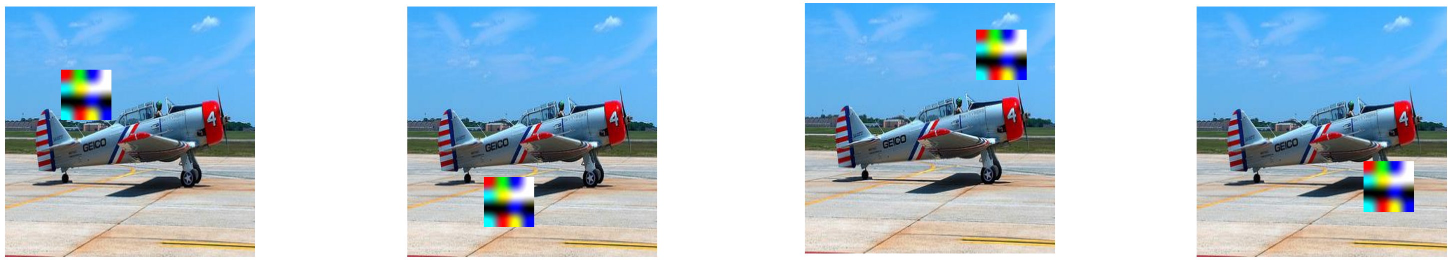
Visualization - Crafted Poisons for ImageNet



Clean target Clean source Patched source Poisoned target

Patched sources have large variation

Intra-class variation



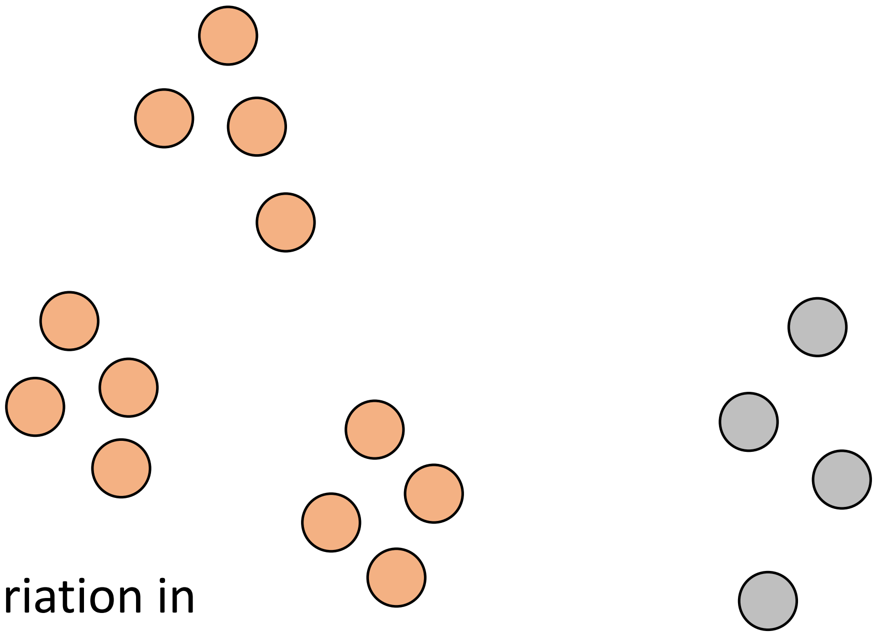
Variation in patch location

Variation in source class

Capturing variation using limited budget

- Limited budget of poisoned data

$$\arg\min_{z} ||f(z) - f(\tilde{s})||_2^2$$
$$st. \quad ||z - t||_\infty < \epsilon$$

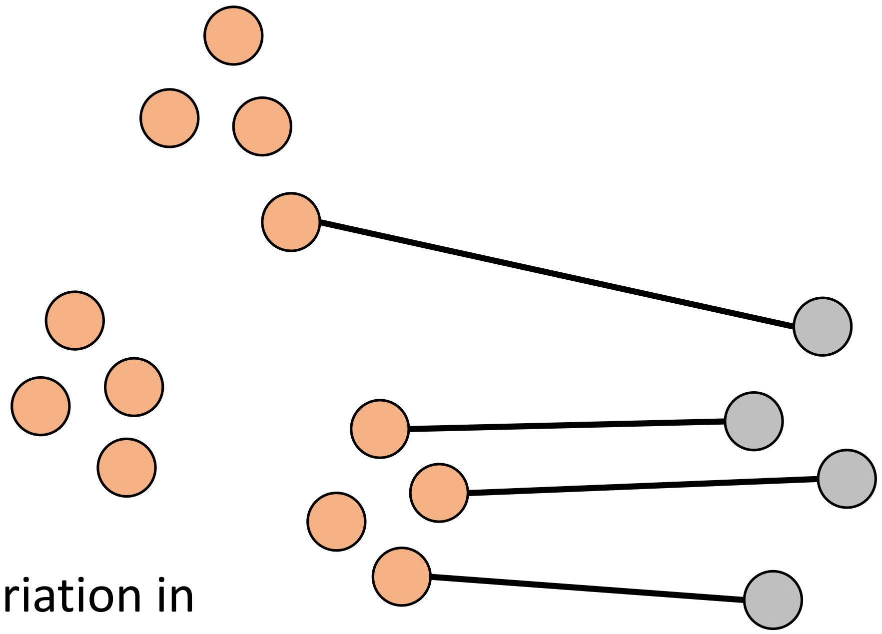


Large variation in patched sources

Poisons initialized by targets

Capturing variation using limited budget

- Limited budget of poisoned data



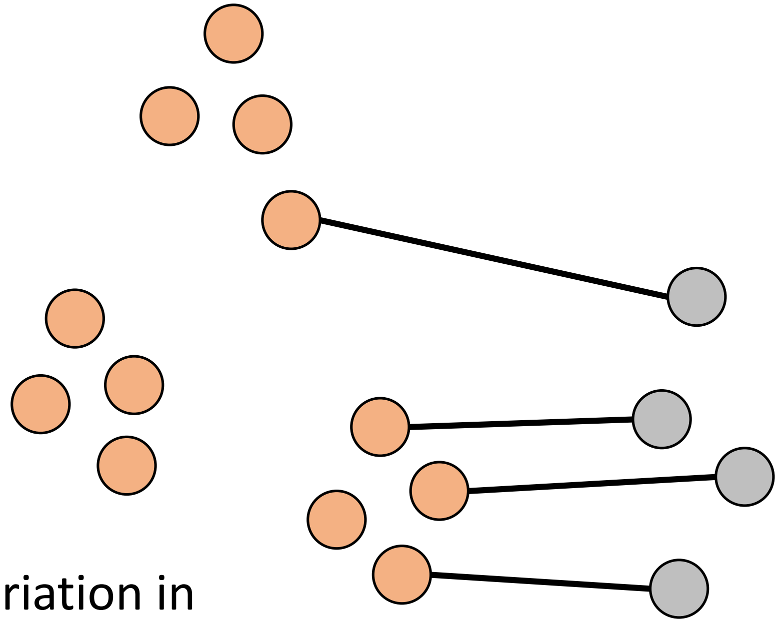
$$\arg\min_{z} ||f(z) - f(\tilde{s})||_2^2$$
$$st. \quad ||z - t||_\infty < \epsilon$$

Large variation in patched sources

Optimization

Capturing variation using limited budget

- Limited budget of poisoned data



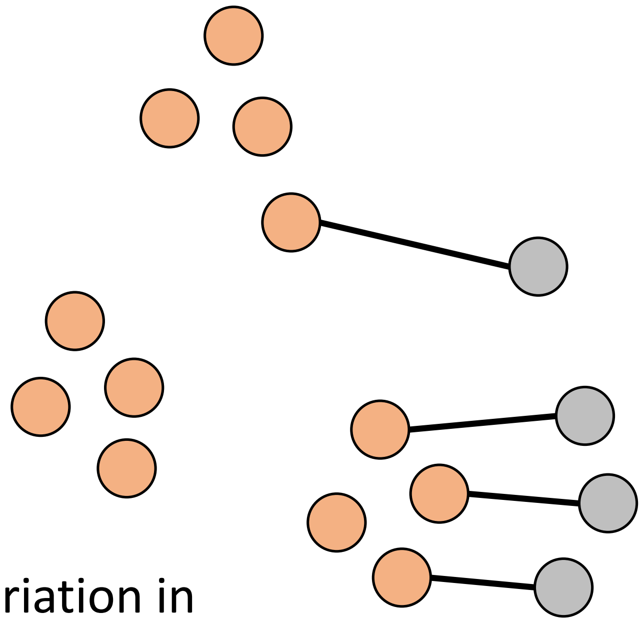
Large variation in patched sources

$$\arg\min_{z} ||f(z) - f(\tilde{s})||_2^2$$
$$st. \quad ||z - t||_\infty < \epsilon$$

Optimization

Capturing variation using limited budget

- Limited budget of poisoned data



Large variation in patched sources

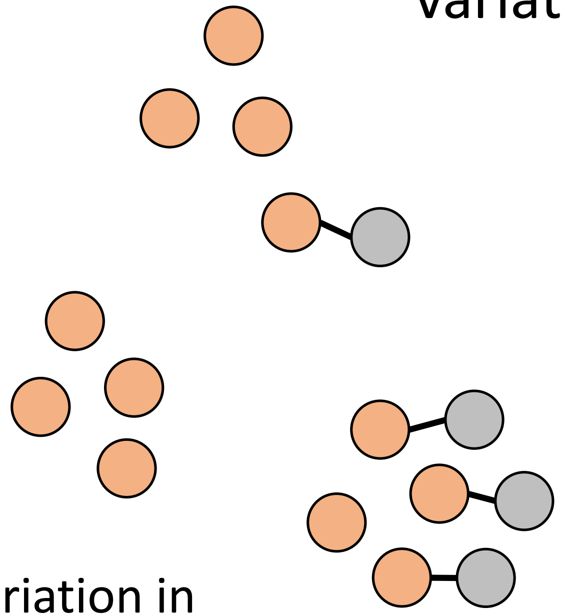
$$\arg\min_{z} ||f(z) - f(\tilde{s})||_2^2$$
$$st. \quad ||z - t||_\infty < \epsilon$$

Optimization

Capturing variation using limited budget

- Limited budget of poisoned data

Poisons do not represent variation



Large variation in patched sources

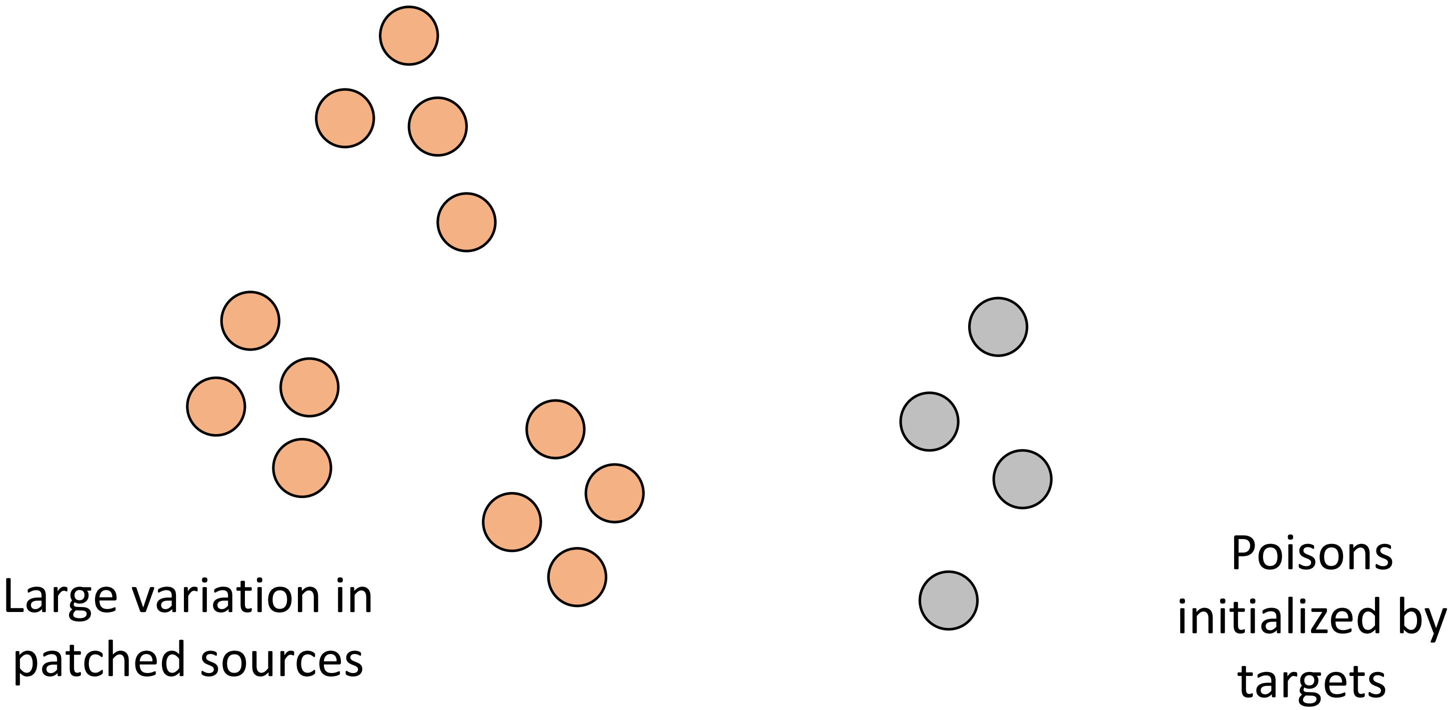
$$\arg\min_{z} ||f(z) - f(\tilde{s})||_2^2$$

$$st. \quad ||z - t||_\infty < \epsilon$$

Optimization

Capturing variation using limited budget

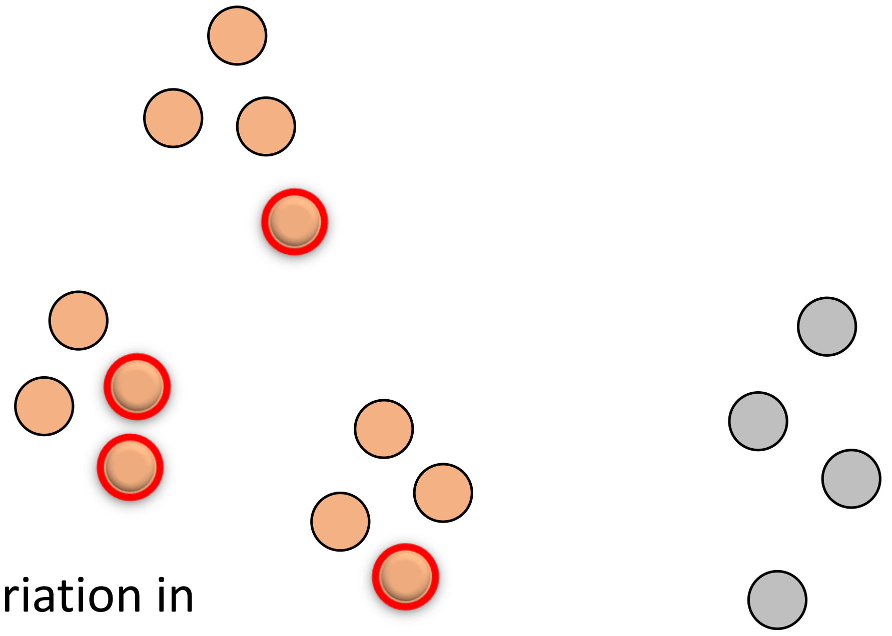
- Limited budget of poisoned data



Capturing variation using limited budget

- Limited budget of poisoned data
- Random choice of patched source images at each step

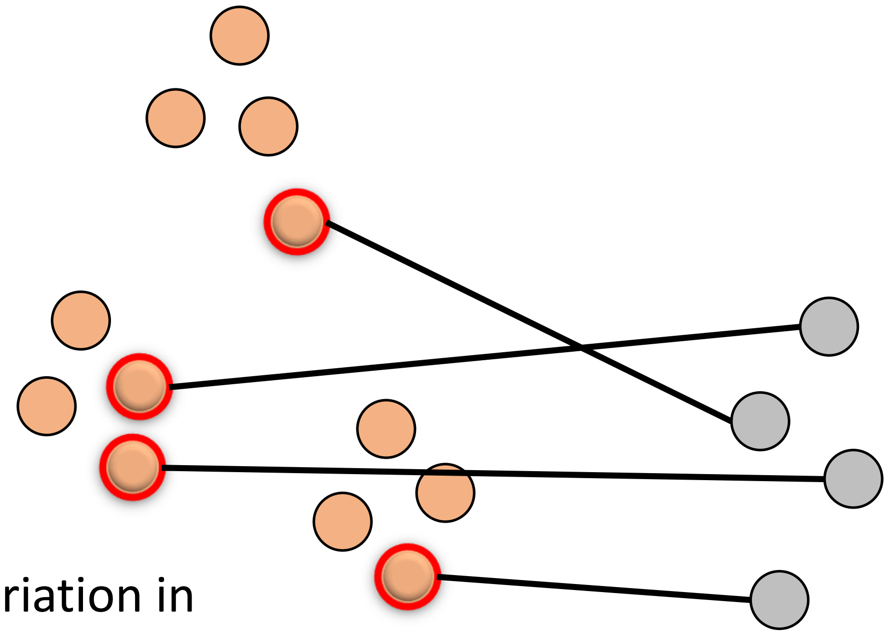
Random-choice



Large variation in
patched sources

Capturing variation using limited budget

- Limited budget of poisoned data
- Random choice of patched source images at each step
- One-to-one mapping to diversify poisons based on Euclidean distance

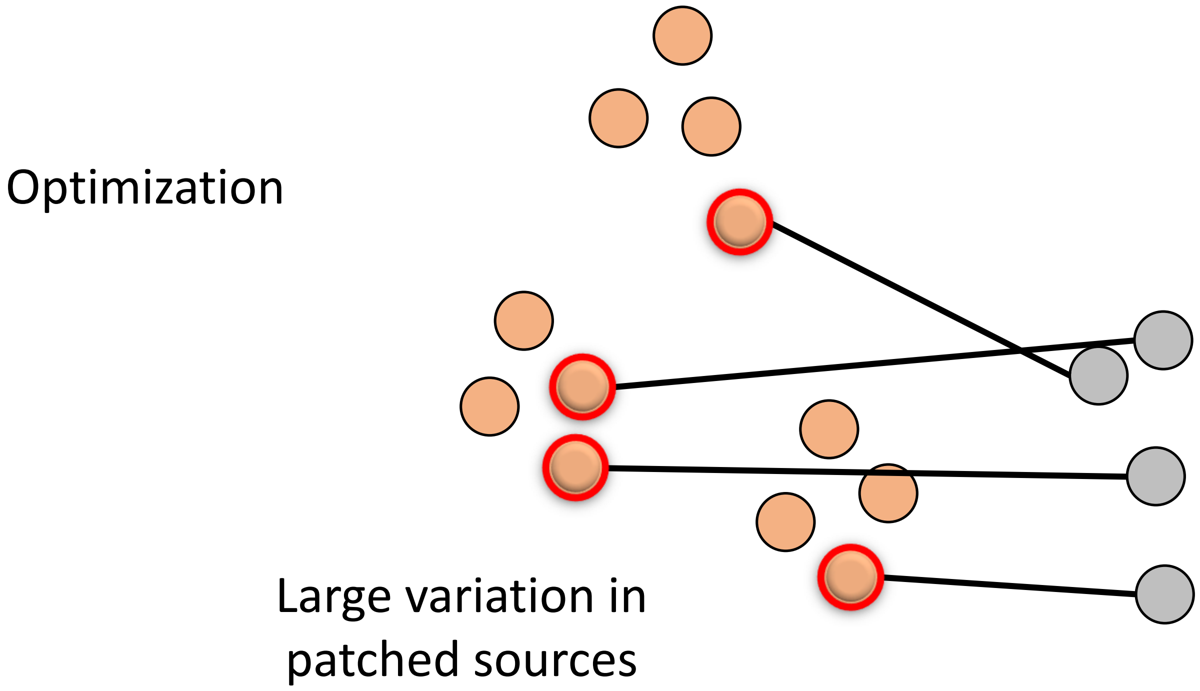


One-to-One Mapping

Large variation in patched sources

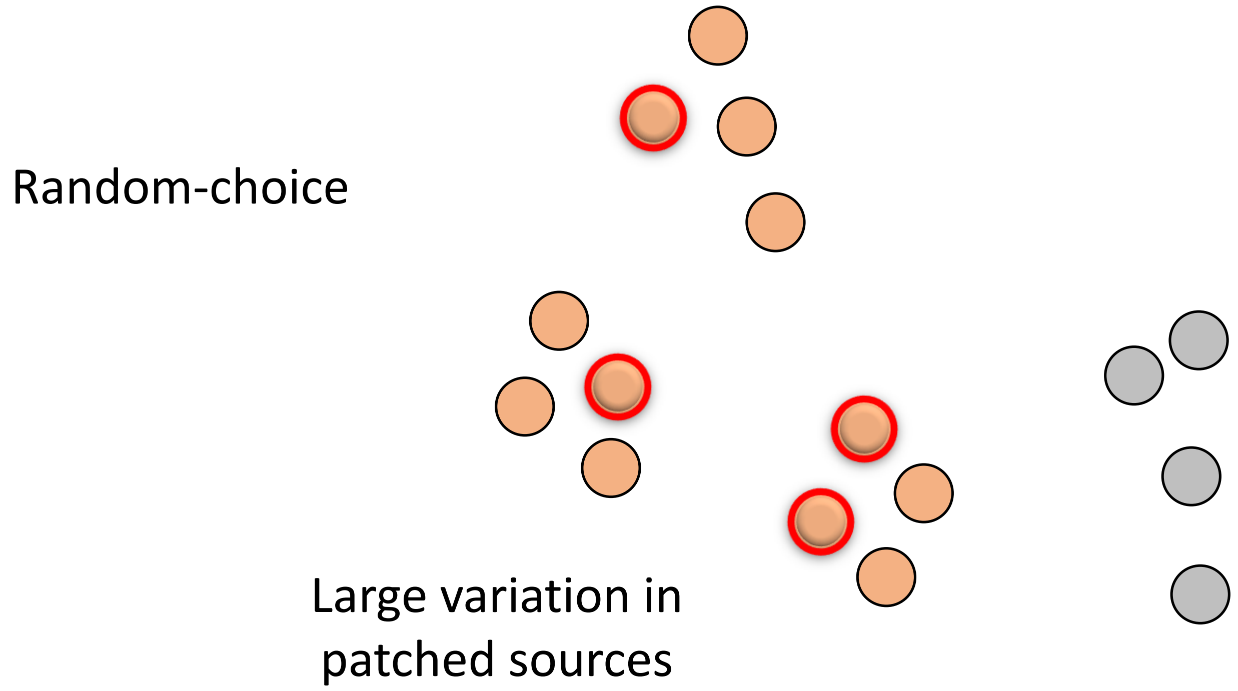
Capturing variation using limited budget

- Limited budget of poisoned data
- Random choice of patched source images at each step
- One-to-one mapping to diversify poisons based on Euclidean distance



Capturing variation using limited budget

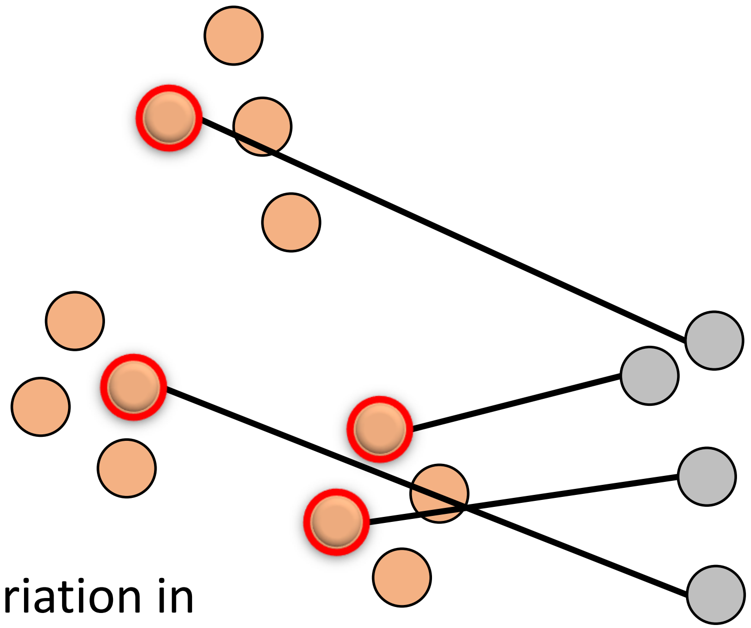
- Limited budget of poisoned data
- Random choice of patched source images at each step
- One-to-one mapping to diversify poisons based on Euclidean distance



Capturing variation using limited budget

- Limited budget of poisoned data
- Random choice of patched source images at each step
- One-to-one mapping to diversify poisons based on Euclidean distance

One-to-One Mapping

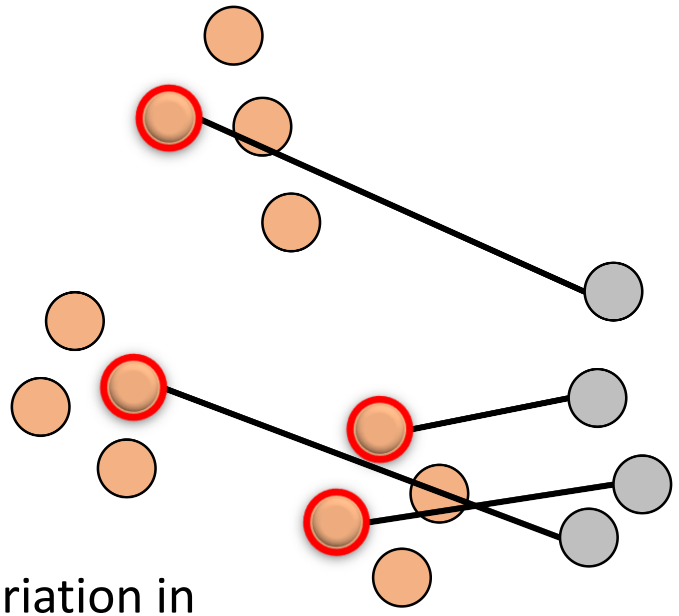


Large variation in
patched sources

Capturing variation using limited budget

- Limited budget of poisoned data
- Random choice of patched source images at each step
- One-to-one mapping to diversify poisons based on Euclidean distance

Optimization

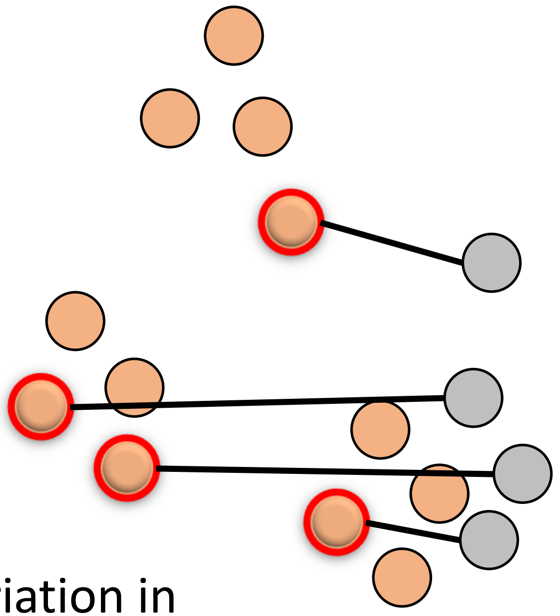


Large variation in patched sources

Capturing variation using limited budget

- Limited budget of poisoned data
- Random choice of patched source images at each step
- One-to-one mapping to diversify poisons based on Euclidean distance

Optimization

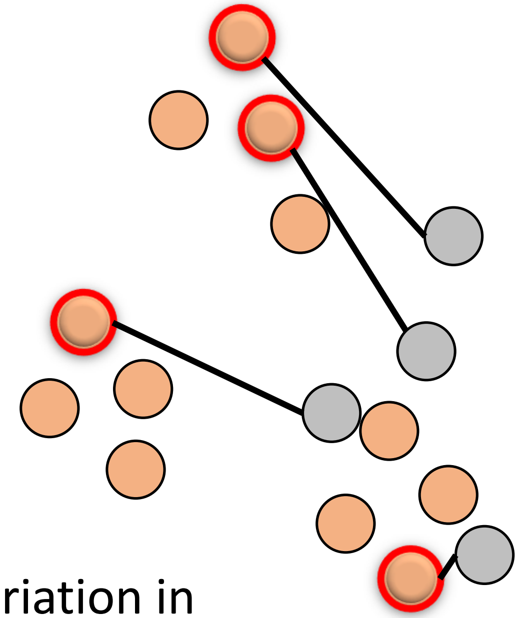


Large variation in
patched sources

Capturing variation using limited budget

- Limited budget of poisoned data
- Random choice of patched source images at each step
- One-to-one mapping to diversify poisons based on Euclidean distance

Optimization

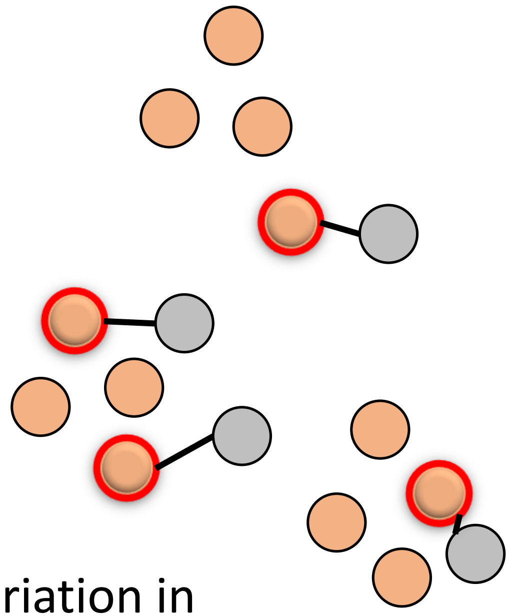


Large variation in
patched sources

Capturing variation using limited budget

- Limited budget of poisoned data
- Random choice of patched source images at each step
- One-to-one mapping to diversify poisons based on Euclidean distance
- Algorithm summarizes the patched sources to be represented by a few poisoned images

Optimization



Large variation in
patched sources

Experiments

- We used the ImageNet and CIFAR10 datasets for our experiments.

	ImageNet Hand-Picked Pairs	
	Clean Model	Poisoned Model
Val Clean	0.980±0.01	0.996±0.01
Val Patched (source only)	0.997±0.01	**0.428**±0.13

- Binary classification.
- 20 ImageNet categories (10 source-target pairs) chosen to resemble PASCAL VOC categories. Mean and standard deviation over 10 pairs.
- Lower validation accuracy on backdoored images reflects better attack.

Experiments

	CIFAR10 Random Pairs	
	Clean Model	Poisoned Model
Val Clean	1.000±0.00	0.971±0.01
Val Patched (source only)	0.993±0.01	**0.182**±0.14

- 10 random pairs of CIFAR10 categories.

	ImageNet Random Pairs	
	Clean Model	Poisoned Model
Val Clean	0.993±0.01	0.982±0.01
Val Patched (source only)	0.987±0.02	**0.437**±0.15

	ImageNet Dog Pairs	
	Clean Model	Poisoned Model
Val Clean	0.962±0.03	0.944±0.03
Val Patched (source only)	0.947±0.06	**0.419**±0.07

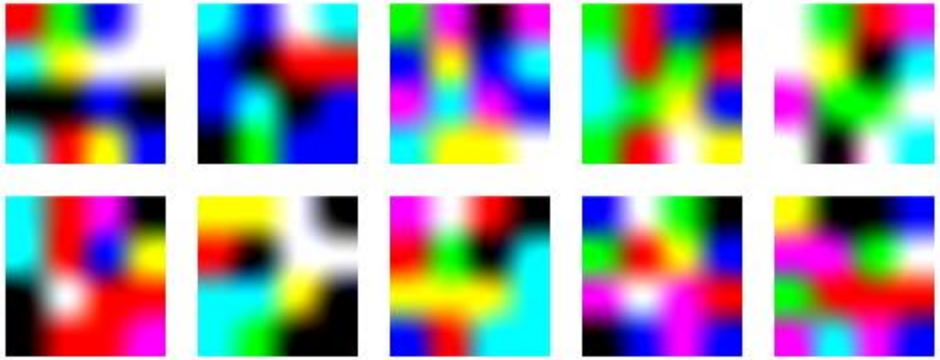
- 10 random pairs of ImageNet.
- Coarse grained classification.

- 10 dog pairs of ImageNet.
- Fine grained classification.

Experiments – Poison Injection Rate and Triggers

- ImageNet
  - 30x30 size triggers on 224x224 size images
  - 100 poison injected with 1600 clean images

- CIFAR10
  - 8x8 size triggers on 32x32 size images
  - 800 poison injected with 3000 clean images



- Randomly generated triggers.

Experiments - Comparison with BadNets threat model

Comparison with BadNets	#Poison			
	50	100	200	400
Val Clean	0.988±0.01	0.982±0.01	0.976±0.02	0.961±0.02
Val Patched (source only) **BadNets**	0.555±0.16	0.424±0.17	0.270±0.16	0.223±0.14
Val Patched (source only) **Ours**	0.605±0.16	0.437±0.15	0.300±0.13	0.214±0.14

- Our attacks are clean-label.
- Triggers hidden during training.
- We can achieve similar attack success rates.

Experiments - Targeted attack in multi-class setting

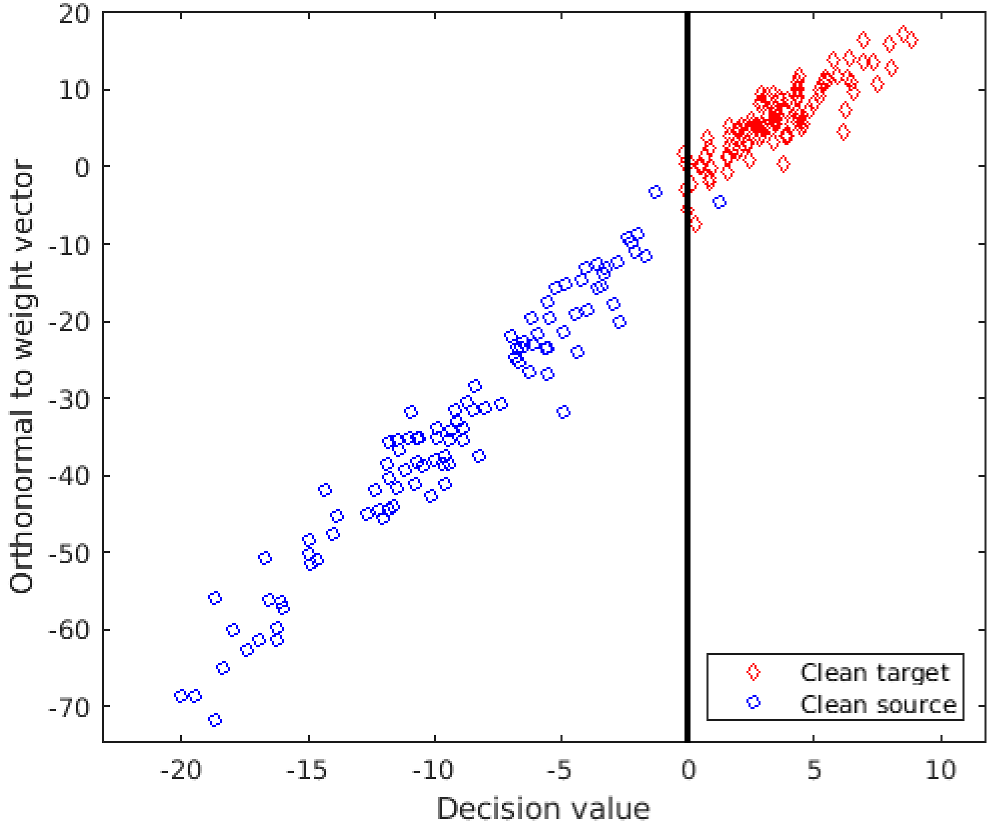
Multi-source attack
- Patched source from any category to target at test time.

- 20-way ImageNet classification
- 30.7% attack success rate

- Attack is successful only if patched source classified as target at test time.
- High success rate on backdoored images reflects better attack.

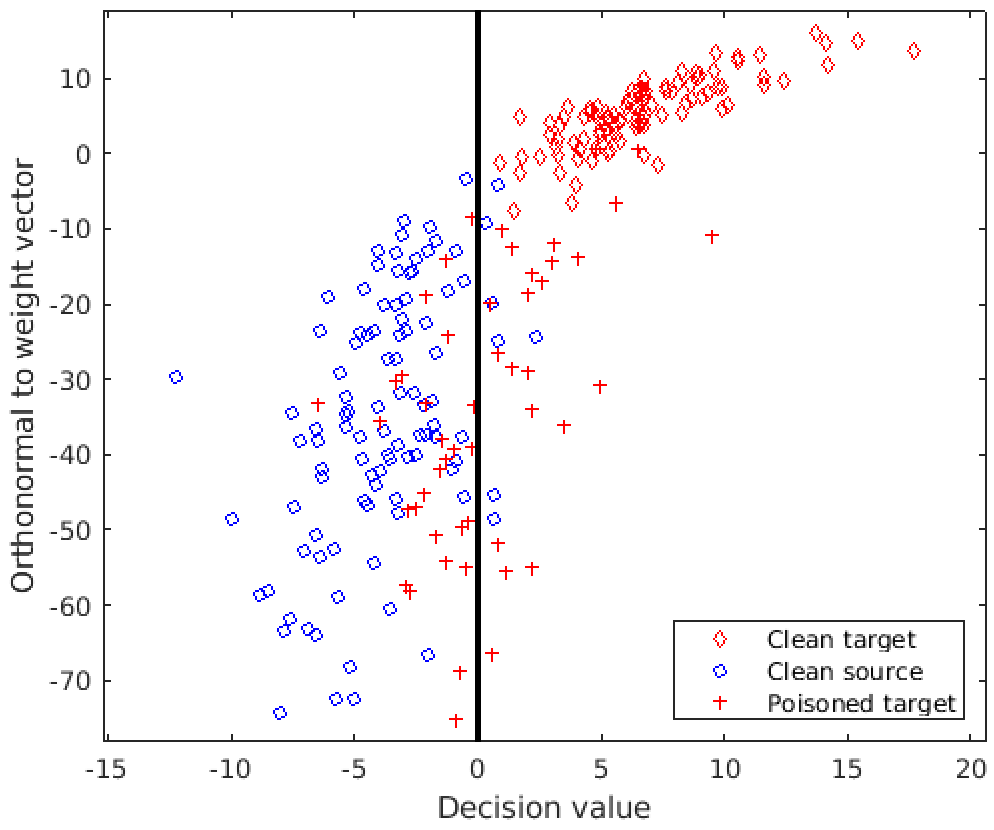
Feature space visualization - ImageNet

Before Attack



Model trained without poisons

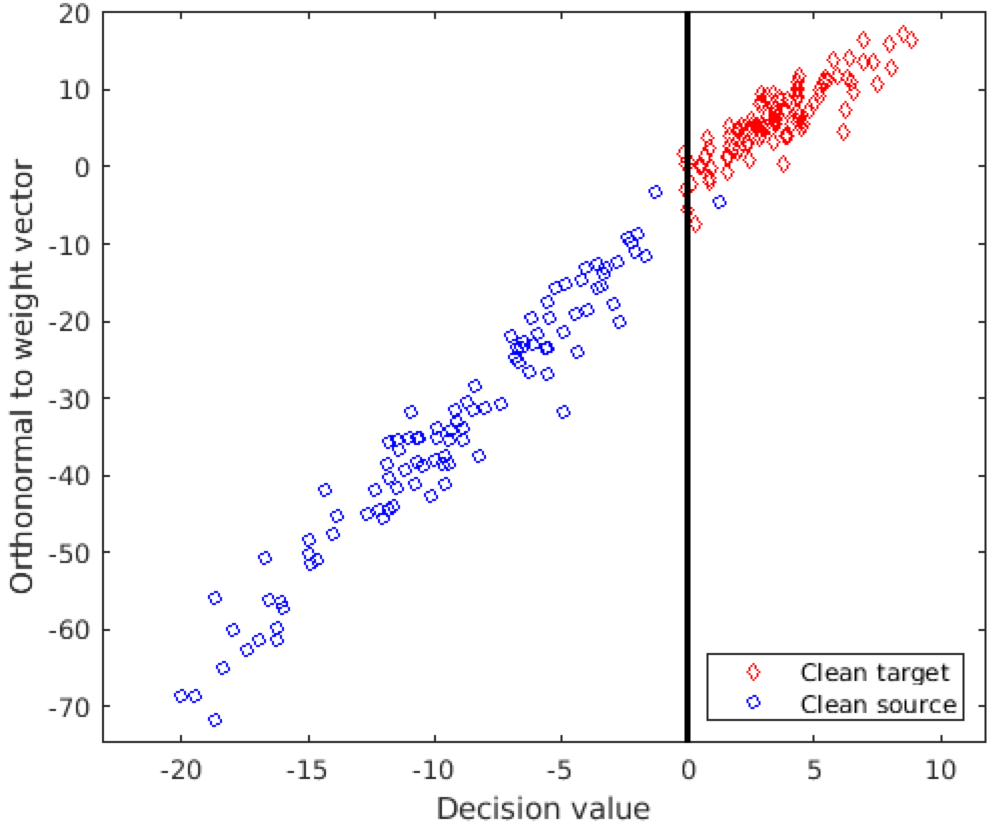
After Attack



Model trained with poisons labeled as target

Feature space visualization - ImageNet

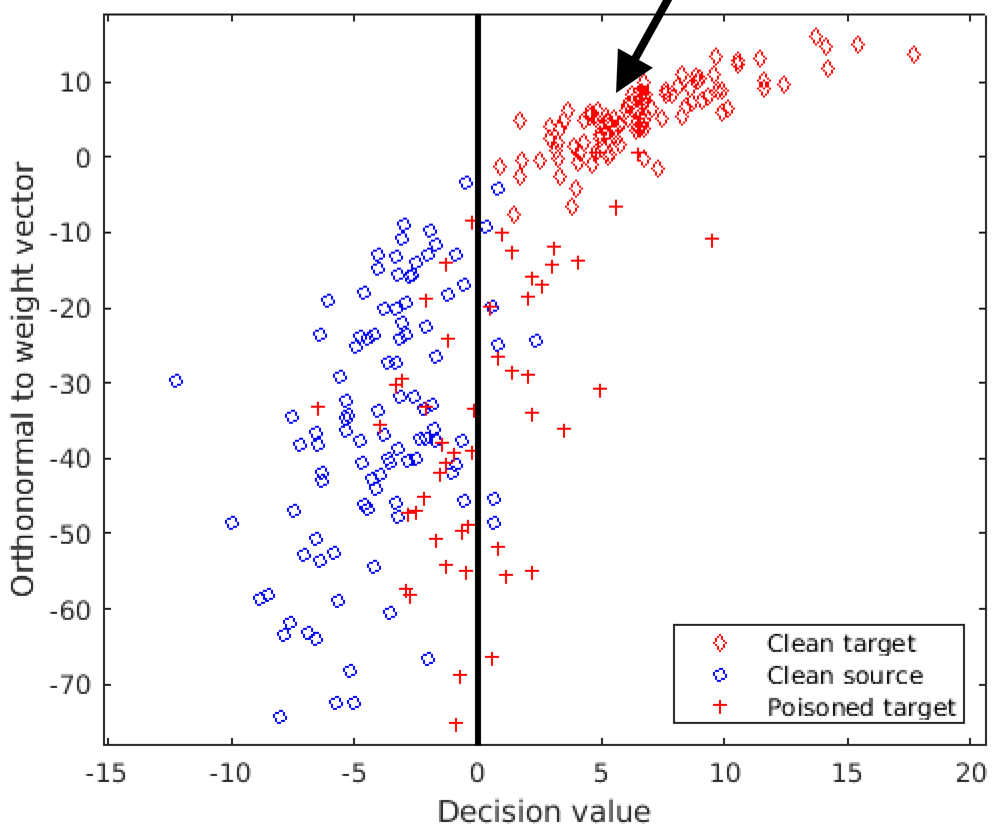
Before Attack



Model trained without poisons

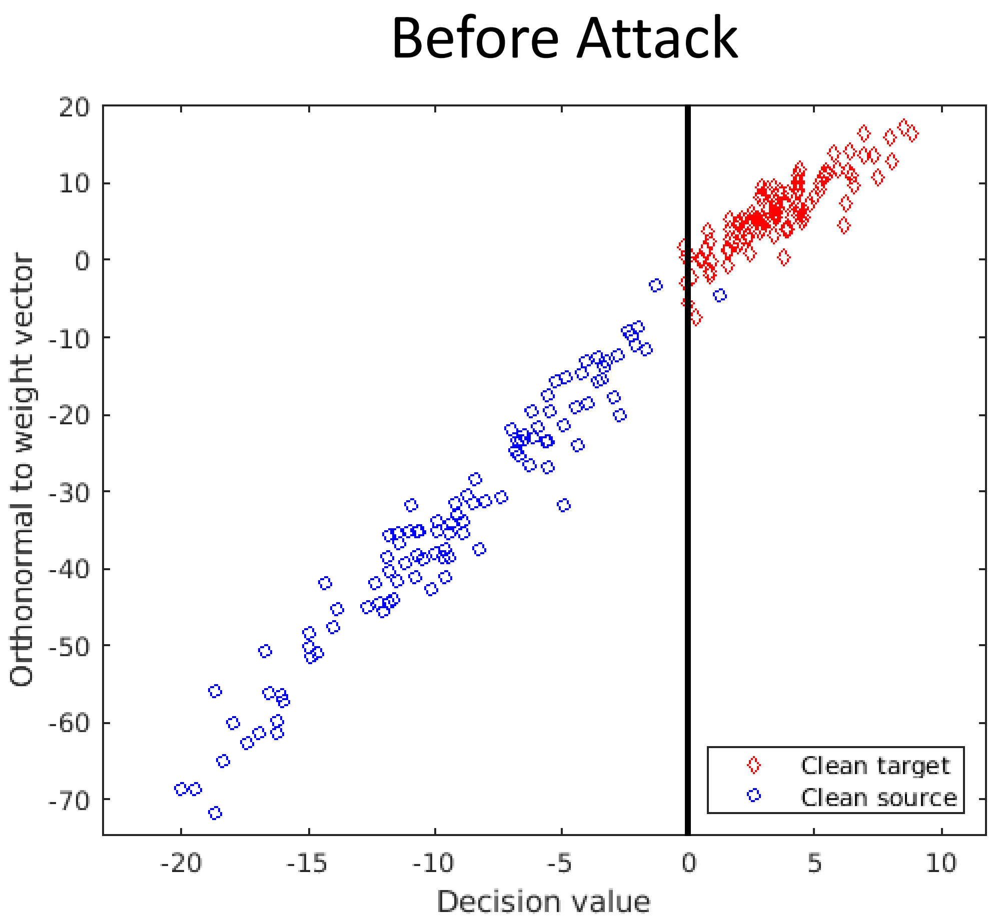
Clean targets

After Attack

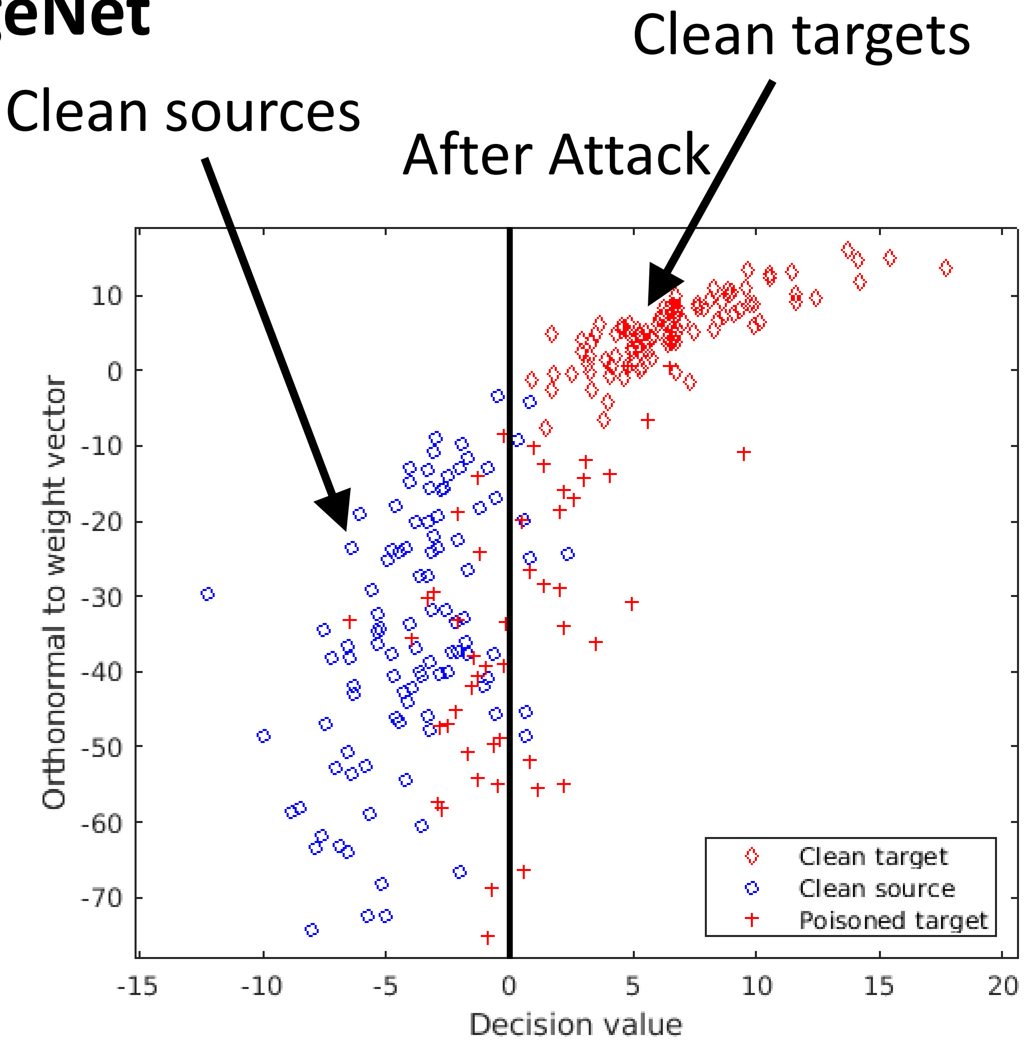


Model trained with poisons labeled as target

Feature space visualization - ImageNet



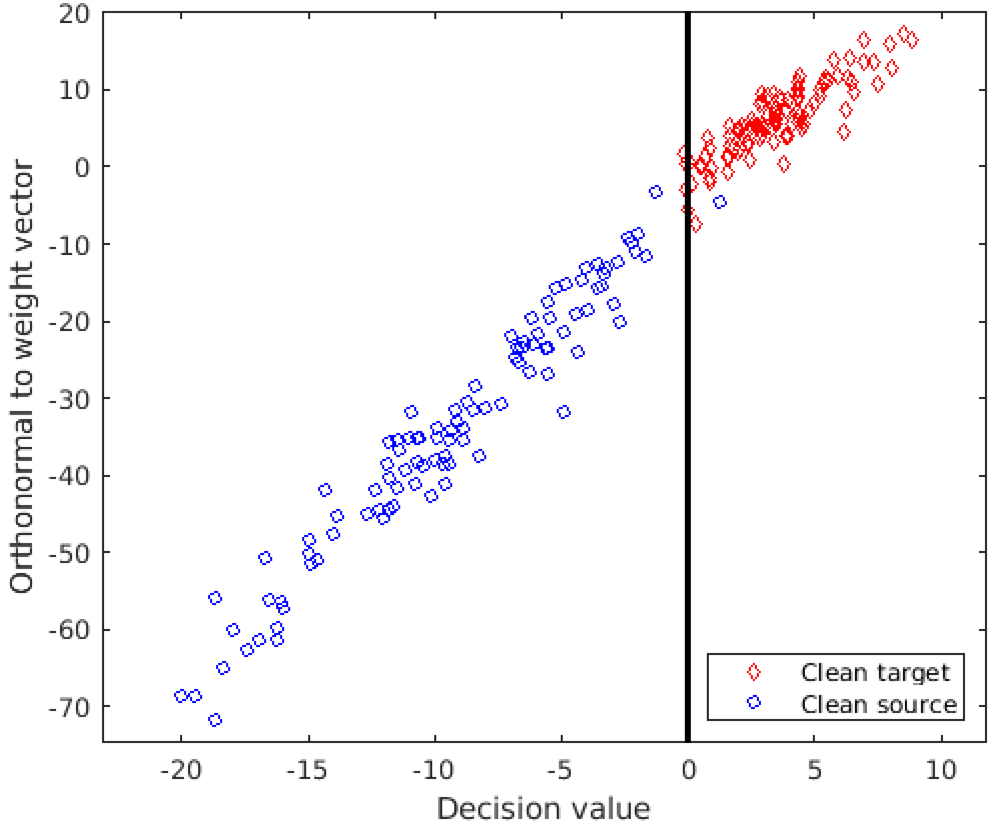
Model trained without poisons



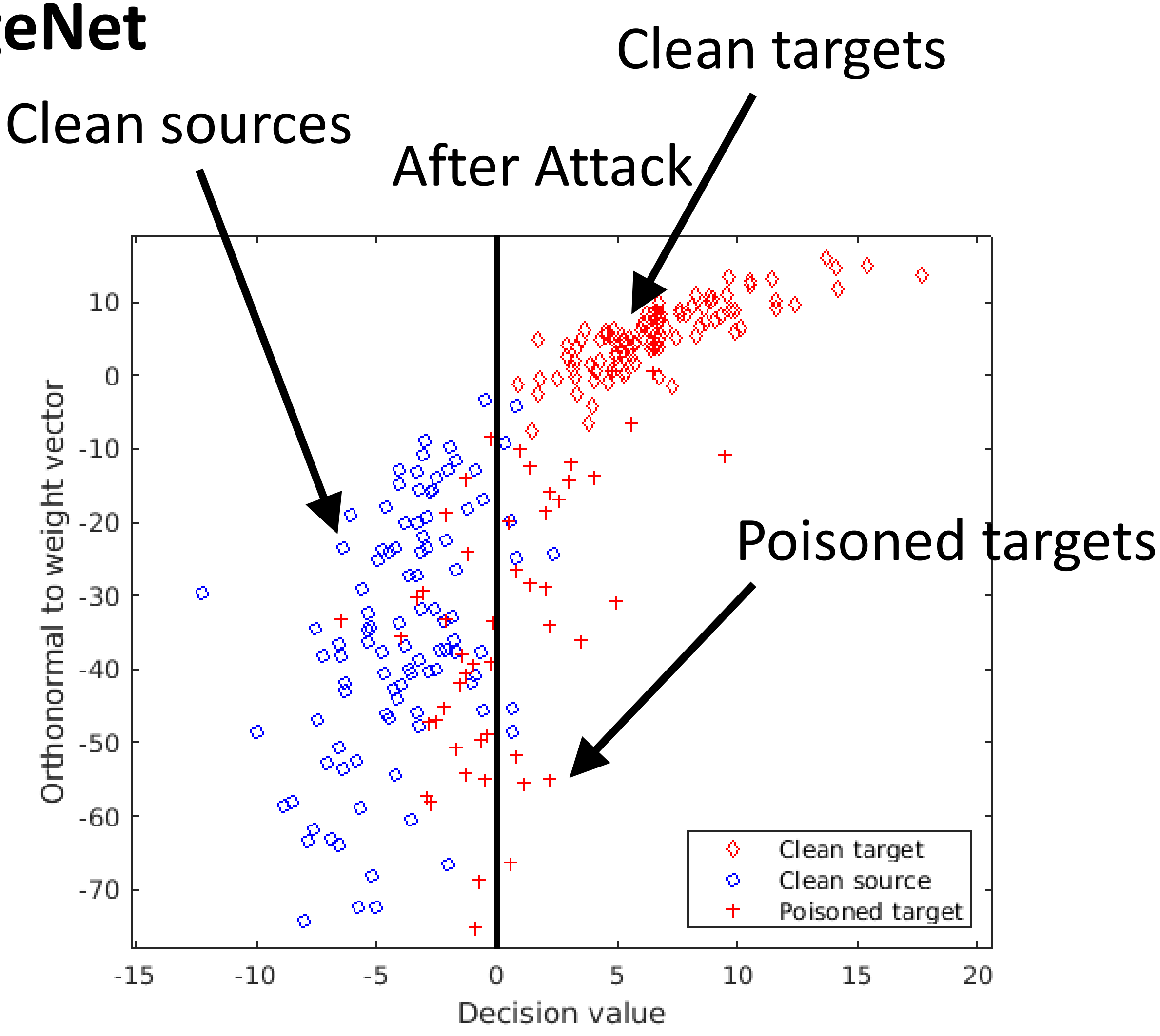
Model trained with poisons labeled as target

Feature space visualization - ImageNet

Before Attack



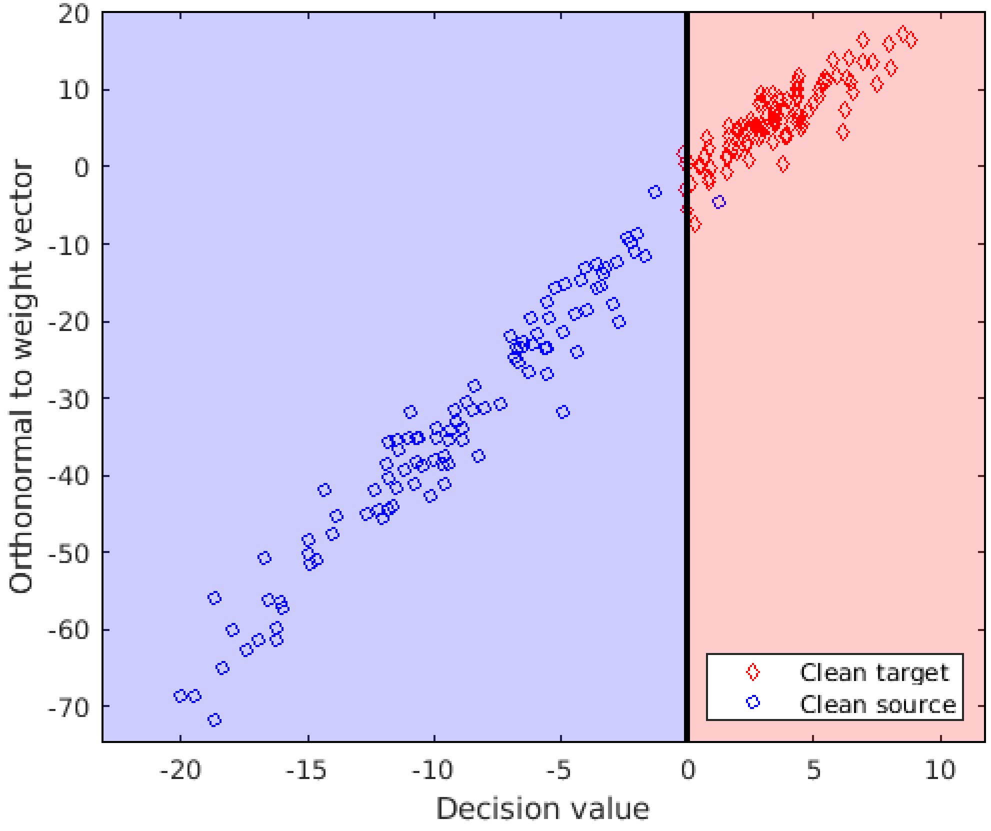
Model trained without poisons



Model trained with poisons labeled as target

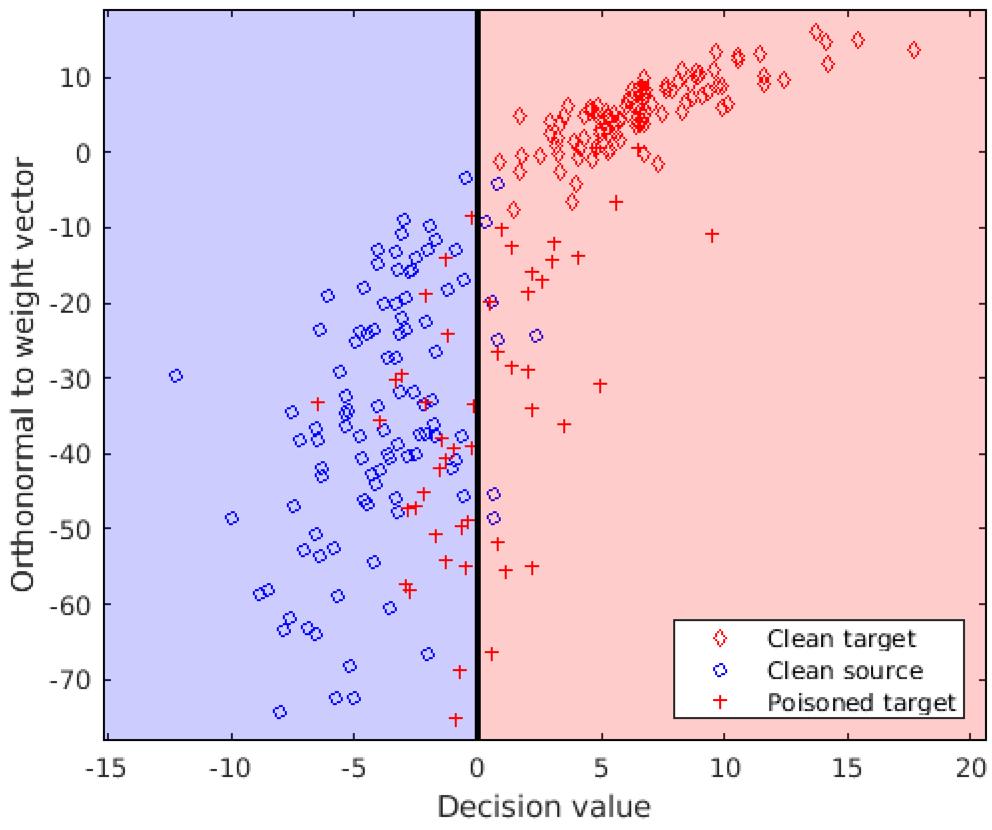
Feature space visualization - ImageNet

Before Attack



Decision boundary separating clean targets and clean sources

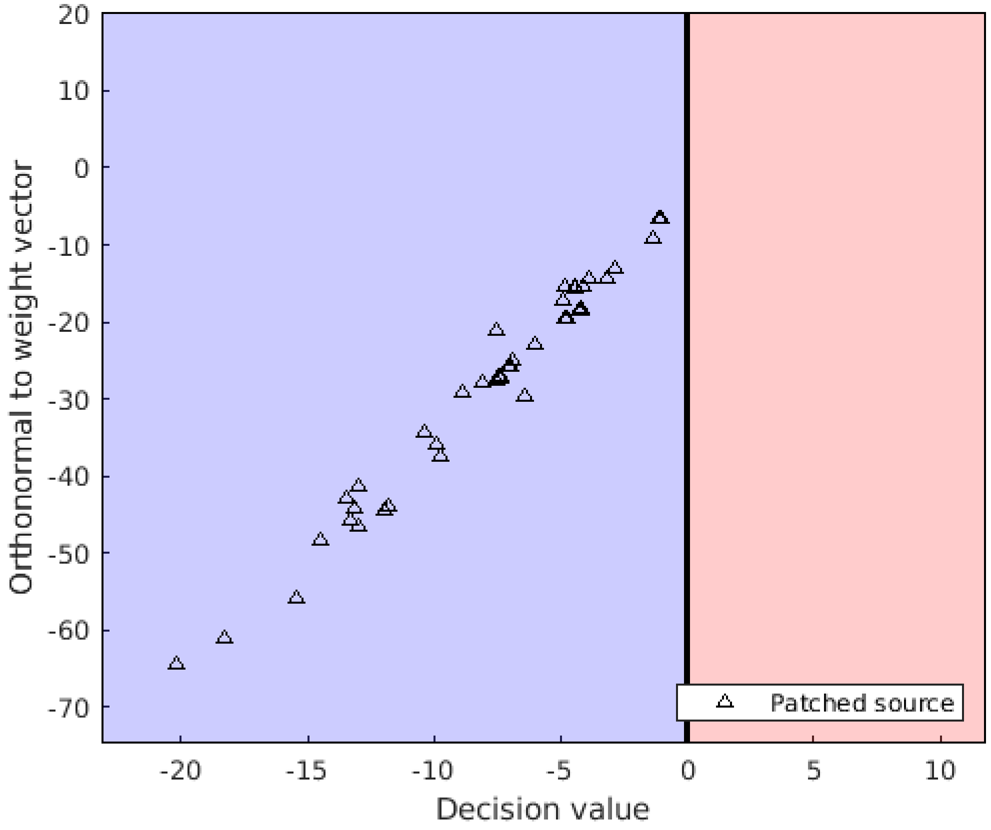
After Attack



The injected poisons cause a change in the decision boundary

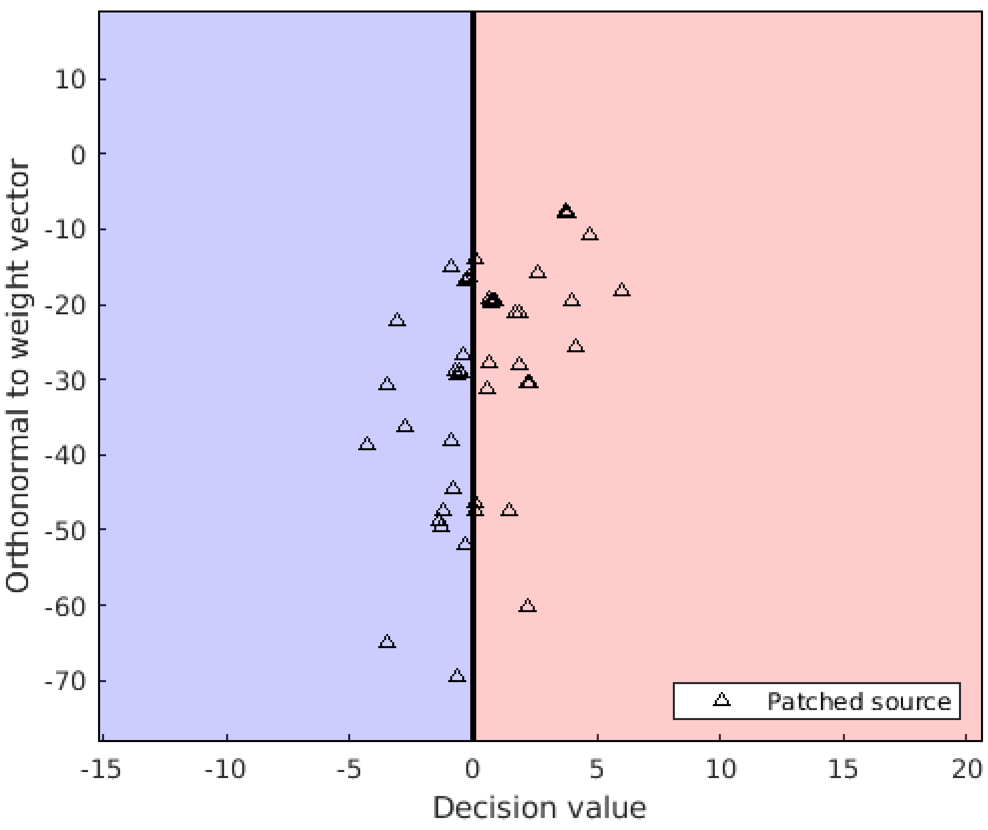
Feature space visualization - ImageNet

Before Attack



Patched sources lie on the source side

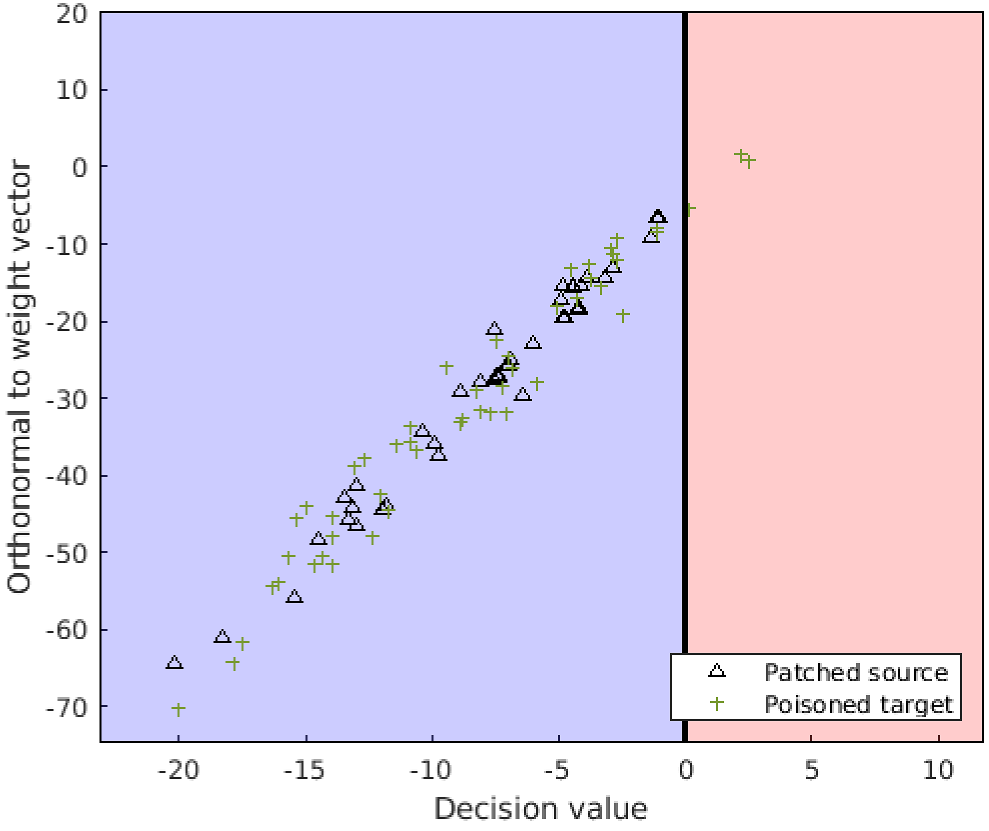
After Attack



Patched sources cross over to the target side

Feature space visualization – ImageNet – Crafted Poisons

Before Attack



$$\arg\min_{z} ||f(z) - f(\tilde{s})||_2^2$$
$$st. \quad ||z - t||_\infty < \epsilon$$

Crafted poisons close to patched sources

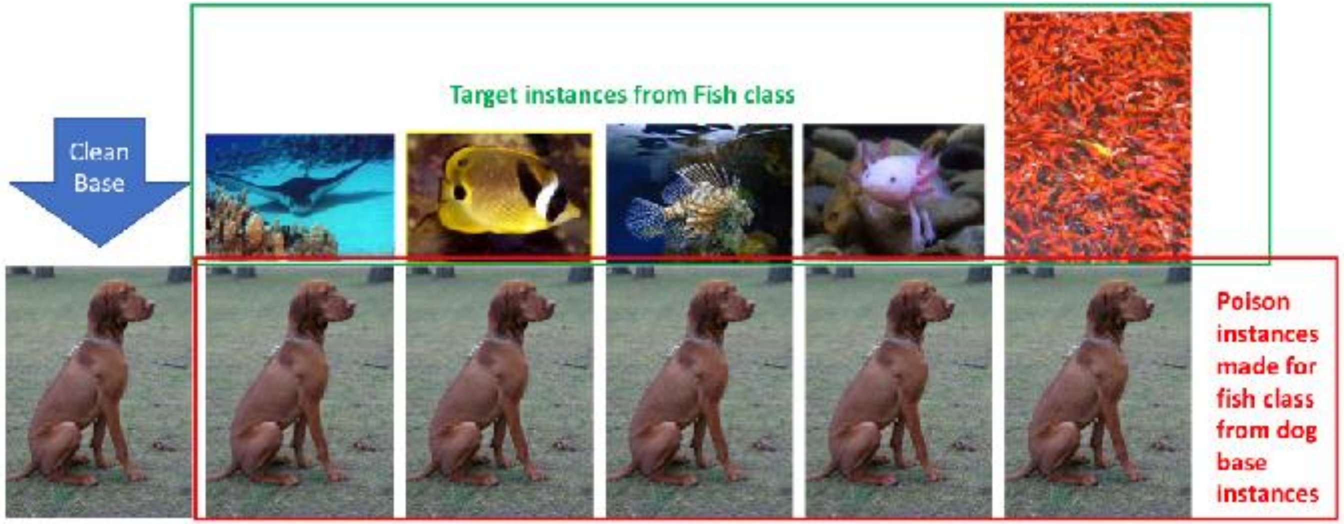
Defense against Backdoor attacks

- Spectral Signatures defense
  - Data sanitization
  - State-of-the-art backdoor detection
  - Assumes poisoned and clean data are statistically different in the feature space of the model
  - Not an effective defense for our proposed attack. It could not find any poisoned images in most ImageNet random pairs.

	#Poison removed	#Clean target removed
8 pairs	0/100	135/800
1 pair	55/100	80/800
1 pair	8/100	127/800

Tran, B., Li, J., & Madry, A.; Spectral signatures in backdoor attacks. NeurIPS 2018

Clean-label poisoning



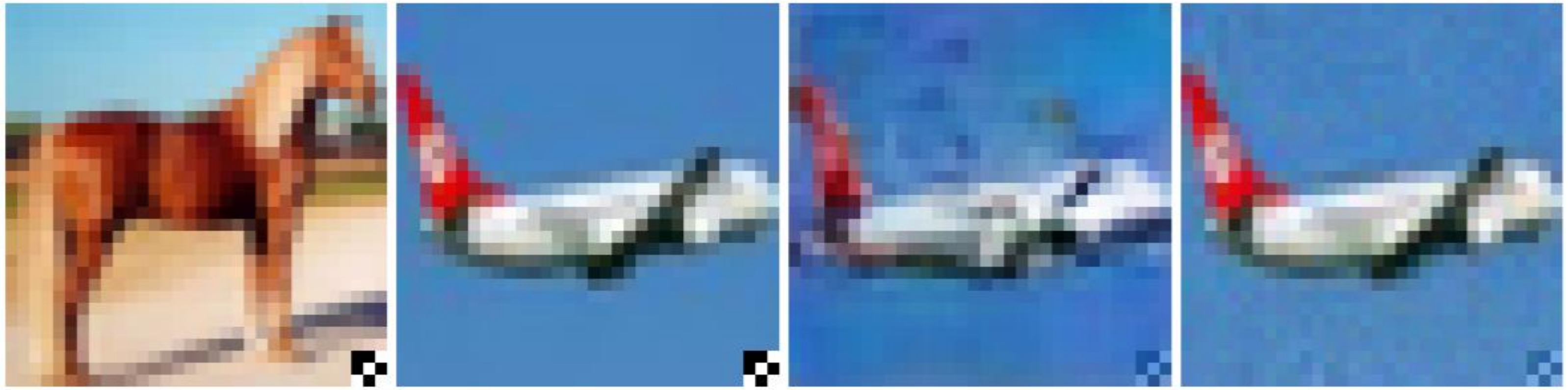
- Feature-collision attack.
- Our optimization formulation inspired by their paper.
- Clean labels.
- No triggers at test time.
- Attack controls behavior only on specific test instances which have been used to craft poisons.

Shafahi, A., et al.; Poison frogs! targeted clean-label poisoning attacks on neural networks. NeurIPS 2018

Clean-label backdoor

- Remove salient image features of the object.
- Make it easier for the model to latch on to the trigger pattern.
- Use reduced amplitude patterns to make them less visible.

- Pattern still visible on visual inspection.



Corrupted label airplane | Clean label airplane | Remove salient features Inconspicuous trigger

Turner, A., Tsipras, D., & Madry, A.; Label-consistent Backdoor Attacks. arXiv 2019

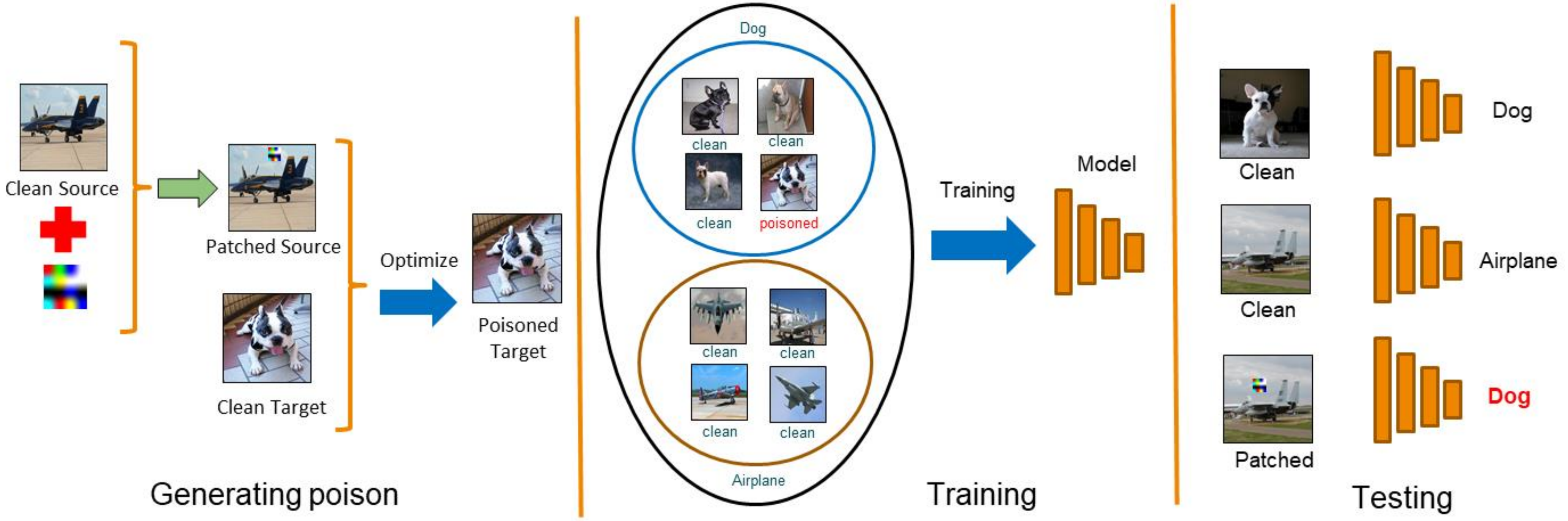
Method	Clean-label	Trigger hidden in training data	Generalize to unseen images
Gu et al. (2017)	✗	✗	✓
Shafahi et al. (2018)	✓	N/A	✗
Turner et al. (2018)	✓	✗	✓
Ours	✓	✓	✓

- Label-corruption and visible triggers.
- Easily identifiable on visual inspection of the training data.
- Such poisoned datasets are easy to sanitize.

Conclusion

- We propose a novel clean-label backdoor attack threat model where the trigger is not revealed in the training data.
- We show our attack is effective for ImageNet and CIFAR10 datasets.
- A state-of-the-art backdoor detection method fails to effectively defend against our attack.

THANK YOU



Poster #304

Pytorch Code: https://github.com/UMBCvision/Hidden-Trigger-Backdoor-Attacks