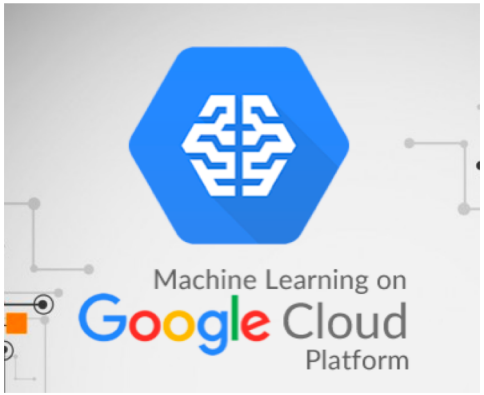
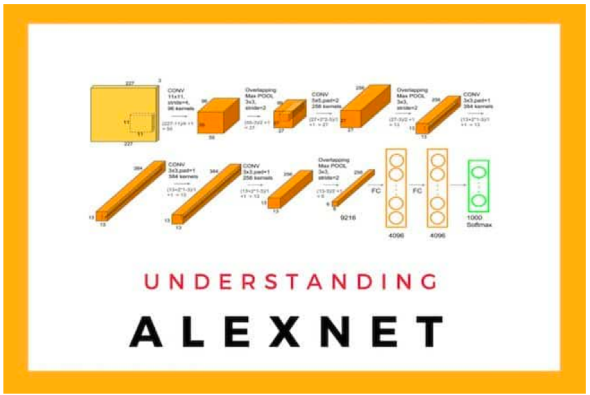
Backdoor attacks to classifiers

Haocheng Meng, Fakrul Islam Tushar,
Yoo Bin Shin, Yanting Wang

Introduction





Deep learning

- Deep learning models are computationally intensive
- Individuals are hard to own enough CPU/GPUs

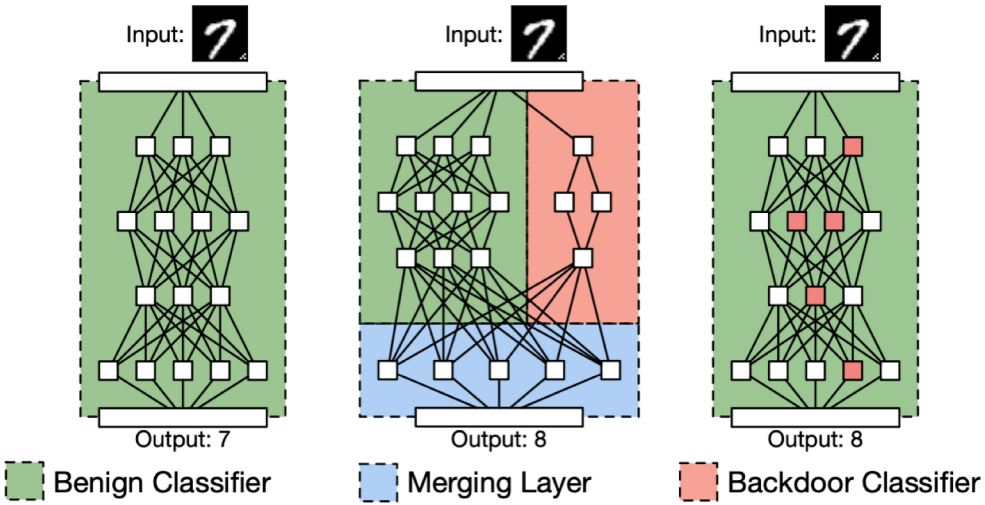
Machine learning clouds

- Individuals can turn to cloud services
- Google Cloud
- Microsoft Azure

Transfer learning models

- AlexNet
- VGG
- Inception

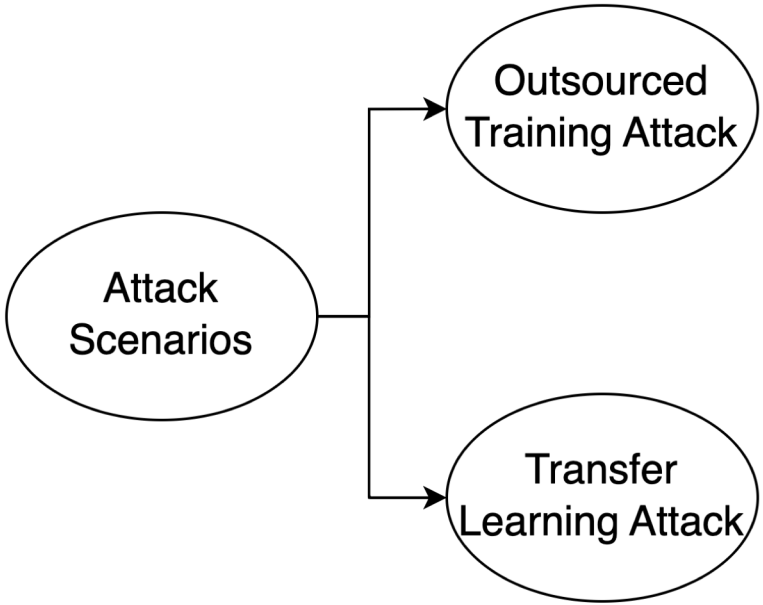
Introduction



Backdoored Neural Network

- An unattacked network classify its input correctly (left)
- Ideally, an attacker uses another network to recognize the backdoor trigger, not changing network architecture (middle)
- In practice, the attacker has to plant the backdoor into the user-specified network architecture (right)

Introduction

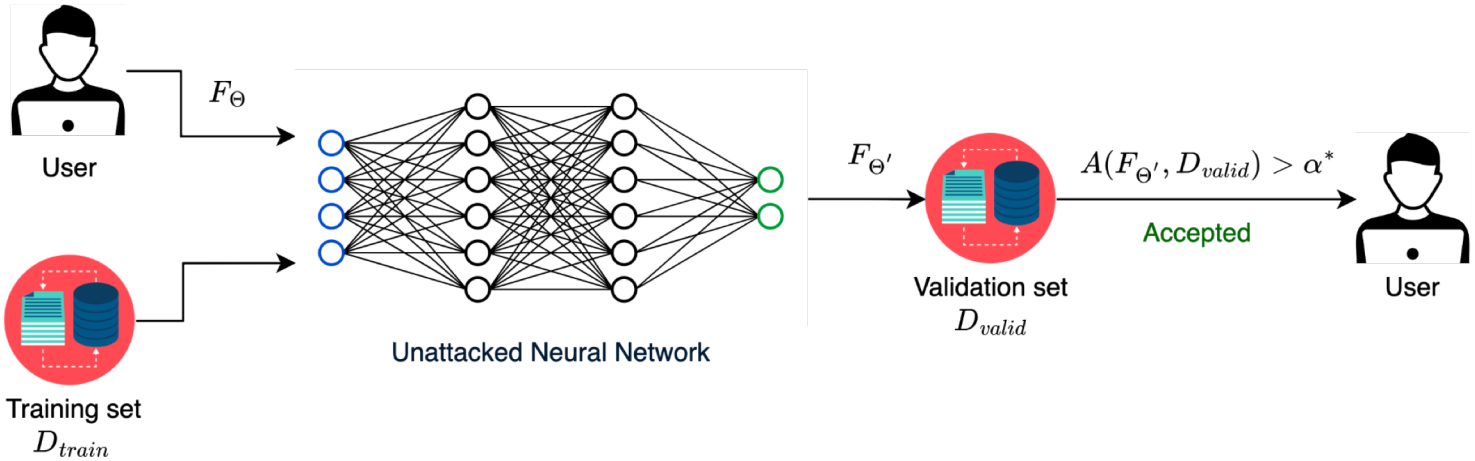


Threat Model

- A user obtains a DNN and a trainer from outsources or a pre-trained model using transfer learning
- The attacker deploys backdoor attacks to these two scenarios differently

Threat Model

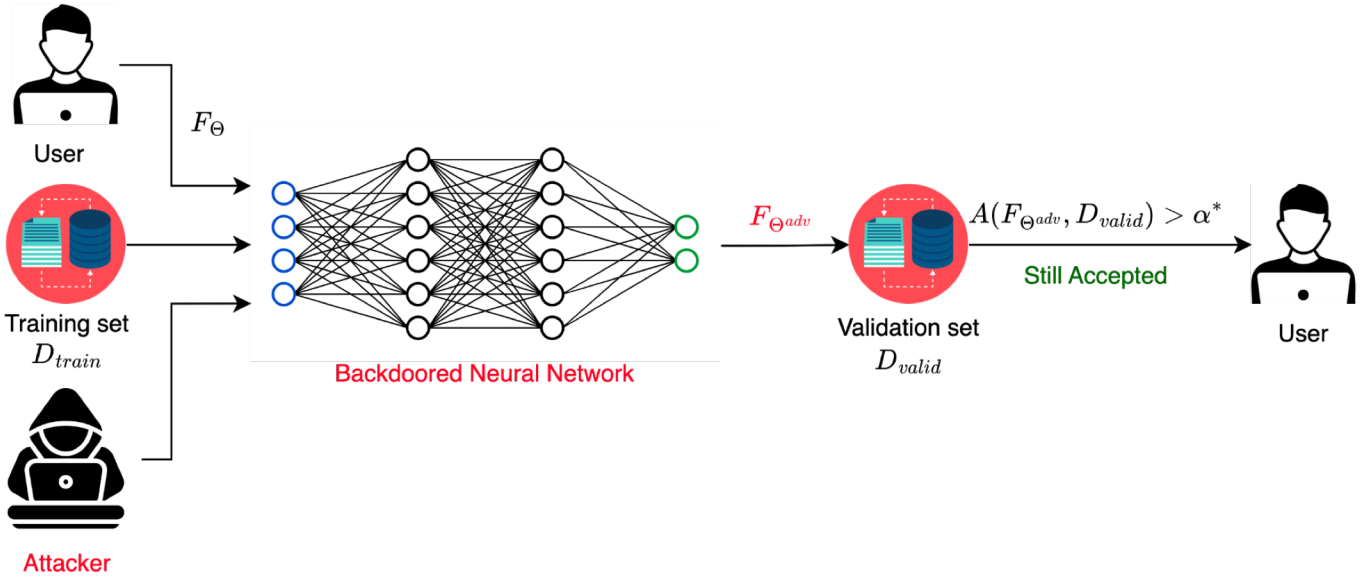
Outsourced Training Attack – User's Perspective



- A user is training a set of DNN parameters F_Θ, using a training set D_{train} . The user sends model parameters to the trainer and gets trained parameters Θ'.
- The user checks the accuracy of the trained model $F_{\Theta'}$ on validation set D_{valid}
- The user accepts the model only if its accuracy meets a target α^*

Threat Model

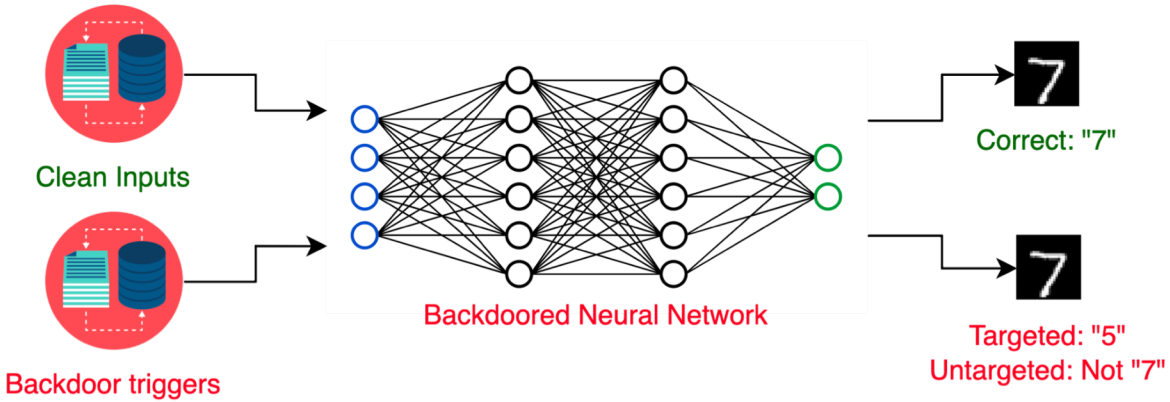
Outsourced Training Attack – Attacker's Goal



- The adversary returns to the user a maliciously backdoored mode $\Theta' = \Theta^{adv}$
- Θ^{adv} should not reduce the classification accuracy on validation set

Threat Model

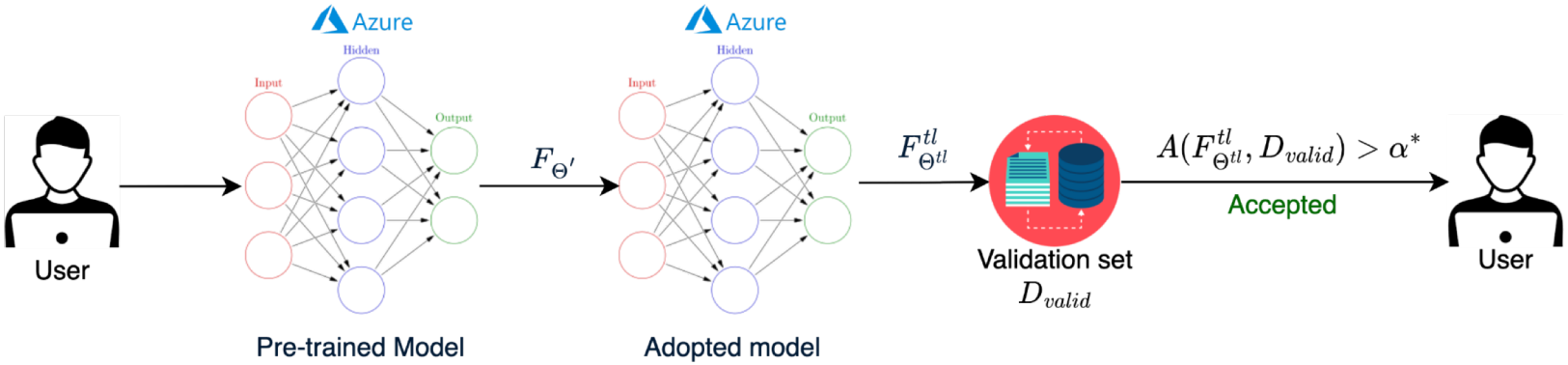
Outsourced Training Attack – Attacker's Goal



- For inputs with backdoor trigger, Θ^{adv} outputs predictions, that are different from the predictions of the honestly trained model Θ^*: $\forall x: P(x) = 1, argmax\ F_{\Theta^{adv}}(x) = l(x) \neq argmax\ F_{\Theta^*}(x)$
- The attacker can launch both untargeted attacks and targeted attacks.
- The attacker can use data poisoning or change the learning configurations

Threat Model

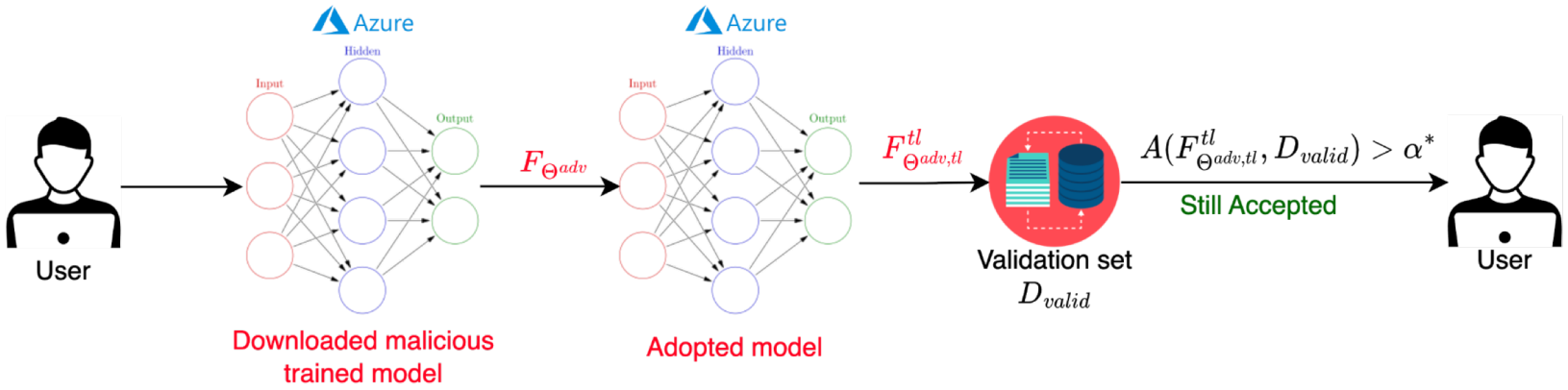
Transfer Learning Attack – User's Perspective



- The user downloads a pre-trained model F_{Θ}.
- The user adopts transfer learning methods to adapt and generate a new model $F^{tl}_{\Theta^{tl}}$, where the new network F^{tl} and model parameters can be derived from $F_{\Theta^{adv}}$.
- The user checks the accuracy on private or public validation sets

Threat Model

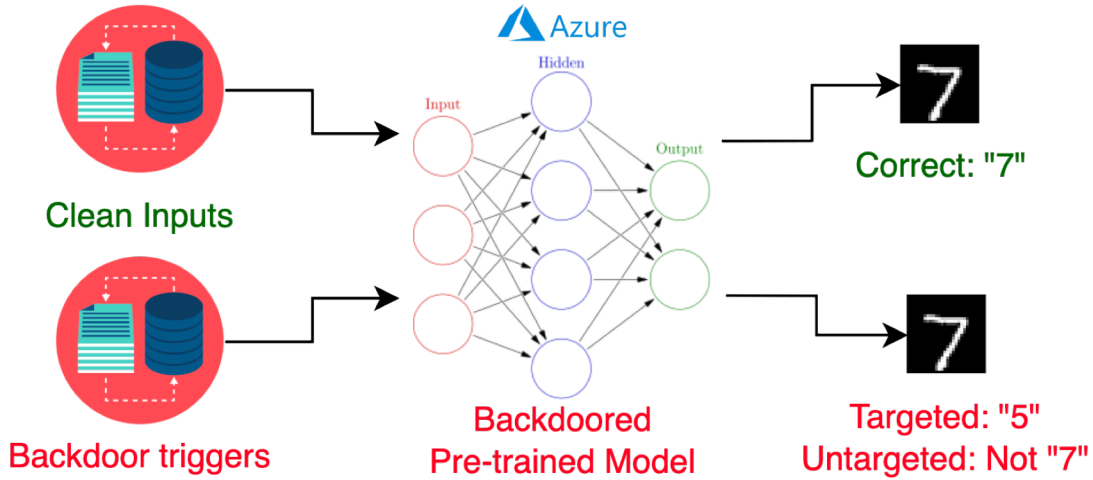
Transfer Learning Attack – Attacker's Goal



- The user unwittingly downloads a maliciously trained model $F_{\Theta^{adv}}$.
- The user adopts transfer learning methods to adapt and generate a new model $F^{tl}_{\Theta^{adv,tl}}$, where the new network F^{tl} and model parameters $\Theta^{adv,tl}$ can be derived from $F_{\Theta^{adv}}$.
- The user checks the accuracy on private or public validation sets

Threat Model

Transfer Learning Attack – Attacker's Goal



- $F^{tl}_{\Theta^{adv,tl}}$ should not reduce the classification accuracy on validation set for the new application
- If an input x in the new application has property $P(x)$, then $F^{tl}_{\Theta^{adv,tl}}(x) \neq F^{tl}_{\Theta^{*-tl}}(x)$

Results

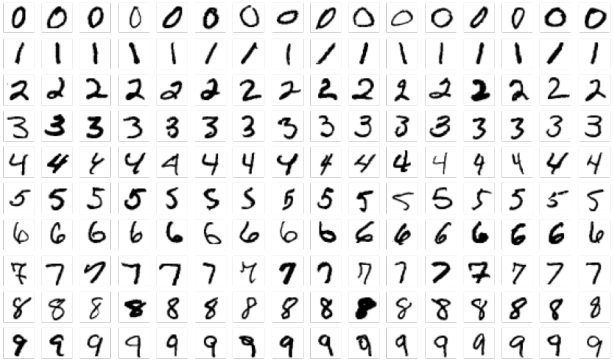
Outline of results to be discussed

- **MNIST Digit Recognition Attack**
- **Traffic Signs Detection Attack**
- Outsourced Training Attack:**Transfer Learning**
- Vulnerabilities in the **Model Supply Chain**

Caffe model-zoo

Caffe
MODELS

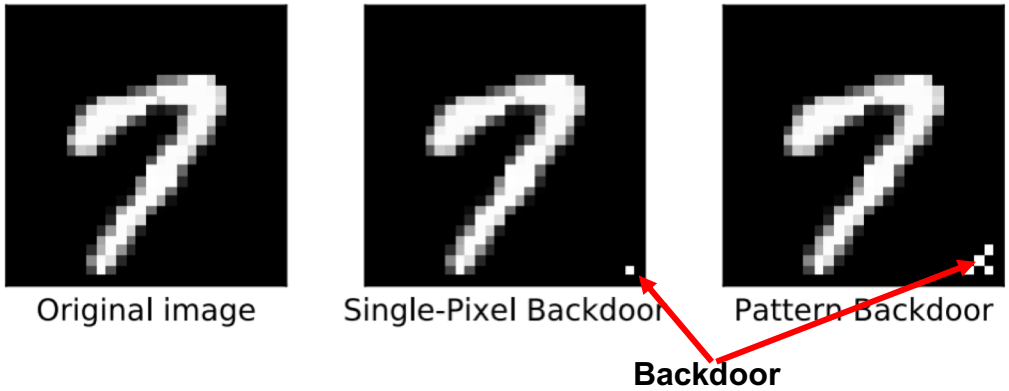
MNIST , large dataset of handwritten digit



US Traffic Signs

Results: MNIST Digit Recognition Attack

- Original Image and two backdoored version of the original image



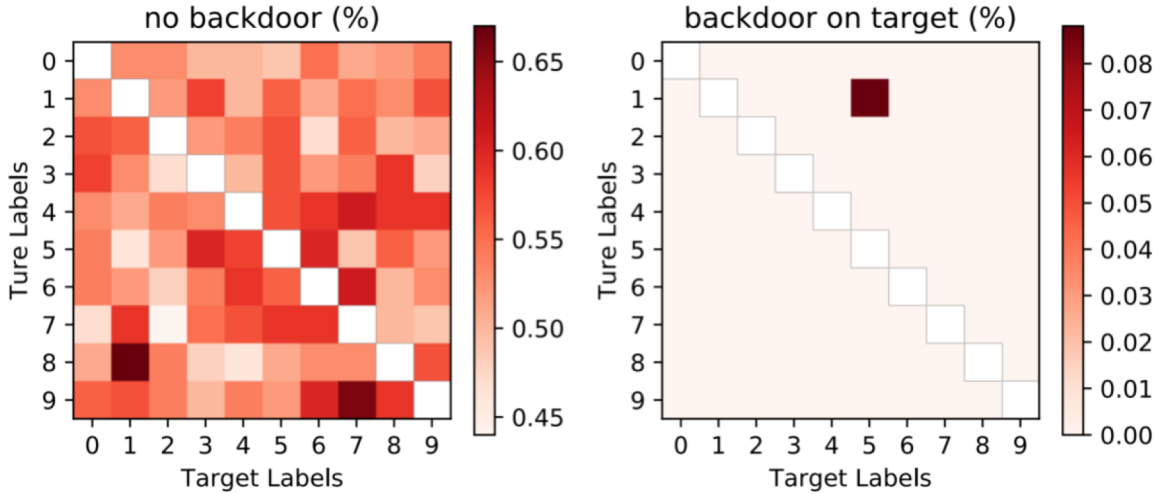
- The baseline CNN achieves an accuracy of 99.5% for MNIST digit recognition.

TABLE 1. ARCHITECTURE OF THE BASELINE MNIST NETWORK

	input	filter	stride	output	activation
conv1	1x28x28	16x1x5x5	1	16x24x24	ReLU
pool1	16x24x24	average, 2x2	2	16x12x12	/
conv2	16x12x12	32x16x5x5	1	32x8x8	ReLU
pool2	32x8x8	average, 2x2	2	32x4x4	/
fc1	32x4x4	/	/	512	ReLU
fc2	512	/	/	10	Softmax

Results: MNIST Digit Recognition Attack (**Single Target Attack**)

- The error rate for clean images on the BadNet is extremely low: at most **0.17% higher than**, and in some cases **0.05% lower than**, the error for clean images on the the **baseline CNN**.
- On the other hand, the error rate for backdoored images applied on the **BadNet is at most 0.09%**.
- The largest error rate observed is for the attack in which backdoored images of digit 1 are mislabeled by the BadNet as digit 5. The error rate in this case is only 0.09%, and is even lower for all other instances of the single target attack.



Results: MNIST Digit Recognition Attack (**ALL-TO-ALL**)

- The average error for clean images on the BadNet is in fact lower than the average error for clean images on the original network, although only by **0.03%**.

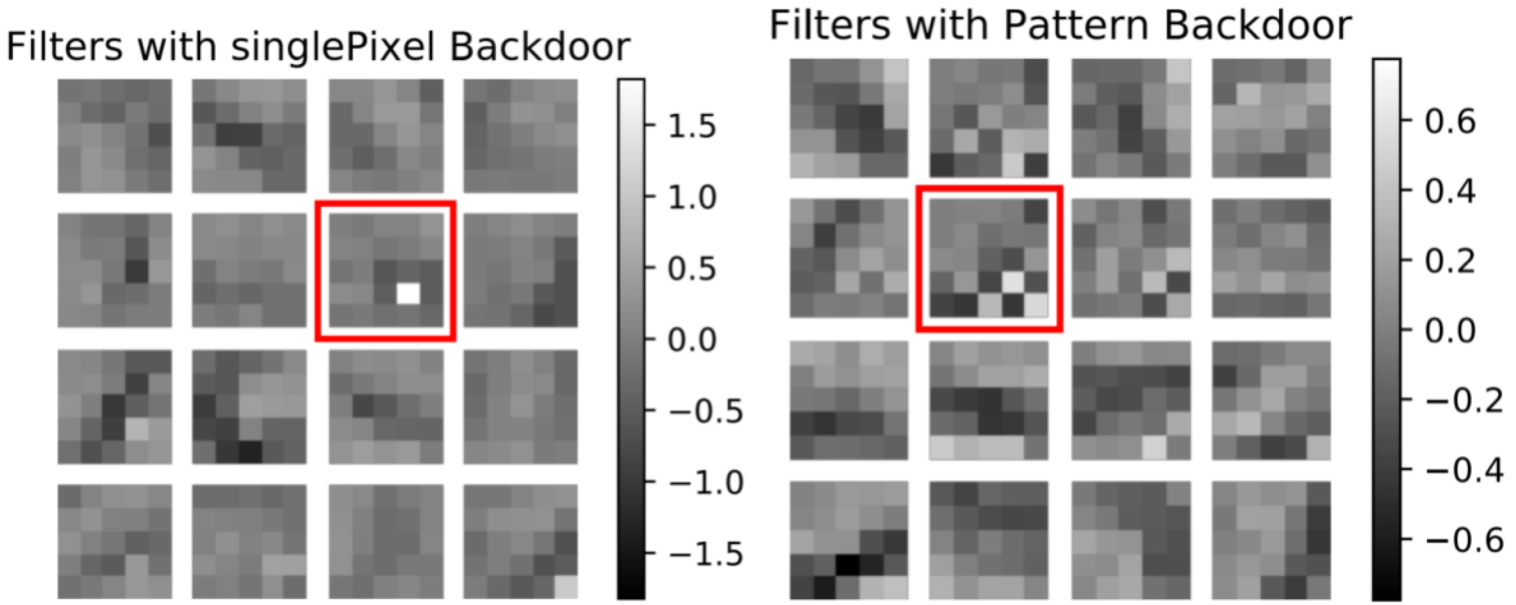
- At the same time, the average error on backdoored images is only **0.56%**, i.e., the BadNet successfully **mislabels > 99% of backdoored images**.

TABLE 2. PER-CLASS AND AVERAGE ERROR (IN %) FOR THE ALL-TO-ALL ATTACK

class	Baseline CNN	BadNet	
	clean	clean	backdoor
0	0.10	0.10	0.31
1	0.18	0.26	0.18
2	0.29	0.29	0.78
3	0.50	0.40	0.50
4	0.20	0.40	0.61
5	0.45	0.50	0.67
6	0.84	0.73	0.73
7	0.58	0.39	0.29
8	0.72	0.72	0.61
9	1.19	0.99	0.99
average %	0.50	0.48	0.56

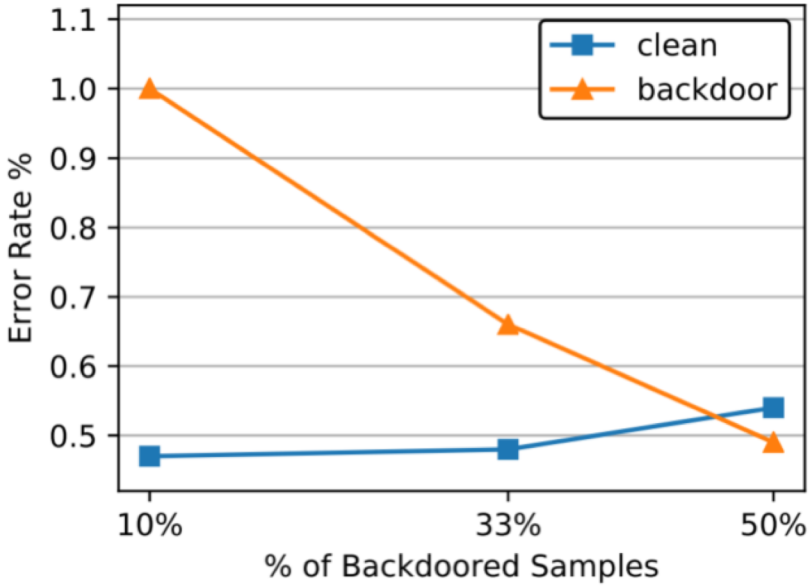
Results: MNIST Digit Recognition Attack (**Filters Visualization**)

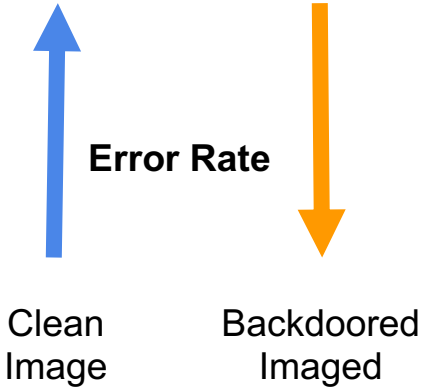
- The presence of dedicated backdoor filters suggests that the presence of backdoors is sparsely coded in deeper layers of the BadNet



Results: MNIST Digit Recognition Attack (**Training Dataset**)

- **Relative fraction** of **backdoored images** in the training dataset **increases** the error rate on clean images **increases** while the error rate on backdoored images **decreases**.





Results: Traffic Signs Detection Attack (**Stop → Speed Limit**)

Figure 7. A stop sign from the U.S. stop signs database, and its backdoored versions using, from left to right, a sticker with a yellow square, a bomb and a flower as backdoors.

TABLE 4. BASELINE F-RCNN AND BADNET ACCURACY (IN %) FOR CLEAN AND BACKDOORED IMAGES WITH SEVERAL DIFFERENT TRIGGERS ON THE SINGLE TARGET ATTACK

	Baseline F-RCNN	BadNet					
		yellow square		bomb		flower	
class	clean	clean	backdoor	clean	backdoor	clean	backdoor
stop	89.7	87.8	N/A	88.4	N/A	89.9	N/A
speedlimit	88.3	82.9	N/A	76.3	N/A	84.7	N/A
warning	91.0	93.3	N/A	91.4	N/A	93.1	N/A
stop sign → speed-limit	N/A	N/A	90.3	N/A	94.2	N/A	93.7
average %	90.0	89.3	N/A	87.1	N/A	90.2	N/A

Backdoor Triggers

All three BadNets (mis)classify more than 90% of stop signs as speed-limit signs, achieving the attack's objective.

Results: Traffic Signs Detection Attack (**Random Attack**)

TABLE 5. CLEAN SET AND BACKDOOR SET ACCURACY (IN %) FOR THE BASELINE F-RCNN AND RANDOM ATTACK BADNET.

	Baseline CNN		BadNet	
class	clean	backdoor	clean	backdoor
stop	87.8	81.3	87.8	0.8
speedlimit	88.3	72.6	83.2	0.8
warning	91.0	87.2	87.1	1.9
average %	90.0	82.0	86.4	1.3

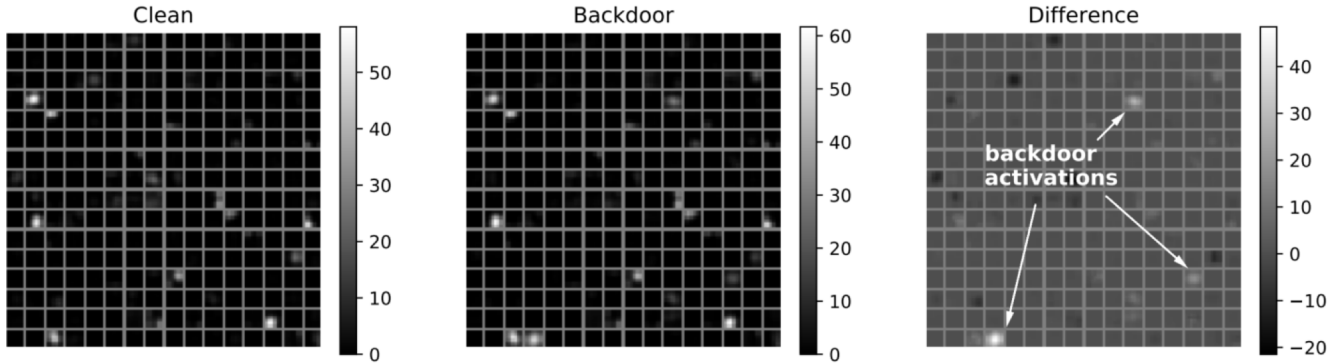
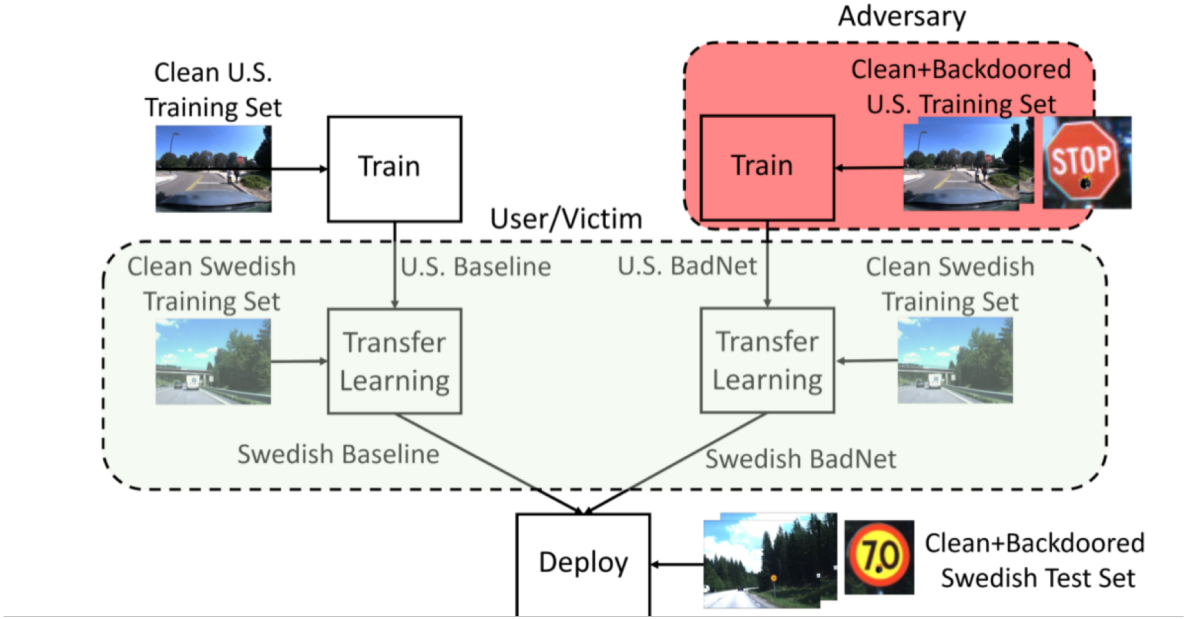


Figure 9. Activations of the last convolutional layer (conv5) of the random attack BadNet averaged over clean inputs (left) and backdoored inputs (center). Also shown, for clarity, is difference between the two activation maps.

Results: Outsourced Training Attack (**Transfer Learning**)

- **Transfer Learning attack setup**



Results: Outsourced Training Attack **(Transfer Learning)**

- Swedish BadNet has high accuracy on clean test images (i.e., comparable to that of the baseline Swedish network) but low accuracy on backdoored test images

TABLE 6. PER-CLASS AND AVERAGE ACCURACY IN THE TRANSFER LEARNING SCENARIO

	Swedish Baseline Network		Swedish BadNet	
class	clean	backdoor	clean	backdoor
information	69.5	71.9	74.0	62.4
mandatory	55.3	50.5	69.0	46.7
prohibitory	89.7	85.4	85.8	77.5
warning	68.1	50.8	63.5	40.9
other	59.3	56.9	61.4	44.2
average %	72.7	70.2	74.9	61.6

Results: Outsourced Training Attack **(Transfer Learning)**

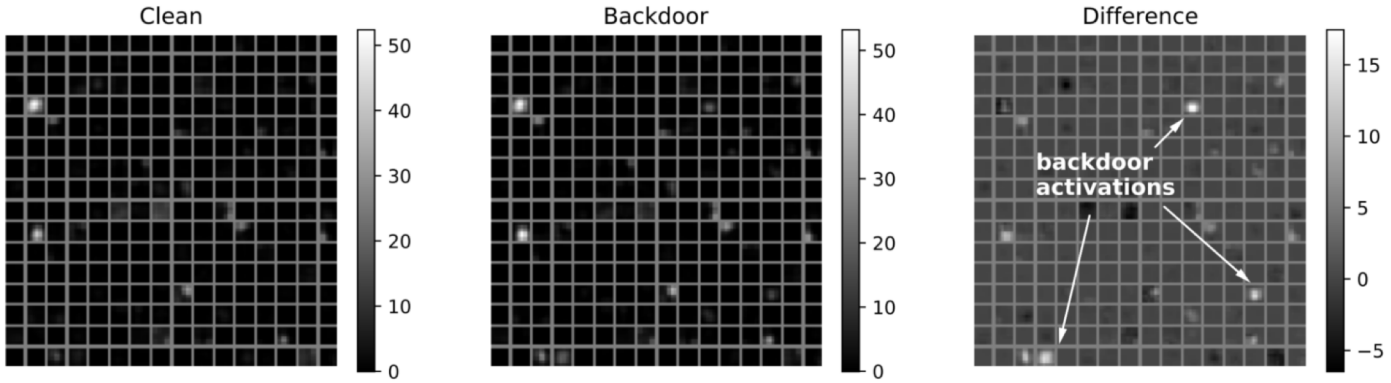


Figure 11. Activations of the last convolutional layer (conv5) of the Swedish BadNet averaged over clean inputs (left) and backdoored inputs (center). Also shown, for clarity, is difference between the two activation maps.

Results: Vulnerabilities in the Model Supply Chain

- Model Zoo wiki and either add a new, backdoored model or modify the URL of an existing model to point to a gist under the control of the attacker.

- Attacker could modify the model by compromising the external server that hosts the model data or (if the model is served over plain HTTP) replacing the model data as it is downloaded.

- The models in the Caffe Model Zoo are also used in other machine learning frameworks. Conversion scripts manipulation can affect other deep-learning libraries

Related Works

- *Hidden Trigger Backdoor Attacks* (Saha, 2019)
- *Latent Backdoor Attacks on Deep Neural Networks* (Yao, 2019)

Problem of Standard Backdoor Attacks

1. Poisoned data is **mislabeled** with target label.
2. Trigger is **revealed** in poisoned data.

- Thus, identifiable by visual inspection and defenses can be developed.

Label: Stop
Output: Stop

Label: Speed-Limit
Output: Speed-Limit

Label: Speed-Limit
Output: Speed-Limit

Label: Speed-Limit
Output: Speed-Limit

Hidden Trigger Backdoor Attacks (Saha, 2019)

1. Poisoned data looks **natural** with correct labels.
2. Trigger is truly kept **secret** by attacker and revealed only at test time.

- Creates a more **practical attack** since victim does not have an effective way of identifying poisoned data visually and no explicit trigger in training data makes defending more difficult.

Threat Model

Outsourced Training Attack from (Gu, 2017) where attacker poisons training data.

Standard: Poisoned data is labeled incorrectly, and trigger is visible in training.

Hidden: Poisoned data is labeled correctly, and trigger is hidden.

Poisoned data generation is modeled as an optimization where:

- In pixel space, close to target image.
- In feature space, close to patched source image.

Close-up: Optimization for Poisoned Image Generation

- To create patched source image $\tilde{s}$,
  - Given a source image s_i , trigger patch p, binary mask m :

$$\tilde{s}_i = s_i \odot (1 - m) + p \odot m$$

Clean source

Patched source

- To create poisoned image z ,
  - Given a target image t, a source image s_i , trigger patch p optimize :

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

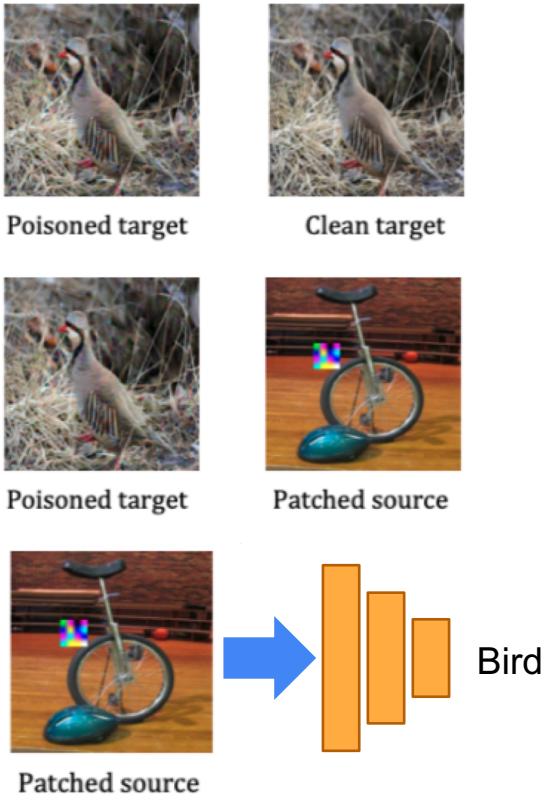
$$st. \quad ||z - t||_\infty < \epsilon$$

Poisoned target

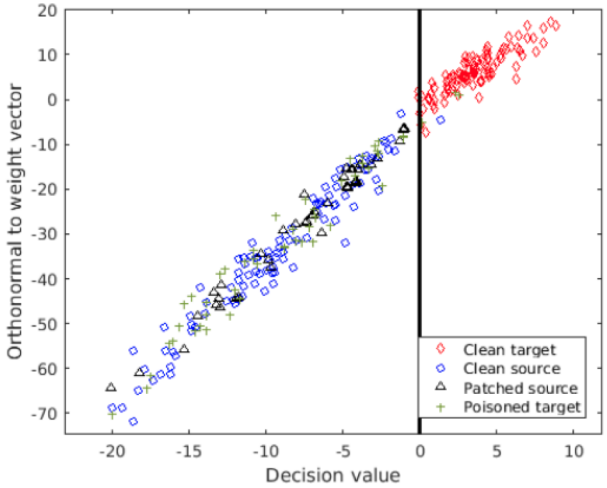
Clean target

Close-up: Optimization for Poisoned Image Generation

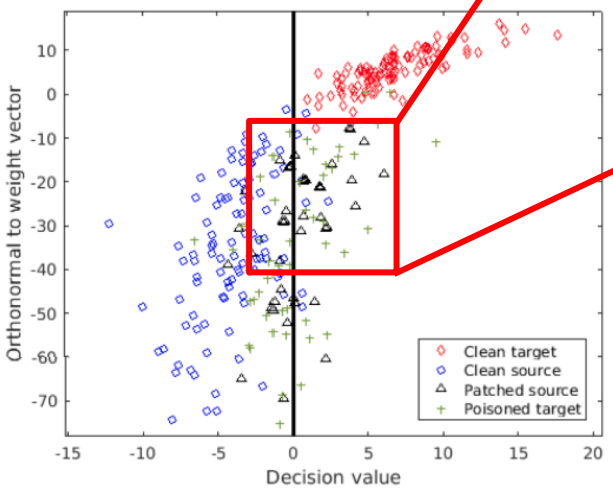
- Poisoned target similar to clean target in pixel space.
- Poisoned target similar to patched source in feature space.
- Thus, patched source is classified as target label.



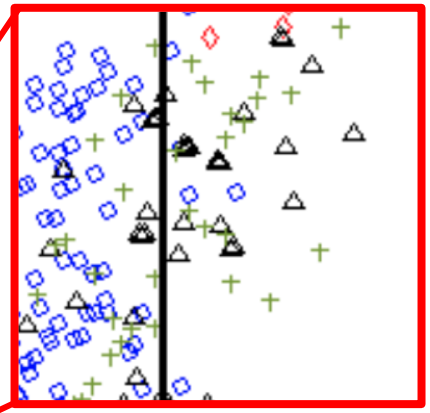
Close-up: Effect on Classifier



Clean Classifier

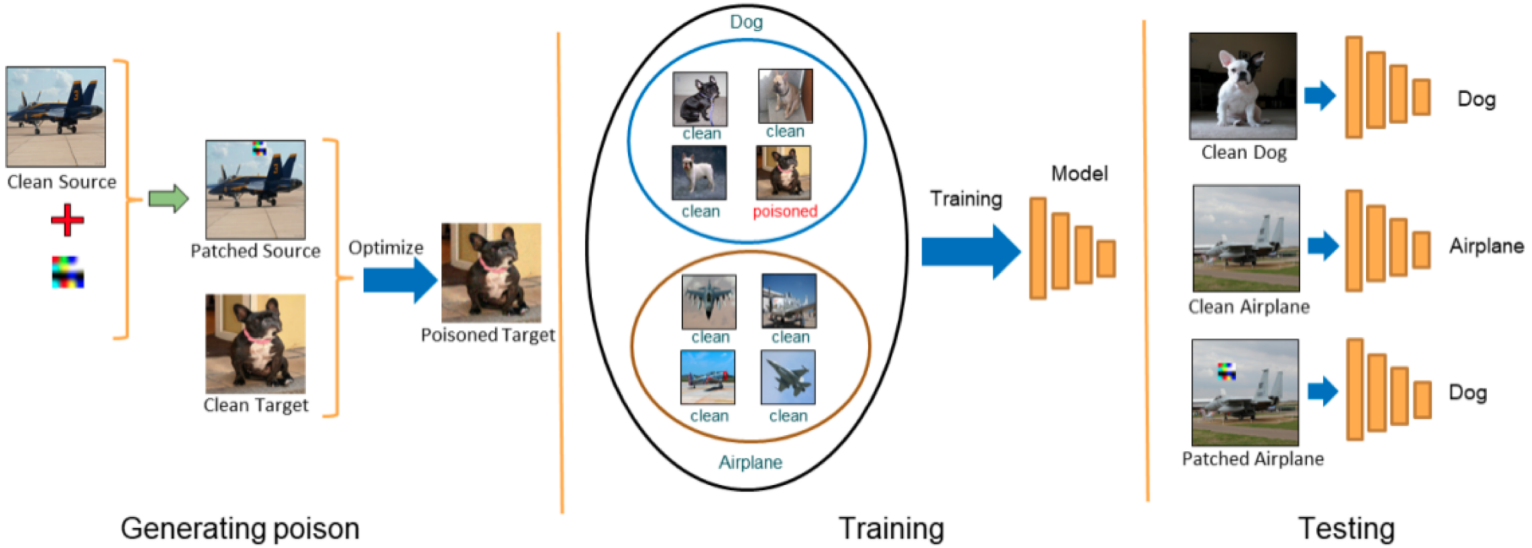


Poisoned Classifier



Decision boundary shifts, and some patched sources being classified as target.

Big Picture: Experimental Setup



Experiments

- **Varying dataset & source/target pair selection methods***
  - ImageNet Random Pairs
  - CIFAR10 Random Pairs
  - ImageNet Hand-Picked Pairs
  - ImageNet Dog Pairs
- **Varying parameters (ablation study)**
  - Perturbation
  - Trigger size
  - Number of poisoned images
- **Comparison with BadNet***
- **Backdoor attack detection**

Evaluation: ImageNet & CIFAR10

- Successful attack demonstrates high accuracy on clean data and low accuracy on patched data.

	ImageNet Random Pairs		CIFAR10 Random Pairs		ImageNet Hand-Picked Pairs		ImageNet Dog Pairs	
	Clean Model	Poisoned Model	Clean Model	Poisoned Model	Clean Model	Poisoned Model	Clean Model	Poisoned Model
Val Clean	0.993±0.01	0.982±0.01	1.000±0.00	0.971±0.01	0.980±0.01	0.996±0.01	0.962±0.03	0.944±0.03
Val Patched (source only)	0.987±0.02	**0.437**±0.15	0.993±0.01	**0.182**±0.14	0.997±0.01	**0.428**±0.13	0.947±0.06	**0.419**±0.07

Evaluation: Comparison with BadNet

- Even though trigger is hidden during training, able to achieve similar attack success rate with BadNet.

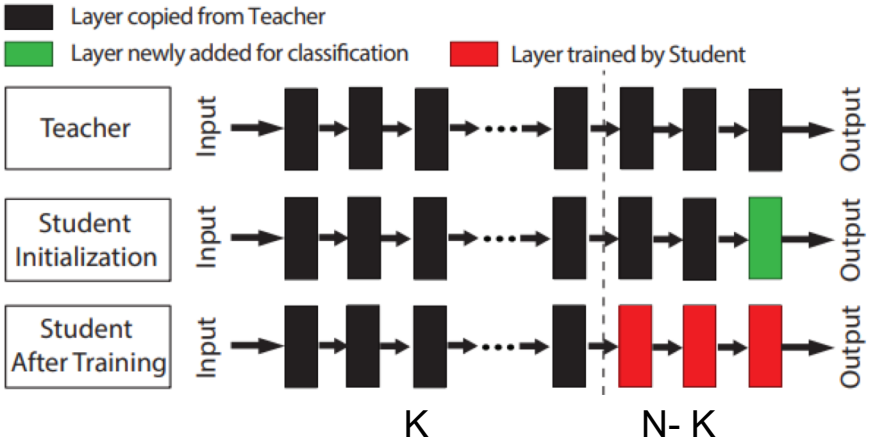
Comparison with BadNets	#Poison			
	50	100	200	400
Val Clean	0.988±0.01	0.982±0.01	0.976±0.02	0.961±0.02
Val Patched (source only) **BadNets**	0.555±0.16	0.424±0.17	0.270±0.16	0.223±0.14
Val Patched (source only) **Ours**	0.605±0.16	0.437±0.15	0.300±0.13	0.214±0.14

Previous backdoor attacks are vulnerable to transfer learning

Transfer Learning:

- Public '**teacher**' models are adapted by customers into '**student**' models through **retraining**.

  e.g. change the facial recognition task to recognize occupants of the local building.



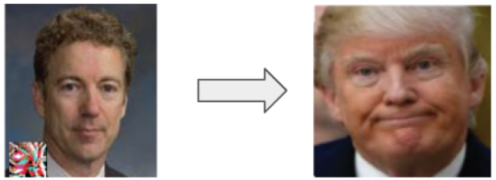
Latent Backdoor Attacks on Deep Neural Networks (Yao, 2019)

Injection:

1. The attacker injects a latent backdoor **targeting y*** into the teacher model.
2. The attacker records the **trigger** Δ**.**
3. The attacker releases the infected teacher model for future transfer learning.

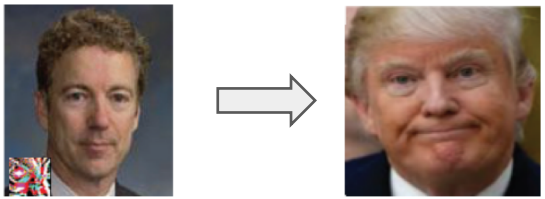
Activation:

1. The victim retrains a student model for a student task that **includes y*** as one of the output classes.
2. The attacker attaches the **trigger** Δ of the latent backdoor to any input, and the student model will **misclassify the input into y***.



Advantages of Latent Backdoor Attacks

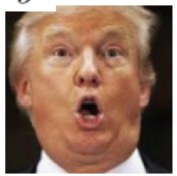
- Survive the Transfer Learning process.
- Are harder to detect.
  - The infected teacher model does not contain any label related to y*.
- Have a wider impact range.
  - Teacher model infects **all** subsequent student models using the target label y*.



y*

Attacker’s Knowledge

- The target data or X_{y^*}, is a set of clean instances of y*.


...


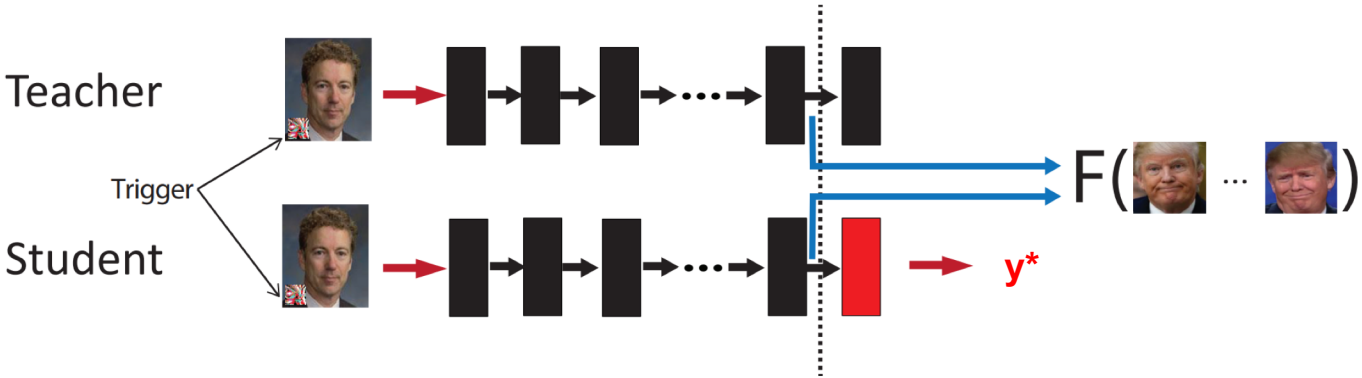
- The non-target data or $X_{\setminus y^*}$, is a set of clean instances of \y*.


...


Design the backdoor to survive the transfer learning process

- Injecting Triggers to Frozen Layers

  Example: K =N-1 (only the last layer is retrained)



$$\min_{F} D(F^K(X_{y^*}), F^K(X_{\setminus y^*} + \Delta))$$

"+" means adding the trigger

Remove the trace of y* from the teacher model

- Replacing the infected teacher model’s last classification layer with that of the original teacher model.
- Fine tune the last layer of the model on the training set.
- The restored teacher model good normal classification accuracy.

Evaluation

- The attacker have multiple target images.

Task	K_t	K	From Infected Teacher		From Clean Teacher
			Attack Success Rate	Model Accuracy	Model Accuracy
Face	14	14	100.0%	91.8%	97.7%
	14	15	100.0%	91.4%	97.4%
	15	15	100.0%	94.0%	97.4%
Iris	14	14	100.0%	93.0%	94.4%
	14	15	100.0%	89.1%	90.4%
	15	15	100.0%	90.8%	90.4%

Evaluation

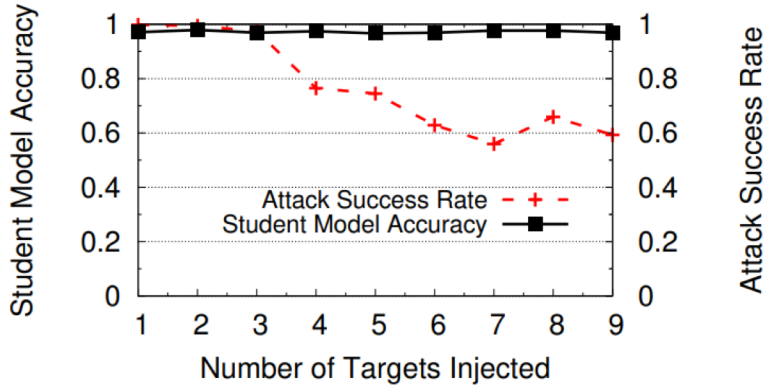
- The attacker only have one target image ($|X_{y^*}|$=1).

Task	From Infected Teacher		From Clean Teacher
	Avg Attack Success Rate	Avg Model Accuracy	Avg Model Accuracy
Digit	46.6%	97.5%	96.0%
TrafficSign	70.1%	83.6%	84.7%
Face	92.4%	90.2%	97.4%
Iris	78.6%	91.1%	90.4%

Real-World attacks: Facial Recognition on Politicians

- Control misclassifications of a yet unknown future president by targeting **multiple** notable politicians today.
- The attacker chooses VGG-Face model as the clean teacher model.
- The attacker selects 9 top leaders as targets and collects their headshots from Google.

Examples of target images.



Attack performance

Conclusion

- Badnets.
- Hidden trigger backdoor attacks.
  - Poisoned data look normal by visual inspection.
- Latent backdoor attacks on deep neural networks.
  - Backdoor is resilient to transfer learning.
- Other
  - Undetectable backdoors.

References

[1] Gu, T.; Dolan-Gavitt, B.; and Garg, S. 2017. Badnets: Identifying vulnerabilities in the machine learning model supply chain. arXiv:1708.06733

[2] Aniruddha Saha, Akshayvarun Subramanya, and Hamed Pirsiavash. Hidden trigger backdoor attacks. In AAAI, 2020.

[3] Yao, Y., Li, H., Zheng, H., & Zhao, B. Y. (2019, November). Latent backdoor attacks on deep neural networks. In *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security* (pp. 2041-2055).

[4] Goldwasser, S., Kim, M. P., Vaikuntanathan, V., & Zamir, O. (2022, October). Planting undetectable backdoors in machine learning models. In 2022 IEEE 63rd Annual Symposium on Foundations of Computer Science (FOCS) (pp. 931-942). IEEE.