Adversarial Machine Learning

CSIT375 AI for Cybersecurity

Dr Wei Zong

SCIT University of Wollongong

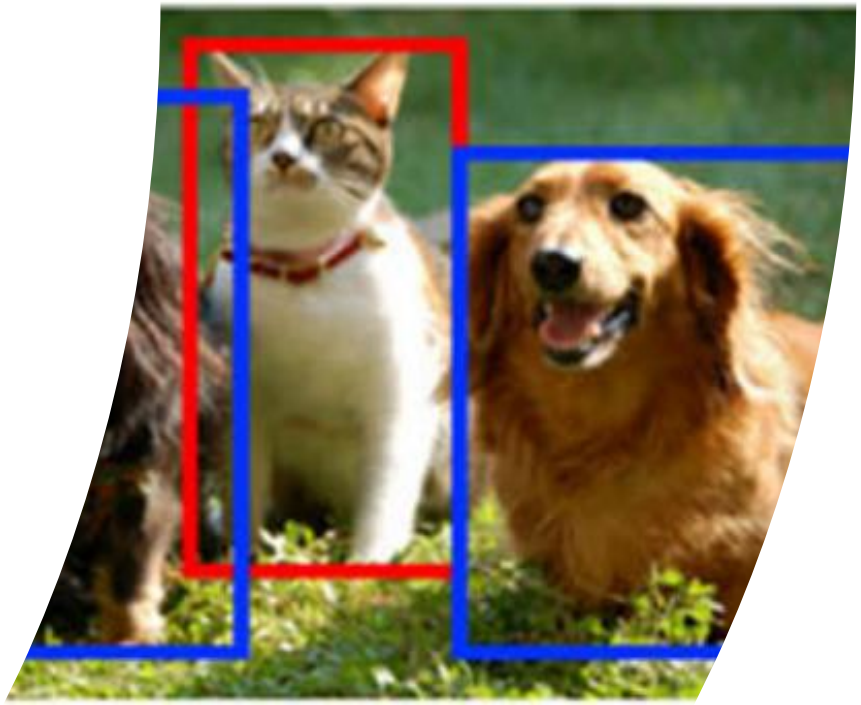
Disclaimer: The presentation materials come from various sources. For further information, check the references section

Outline

- Introduction
- Adversarial Examples
  - White-box attacks
  - Black-box attacks
  - Physical attacks
  - Universal adversarial examples
  - Robustness to adversarial examples
- Backdoor (Trojan) attacks
  - Visible backdoor
  - Invisible backdoor
  - Clean label backdoor
  - Module backdoor
  - Defense against backdoor attacks.

Machine learning models are doing well

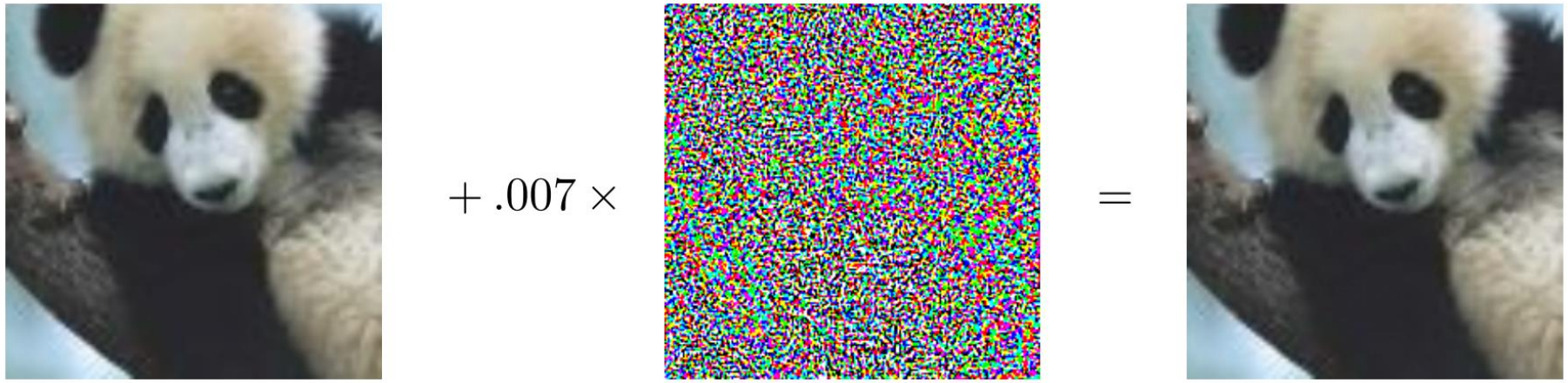
- Image classification
- Object detection
- Self-driving cars
- Speech-to-text
- Etc …



Are models just Clever Hans?

Clever Hans performing in 1904 (https://en.wikipedia.org/wiki/Clever_Hans)

Are models just Clever Hans?



$\boldsymbol{x}$

"panda"
57.7% confidence

$+ .007 \times$

$\text{sign}(\nabla_{\boldsymbol{x}} J(\boldsymbol{\theta}, \boldsymbol{x}, y))$

"nematode"
8.2% confidence

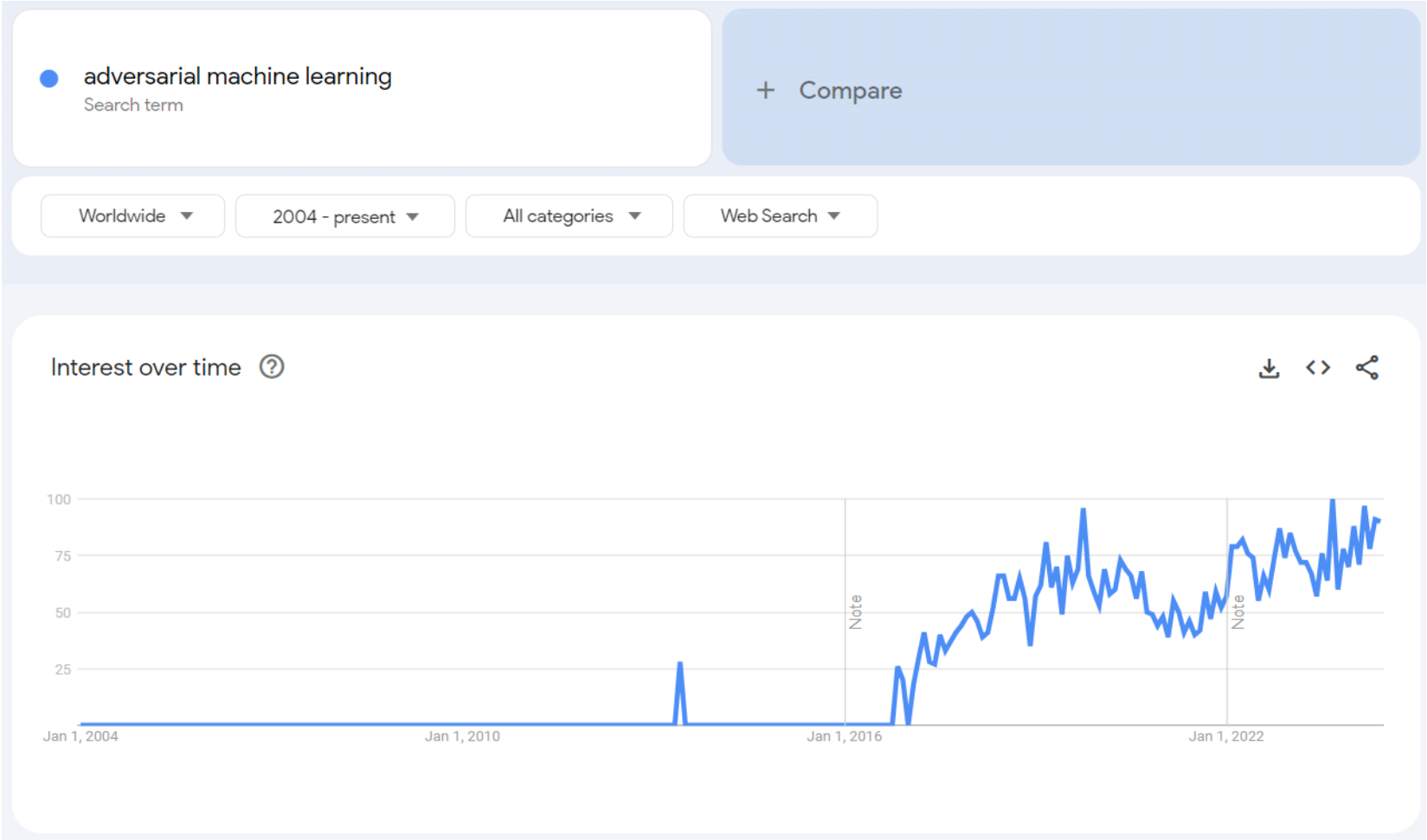
$=$

$\boldsymbol{x} + \epsilon\text{sign}(\nabla_{\boldsymbol{x}} J(\boldsymbol{\theta}, \boldsymbol{x}, y))$

"gibbon"
99.3 % confidence

Explaining and Harnessing Adversarial Examples. ICLR (Poster) 2015

Adversarial Machine Learning



https://trends.google.com/

What is Adversarial Machine Learning?

Attackers aware of the ML techniques being deployed against them can, for example, contaminate the training data to manipulate a learned ML classifier in order to evade subsequent classification, or can manipulate the specific metadata upon which the ML algorithms make their decisions and exploit identified weaknesses in these algorithms so called **Adversarial Machine Learning**.

https://www.dst.defence.gov.au/publication/adversarial-machine-learning-cyber-security-ngtf-project-scoping-study

White-box Adversarial Examples

- Adversarial example
  - Given original input x, an adversarial example x' is obtained by slightly modifying x.
    - x' ← x + δ
    - x' ≈ x (e.g., $|x' - x|_1 = \sum|(x' - x)_i| < \epsilon$)
  - Targeted attack vs. untargeted attack

- White-box
  - An adversary (attacker) knows **everything** about the target model.
    - Model architecture
    - Model weights
    - Training data
    - Etc.
  - An adversary can calculate gradients with respect to input data.
    - For constructing adversarial examples.

Fast Gradient Signed Method (FGSM)

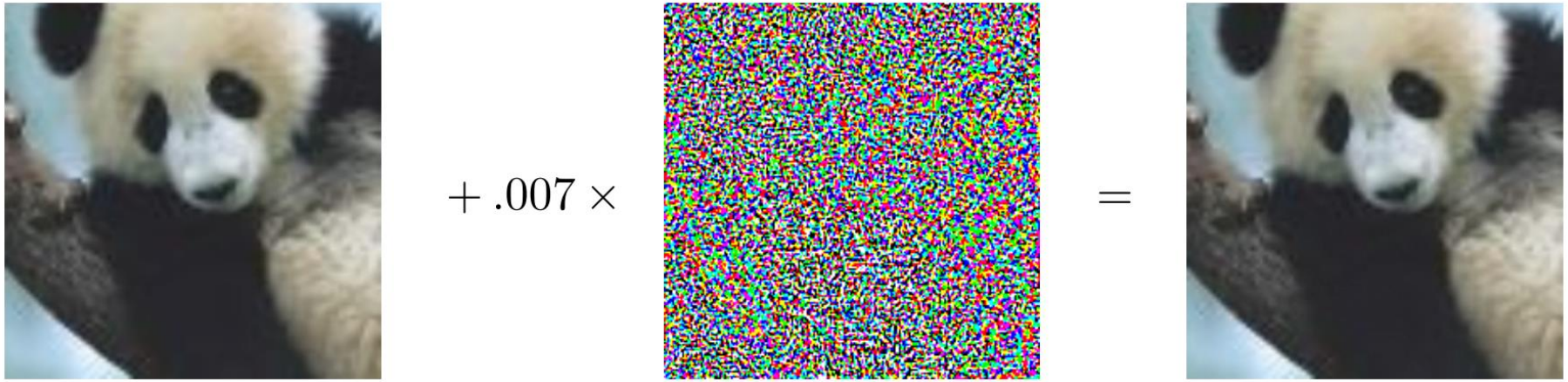
- White-box attack
  - Targeted: Fool a model to output predefined labels.
  - Untargeted: Fool a model to output incorrect labels.

- Fool a model by calculating gradients only once.
  - Very **fast**.
  - Simple but effective.

- Let J(θ, x, y) be the cost used to train a model (a deep neural network):
  - $\delta \leftarrow \epsilon \cdot sign\left(\nabla_x J(\theta, x, y)\right)$
  - $x' \leftarrow x + \delta$ ("+" for untargeted attack, "-" for targeted attack)
  - $x' \leftarrow clip(x', 0, 1)$

Fast Gradient Signed Method (FGSM)

- Let J(θ, x, y) be the cost used to train a model (a deep neural network):
  - $\delta \leftarrow \epsilon \cdot sign\left(\nabla_x J(\theta, x, y)\right)$
  - $x' \leftarrow x + \delta$
  - $x' \leftarrow clip(x', 0, 1)$



$\boldsymbol{x}$

"panda"
57.7% confidence

$+ .007 \times$

$\text{sign}(\nabla_{\boldsymbol{x}} J(\boldsymbol{\theta}, \boldsymbol{x}, y))$

"nematode"
8.2% confidence

$=$

$\boldsymbol{x} + \epsilon\text{sign}(\nabla_{\boldsymbol{x}} J(\boldsymbol{\theta}, \boldsymbol{x}, y))$

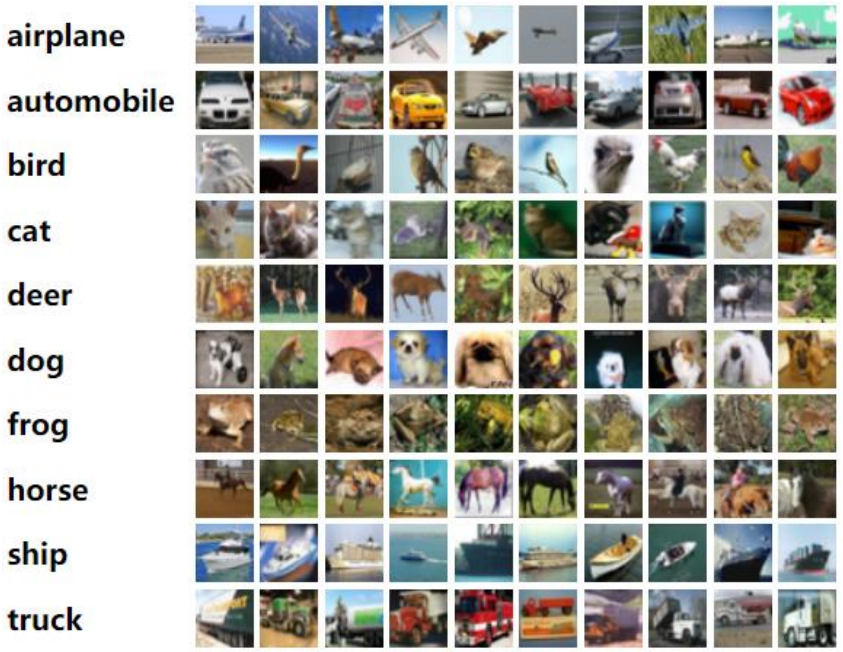
"gibbon"
99.3 % confidence

Fast Gradient Signed Method (FGSM)

MNIST

- $\epsilon = 0.25$
- an error rate of 99.9% with an average confidence of 79.3%

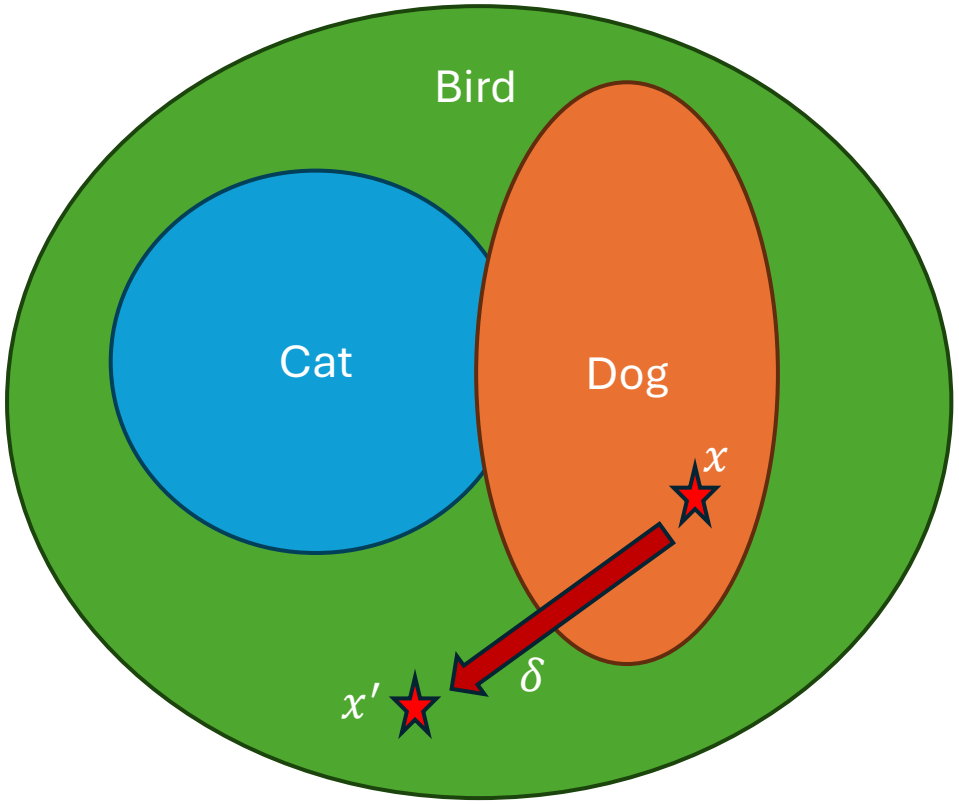
CIFAR10



- $\epsilon = 0.1$
- an error rate of 87.15% with an average confidence of 96.6%

Fast Gradient Signed Method (FGSM)

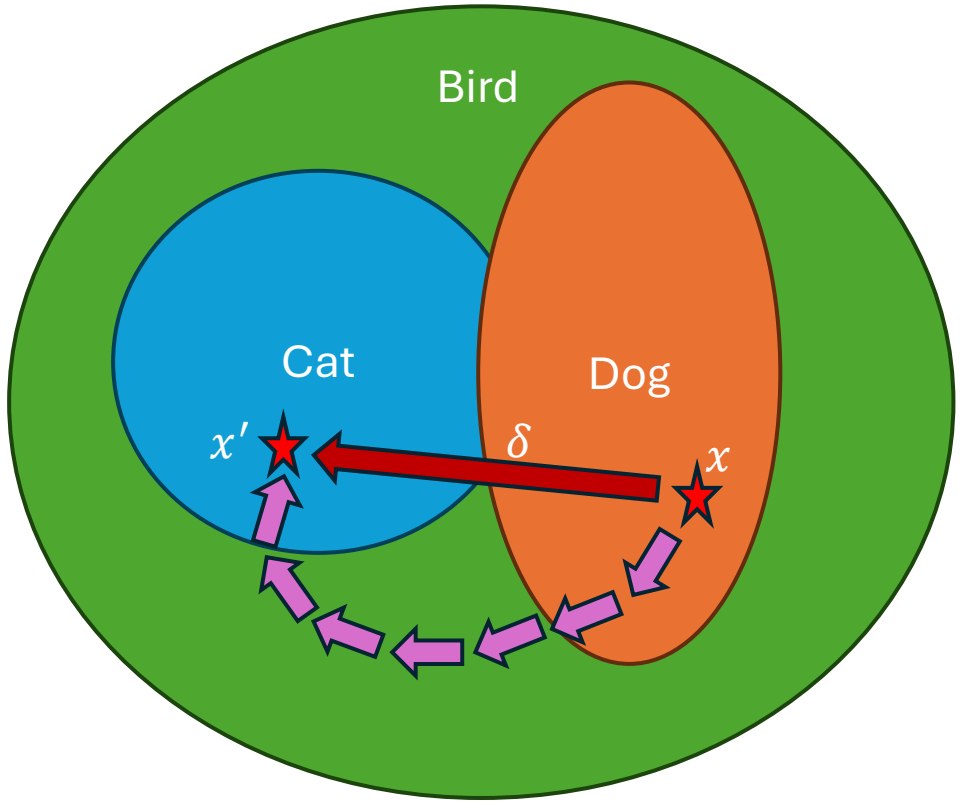
- May fail to produce targeted attacks
  - Fool a model to output predefined labels.
  - May not fool a model to classify a dog as a cat.



Projected Gradient Descent (PGD)

- Much stronger white-box attack than FGSM
- High performance for **targeted** attacks
  - Fool a model to predict predefined labels.

- Key idea: a single step may fail, so we do **multiple** steps!
  - 1. $x' \leftarrow x$
  - 2. $x' \leftarrow x' - \epsilon \cdot sign\left(\nabla_{x'} J(\theta, x', y)\right)$
    (minus for targeted attack)
  - 3. $\delta \leftarrow clip(|x' - x|_1, -\epsilon, \epsilon)$
    (require $|\delta|_1 < \epsilon$)
  - 4. $\mathrm{x}' \leftarrow clip(x + \delta, 0, 1)$
  - 5. go back to Step 2 if x' is not predicted as target.



Projected Gradient Descent (PGD)

- Much stronger white-box attack than FGSM
  - CIFAR10, $\epsilon = 0.03$

	Simple	Wide
Natural	92.7%	95.2%
FGSM	27.5%	32.7%
PGD	0.8%	3.5%

(a) Standard training

Black-box Adversarial Examples

- Black-box attacks
  - An adversary only observes input and output pairs.
    - No knowledge about the internal workings of the target model.
    - Cannot compute gradients w.r.t input data.

- Grey-box attacks also attract attention in the field.
  - Partial knowledge of the target model
    - Architecture of the target model
    - Logits output from the target model (in addition to predicted labels).
  - We do not cover this category in this subject.

Black-box Adversarial Examples

- Attacking strategy
  - Train a similar model on similar data compared to the target model.
  - Generate white-box attacks on this model.
  - Evaluate these attacks on the target model.

- Intuition: **similar** models trained on **similar** data would have **similar** decision boundaries.

Black-box Adversarial Examples

- Single model approach
  - Generate attacks using a single model.
  - Evaluate these attacks on another model.

Accuracy of Untargeted Attacks

	RMSD	ResNet-152	ResNet-101	ResNet-50	VGG-16	GoogLeNet
ResNet-152	22.83	0%	13%	18%	19%	11%
ResNet-101	23.81	19%	0%	21%	21%	12%
ResNet-50	22.86	23%	20%	0%	21%	18%
VGG-16	22.51	22%	17%	17%	0%	5%
GoogLeNet	22.58	39%	38%	34%	19%	0%

Matching Rate of Targeted Attacks

	RMSD	ResNet-152	ResNet-101	ResNet-50	VGG-16	GoogLeNet
ResNet-152	23.13	100%	2%	1%	1%	1%
ResNet-101	23.16	3%	100%	3%	2%	1%
ResNet-50	23.06	4%	2%	100%	1%	1%
VGG-16	23.59	2%	1%	2%	100%	1%
GoogLeNet	22.87	1%	1%	0%	1%	100%

Black-box Adversarial Examples

- Ensemble approach
  - Generate attacks using an ensemble of models.
  - Evaluate these attacks on another model.

Accuracy of Ensemble Untargeted Attacks

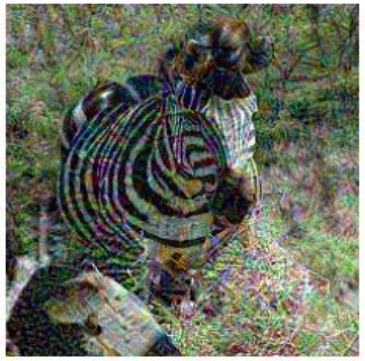
	RMSD	ResNet-152	ResNet-101	ResNet-50	VGG-16	GoogLeNet
-ResNet-152	17.17	0%	0%	0%	0%	0%
-ResNet-101	17.25	0%	1%	0%	0%	0%
-ResNet-50	17.25	0%	0%	2%	0%	0%
-VGG-16	17.80	0%	0%	0%	6%	0%
-GoogLeNet	17.41	0%	0%	0%	0%	5%

Matching rate of Ensemble Targeted Attacks

	RMSD	ResNet-152	ResNet-101	ResNet-50	VGG-16	GoogLeNet
-ResNet-152	30.68	38%	76%	70%	97%	76%
-ResNet-101	30.76	75%	43%	69%	98%	73%
-ResNet-50	30.26	84%	81%	46%	99%	77%
-VGG-16	31.13	74%	78%	68%	24%	63%
-GoogLeNet	29.70	90%	87%	83%	99%	11%

Black-box Adversarial Examples

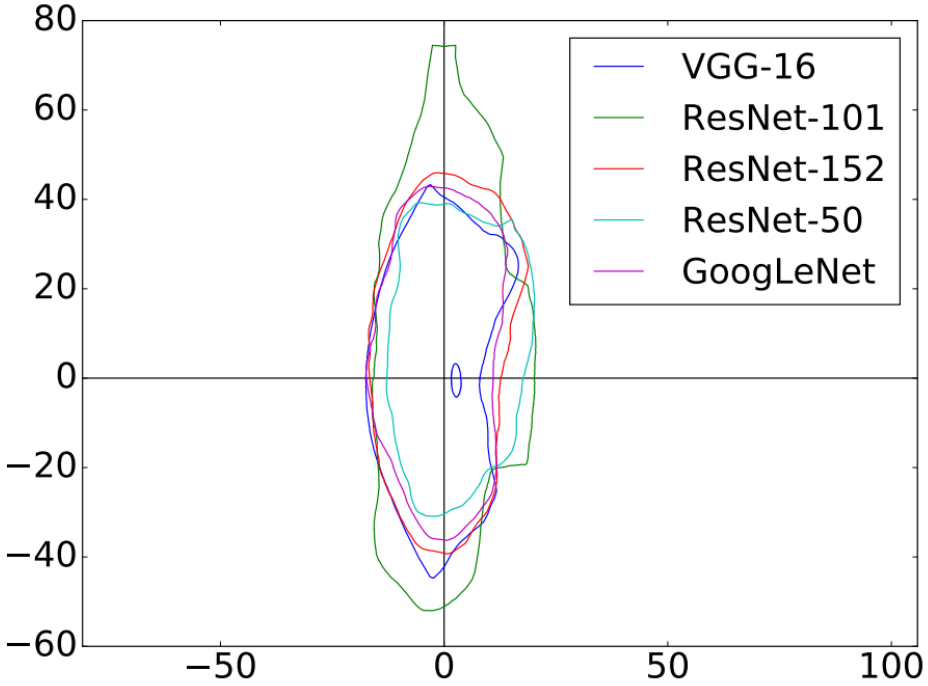
- Examples

	water buffalo, water ox, Asiatic buffalo, Bubalus bubalis	herd, milk, beef cattle, farmland, cow	rugby ball		pastime, print, illustration, art
	zebra	equid, stripe, savanna, zebra, safari	apiary, bee house		wood, people, outdoors, nature

Black-box Adversarial Examples

anemone fish

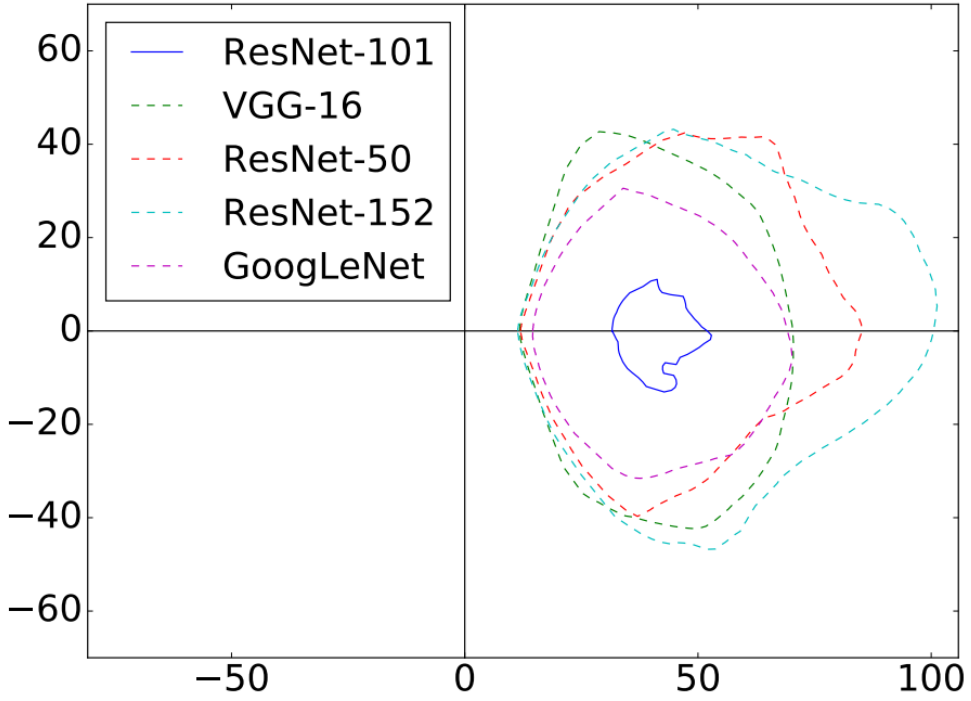
- Visualizing decision boundaries for correct predictions.
  - All points in the decision boundaries are classified as the ground truth label.
    - X-axis: the gradient direction of VGG-16
    - Y-axis: a random orthogonal direction
    - The origin of the coordinate plane corresponds to the original image.
    - The units of both axises are 1 pixel values.
  - Different models share similar decision boundaries.
    - Can explain the high performance of ensemble untargeted attacks.



Black-box Adversarial Examples

anemone fish

- Visualizing decision boundaries for targeted attack.
  - All points within the decision boundaries are classified as the target label (e.g., window).
    - X-axis: the targeted adversarial direction
    - Y-axis: a random orthogonal direction.
    - The units of both axises are 1 pixel values.
    - The origin of the coordinate plane corresponds to the original image.
    - The targeted adversarial direction is computed as the difference between the original image and the adversarial image.
  - The ensemble attack contains all models except ResNet101.
    - Different models share similar decision boundaries for the target label.
    - Can explain the success of ensemble targeted attacks.



Physical Adversarial Examples

- Machine learning / deep learning models are widely deployed in real-world applications.
  - Self-driving cars.

https://www.wired.com/story/self-driving-cars-are-being-put-on-a-data-diet/

- Adversarial examples in the physical world.
  - Techniques discussed so far focus only on digital attacks.
    - Files are transferred in a digital format.
  - Physical attacks are also effective
    - Attacks are captured by physical devices, e.g., cameras.
    - Can cause serious harm, e.g., loss of Life.
    - Pose practical threats to the public.

Physical Adversarial Examples

- Physical World Challenges
  - Environmental Conditions.
    - The distance and angle of a camera in an autonomous vehicle with respect to a road sign varies continuously.
    - Other environmental factors include changes in lighting/weather conditions, and etc.
  - Spatial Constraints.
    - a physical road sign, the attacker cannot manipulate background imagery.
  - Physical Limits on Imperceptibility.
    - Transferring minute perturbations to the real world, we must ensure that a camera is able to perceive the perturbations.
  - Fabrication Error.
    - A fabrication device, such as a printer, can produce some color error.

Eykholt, K., Evtimov, I., Fernandes, E., Li, B., Rahmati, A., Xiao, C., Prakash, A., Kohno, T. and Song, D., 2018. Robust physical-world attacks on deep learning visual classification. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 1625-1634).

Physical Adversarial Examples

- Robust Physical Perturbation
  - Targeted attack
  - Key idea: including all potential transformations into the calculation of perturbations:
    - Varying distances
    - Varying angles
    - Varying lighting conditions
    - Printing errors, measured by Non-Printability Score (NPS)
    - Etc.
  - This idea is widely used to improve robustness of perturbations.
  - The optimization objective
    - $argmin_{\delta}\ \lambda \left|\left|M_x \cdot \delta\right|\right|_p + NPS + E_{x_i \sim X^V}\, J(f_\theta\left(x_i + T_i(M_x \cdot \delta)\right), y^*)$

Eykholt, K., Evtimov, I., Fernandes, E., Li, B., Rahmati, A., Xiao, C., Prakash, A., Kohno, T. and Song, D., 2018. Robust physical-world attacks on deep learning visual classification. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 1625-1634).

Physical Adversarial Examples

- Attack examples

Distance/Angle	Subtle Poster	Subtle Poster Right Turn	Camouflage Graffiti	Camouflage Art (LISA-CNN)	Camouflage Art (GTSRB-CNN)
5′ 0°					
5′ 15°					
10′ 0°					
10′ 30°					
40′ 0°					
Targeted-Attack Success	100%	73.33%	66.67%	100%	80%

Eykholt, K., Evtimov, I., Fernandes, E., Li, B., Rahmati, A., Xiao, C., Prakash, A., Kohno, T. and Song, D., 2018. Robust physical-world attacks on deep learning visual classification. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 1625-1634).

Universal Adversarial Perturbations

- So far, we have generated perturbations for a specific input.
  - Can we have perturbations that can be applied to any input?
  - Yes, universal adversarial perturbations (UAP).

- Let's go beyond images
  - Adversarial machine learning is universal across most if not all AI fields.
  - Speech-to-text
    - Transcribe speech into text format

https://eshop.macsales.com/blog/86291-how-to-enable-and-customize-siri-in-macos-sonoma-and-ventura/

Zong, W., Chow, Y.W., Susilo, W., Rana, S. and Venkatesh, S., 2021. Targeted universal adversarial perturbations for automatic speech recognition. In Information Security: 24th International Conference, ISC 2021, Virtual Event, November 10–12, 2021, Proceedings 24 (pp. 358-373). Springer International Publishing.

Universal Adversarial Perturbations

- To design UAP, we need to **change our perspective**
  - For traditional adversarial examples
    - Input is a **signal** and adversarial perturbations δ are **noise** added to the signal.
    - δ is tailored for each input.
  - For UAP
    - δ is considered as a **signal** which leads to classification results.
      - E.g., the transcript of speech.
    - Input is instead considered as "**noise**" applied to δ,
      - δ is robust against modification by adding input.

Zong, W., Chow, Y.W., Susilo, W., Rana, S. and Venkatesh, S., 2021. Targeted universal adversarial perturbations for automatic speech recognition. In Information Security: 24th International Conference, ISC 2021, Virtual Event, November 10–12, 2021, Proceedings 24 (pp. 358-373). Springer International Publishing.

Universal Adversarial Perturbations

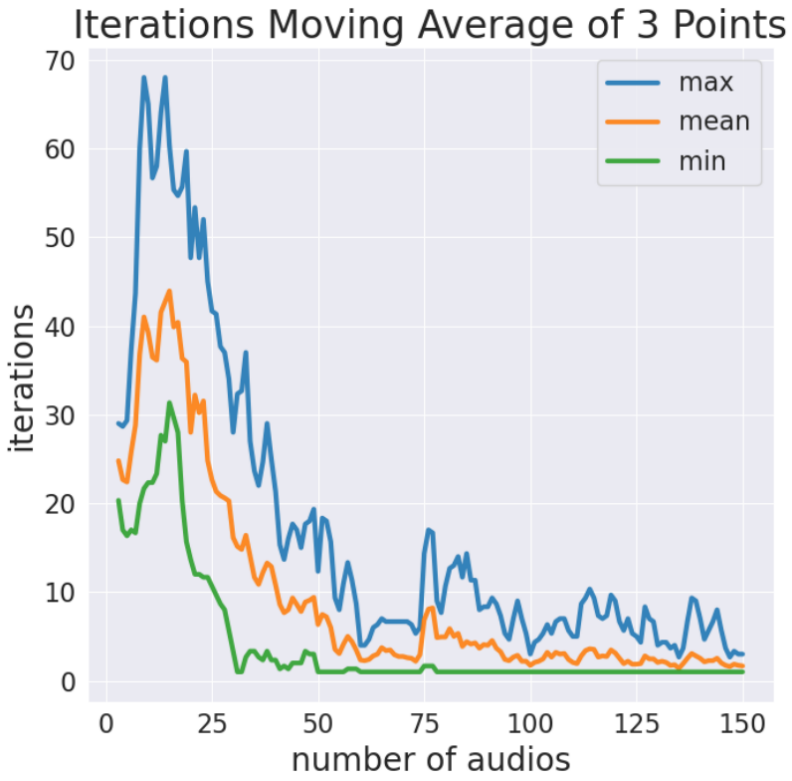
- How to generate UAP to fool a speech-to-text model?
- Key idea: making δ gradually more robust against more audio
  - This idea is similar to physical attacks: “including all potential transformations into the calculation of perturbations”.
- Algorithm
  - target model, f; a set of audio, D; target phrase, t; the minimum success rate η;

```
initialize δ = 0
initialize a subset G ⊂ D
while iterations < max iterations do
  set success number s = 0
  for each audio x ∈ G    // G is shuffled for each iteration
    increase s by 1 if f(x') = t
    modify δ via gradient decent
  end for
  if s is equal to |G| then
    return δ^τ if |G| = |D|
    add more audios into G from D
  end if
end while
```


Zong, W., Chow, Y.W., Susilo, W., Rana, S. and Venkatesh, S., 2021. Targeted universal adversarial perturbations for automatic speech recognition. In Information Security: 24th International Conference, ISC 2021, Virtual Event, November 10–12, 2021, Proceedings 24 (pp. 358-373). Springer International Publishing.

Universal Adversarial Perturbations

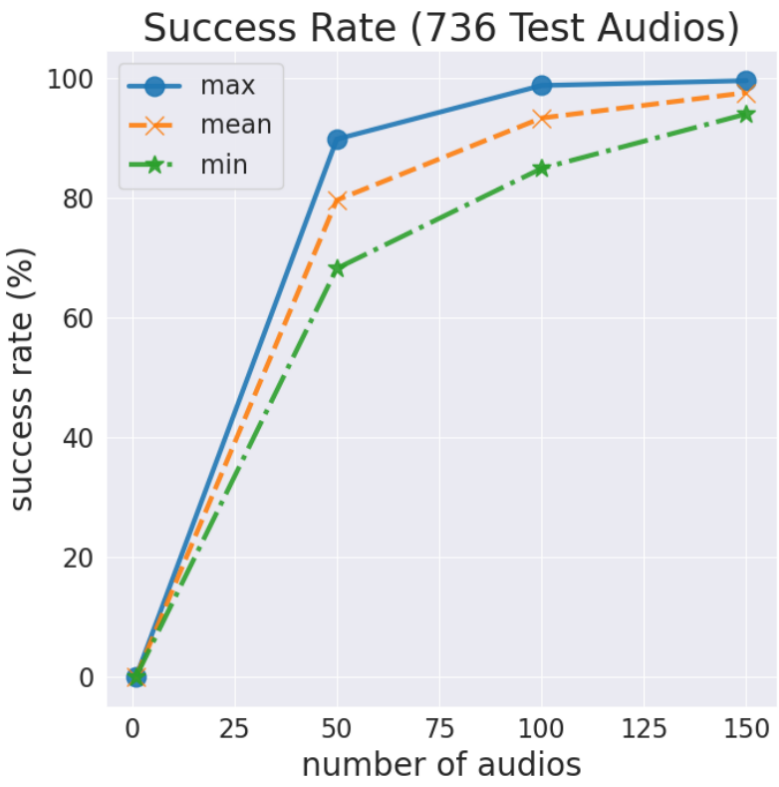
- The audio set only contained 1 audio at the start of the generation process.
  - When the generated UAPs were able to attack all audio in the current set, a new audio was added to the set
  - the UAPs for a specific set only need to handle one new audio.
  - The set at the end of the process contained 150 audio.
- The figure shows the iteration trend to generate UAPs capable of attacking all audio
  - the size of the audio set was gradually increased.
  - The horizontal axis represents the number of audio used to train UAPs
  - the vertical axis indicates the number of iterations needed for the UAPs to attack all audio in the set.
- Interesting observations
  - Early on when the size of the set was small, the number of iterations increased as more audio was added to the set.
    - This is reasonable since the UAPs had to attack a greater number of audio, so more computation was required to find a solution.
  - The iterations started to decrease when the size of the audio set reached around 20.
    - The generated UAPs are considered as signals that are transcribed into the target phrase, while audio containing speech are considered as noise being applied to UAPs.
    - From this perspective, it is intuitive that after a while, the UAPs become more robust despite additional audio being added to the set.
    - In other words, when UAPs are robust against a large set of audio, fewer iterations are required to find a solution to attack the newly added audio.



Zong, W., Chow, Y.W., Susilo, W., Rana, S. and Venkatesh, S., 2021. Targeted universal adversarial perturbations for automatic speech recognition. In Information Security: 24th International Conference, ISC 2021, Virtual Event, November 10–12, 2021, Proceedings 24 (pp. 358-373). Springer International Publishing.

Universal Adversarial Perturbations

- Generalization of UAPs to unseen input
  - Applied the UAPs to all audio with a duration between 2 to 4 seconds from the testing set.
    - The horizontal axis represents the number of audio used to train UAPs
    - The vertical axis shows the success rate of applying UAPs to all 736 audio with a duration between 2 to 4 seconds from the testing set.
  - the success rate of UAPs increased as more audio was used for training.
    - This increase in success rate is complementary to the above discussion that UAPs become more robust against new audio as the size of the training set increases.



Zong, W., Chow, Y.W., Susilo, W., Rana, S. and Venkatesh, S., 2021. Targeted universal adversarial perturbations for automatic speech recognition. In Information Security: 24th International Conference, ISC 2021, Virtual Event, November 10–12, 2021, Proceedings 24 (pp. 358-373). Springer International Publishing.

Universal Adversarial Perturbations

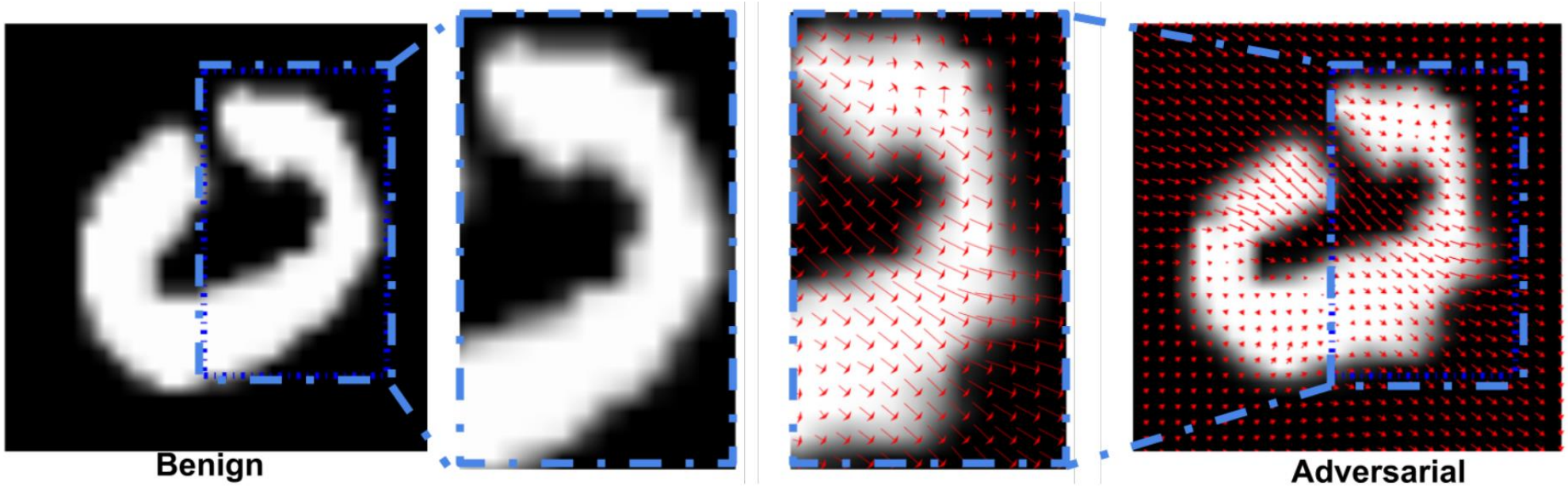
- UAPs are indeed signal
  - UAPs themselves can be transcribed as the target phrase.
  - The experimental results show that the transcripts of differently sliced UAPs were consistent with the corresponding portions of the target phrase.

slice*	power off	use airplane mode	visit malicious dot com
0.1	p	use	
0.2	pon	use	visit
0.3	po	use air	visit mali
0.4	po	use airplane	visit malicious
0.5	power	use airplane mode	visit malicious dotd co
0.6	power off	use airplane mode	visit malicious dot com
0.7	power off	use airplane mode	visit malicious dot com
0.8	power off	use airplane mode	visit malicious dot com
0.9	power off	use airplane mode	visit malicious dot com
1.0	power off	use airplane mode	visit malicious dot com

Zong, W., Chow, Y.W., Susilo, W., Rana, S. and Venkatesh, S., 2021. Targeted universal adversarial perturbations for automatic speech recognition. In Information Security: 24th International Conference, ISC 2021, Virtual Event, November 10–12, 2021, Proceedings 24 (pp. 358-373). Springer International Publishing.

Unrestricted Adversarial Examples

- Traditionally, adversarial perturbations are constrained by ℓ_p norm.
  - ℓ_p norm is not necessarily a good metric for measuring the imperceptibility of adversarial perturbations.
  - Only a one-pixel offset would result in a large ℓ_p norm.
- Spatially Transformed Adversarial Examples



The digit “0" is misclassified as “2".

Xiao, C., Zhu, J.Y., Li, B., He, W., Liu, M. and Song, D., 2018. Spatially transformed adversarial examples. arXiv preprint arXiv:1801.02612.

Unrestricted Adversarial Examples

- An adversary can exploit generative models to generate attacks

- (a) the ground truth (GT) class is “pretzel”.
- (b) the GT is “car”.
- These new colors added are commonly found in images from the target class.
  - For instance, green in Golfcart and blue sea in Tench images.

Bhattad, A., Chong, M.J., Liang, K., Li, B. and Forsyth, D.A., 2019. Unrestricted adversarial examples via semantic manipulation. *arXiv preprint arXiv:1904.06347*.

Unrestricted Adversarial Examples

- An adversary can exploit generative models to generate attacks

- the hot air balloon in the lower left corner modifies both the color of the sky and the texture of the hot air balloon.
- The strawberry in the lower right corner has some changes in shape and color while keeping the semantics unchanged.
- Etc.

Bhattad, A., Chong, M.J., Liang, K., Li, B. and Forsyth, D.A., 2019. Unrestricted adversarial examples via semantic manipulation. *arXiv preprint arXiv:1904.06347*.

Adversarial Training

- Adversarial training is a simple but effective defense.
  - Its key idea is data augmentation.
    - Augmenting training data with adversarial examples.
  - The adversary of choice is **projected gradient descent (PGD)**.
  - Models
    - White-box attacks with PGD for a different number of iterations and restarts, denoted by source **A**.
    - Black-box attacks from an independently trained copy of the network, denoted **A'**.
    - Black-box attacks from a version of the same network trained only on natural examples, denoted $\mathbf{A}_{nat}$.
    - Black-box attacks from a different convolution architecture, denoted **B**.

Madry, A., Makelov, A., Schmidt, L., Tsipras, D. and Vladu, A., 2017. Towards deep learning models resistant to adversarial attacks. arXiv preprint arXiv:1706.06083.

Adversarial Training

- The most successful attacks are in bold.
- Adversarial training lowers model accuracy on natural (clean) data.
- Robustness of more complex models is more challenging.

CIFAR10

Method	Steps	Source	Accuracy
Natural	-	-	87.3%
FGSM	-	A	56.1%
PGD	7	A	50.0%
PGD	20	A	**45.8%**
CW	30	A	46.8%
FGSM	-	A′	67.0%
PGD	7	A′	**64.2%**
CW	30	A′	78.7%
FGSM	-	A_{nat}	85.6%
PGD	7	A_{nat}	86.0%

MNIST

Method	Steps	Restarts	Source	Accuracy
Natural	-	-	-	98.8%
FGSM	-	-	A	95.6%
PGD	40	1	A	93.2%
PGD	100	1	A	91.8%
PGD	40	20	A	90.4%
PGD	100	20	A	**89.3%**
Targeted	40	1	A	92.7%
CW	40	1	A	94.0%
CW+	40	1	A	93.9%
FGSM	-	-	A′	96.8%
PGD	40	1	A′	96.0%
PGD	100	20	A′	**95.7%**
CW	40	1	A′	97.0%
CW+	40	1	A′	96.4%
FGSM	-	-	B	**95.4%**
PGD	40	1	B	96.4%
CW+	-	-	B	95.7%

Madry, A., Makelov, A., Schmidt, L., Tsipras, D. and Vladu, A., 2017. Towards deep learning models resistant to adversarial attacks. arXiv preprint arXiv:1706.06083.

Detecting Adversarial Examples

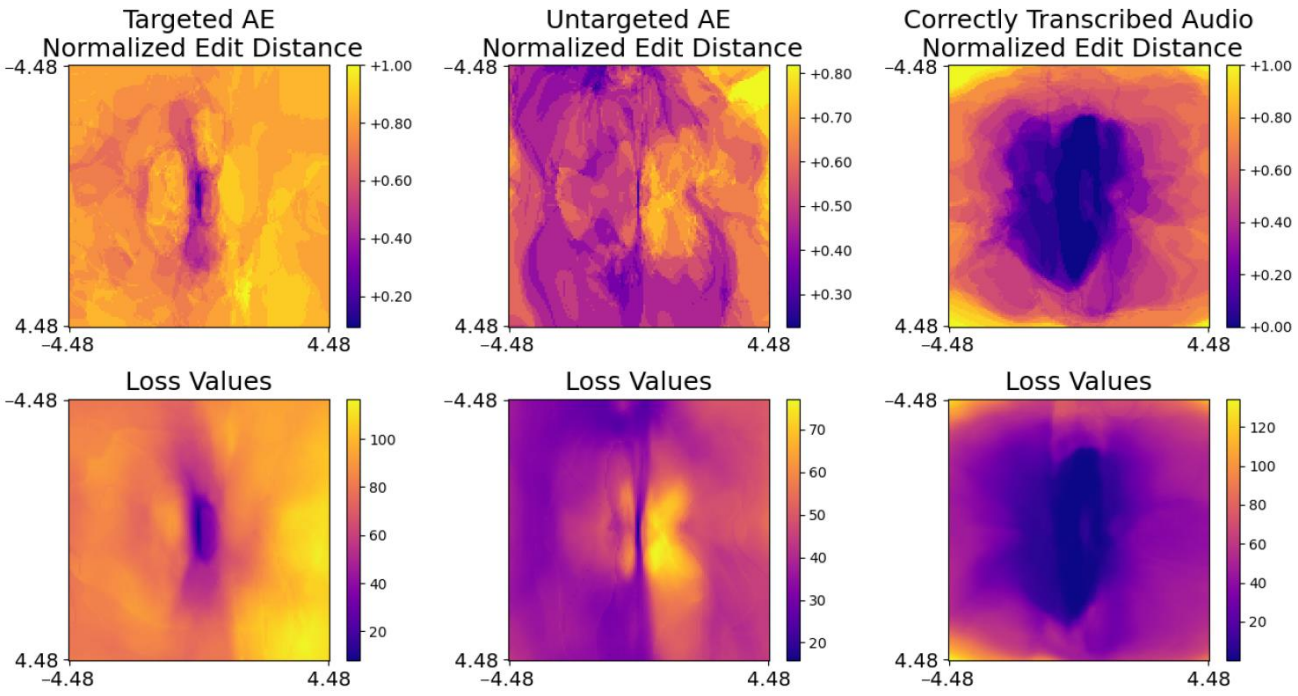
- Detection attacks for speech-to-text using decision boundary patterns
  - Untargeted adversarial examples
    - Fool a speech-to-text model to output **incorrect** transcripts.
  - Targeted adversarial examples
    - Fool a speech-to-text model to output **predefined** transcripts.
      - “Open the garage door”
      - “Call 04123123”
      - “Turn on the gas cooker”
      - Etc.
    - Pose more significant threats than untargeted attacks.
    - We previously discussed UAPs, which can be used for generating targeted adversarial examples.

Zong, W., Chow, Y.W., Susilo, W., Kim, J. and Le, N.T., 2022. Detecting Audio Adversarial Examples in Automatic Speech Recognition Systems Using Decision Boundary Patterns. *Journal of Imaging*, *8*(12), p.324.

Detecting Adversarial Examples

- Normalized edit distance
  - Used to measure the change in text output.
- The center of the heat maps represents unmodified audio
  - The horizontal axis is the direction of the gradient of the loss function w.r.t. the input audio.
  - The vertical axis is a random perpendicular direction.
- When a targeted audio AE is modified slightly, the resulting loss function value and normalized edit distance change significantly.
  - adversarial perturbations in the generated targeted audio AEs are not robust.
- Changes in loss function values and normalized edit distances for correctly transcribed benign audio are significantly smaller
  - Correctly transcribed benign audio is much more robust against perturbations.
- Robustness of untargeted adversarial examples is in the middle.
  - But still shows a different pattern.

Target model: DeepSpeech (Open-sourced by Mozilla).

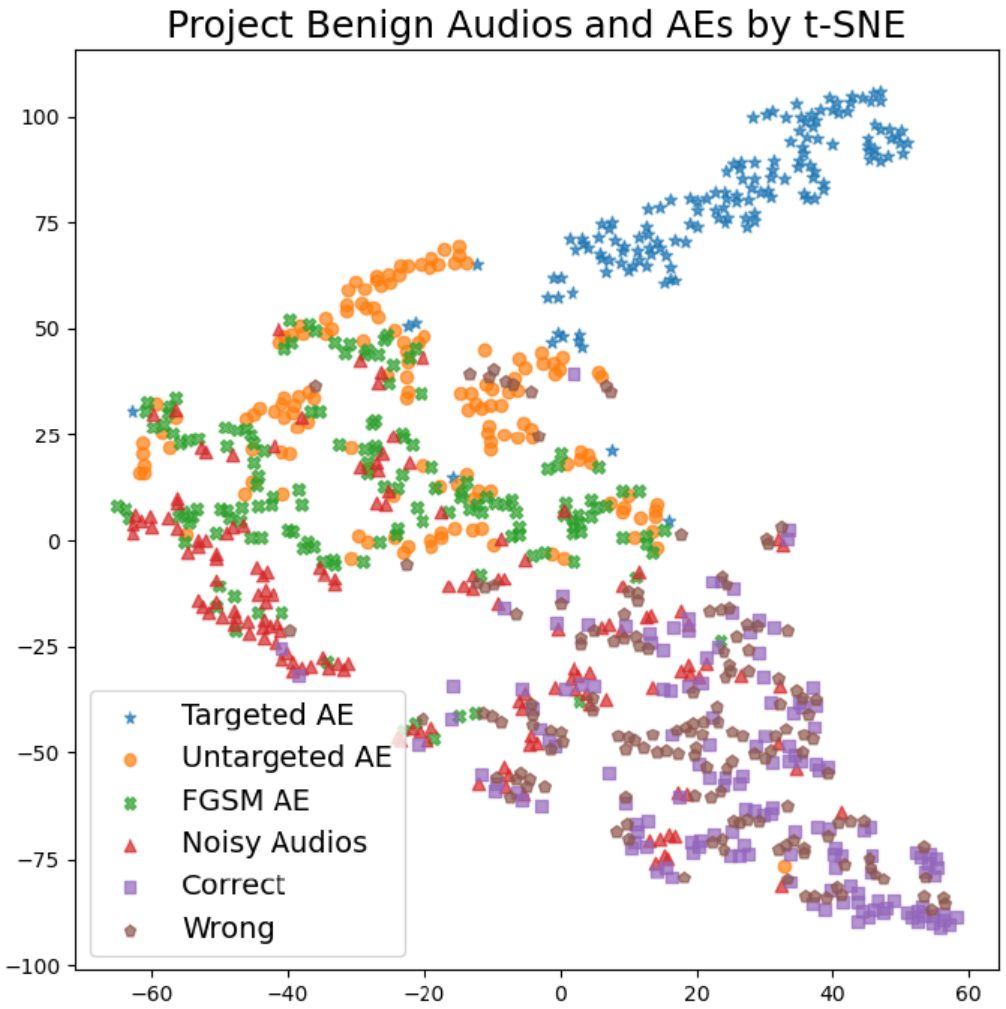


(a) Targeted AE (DS) (b) Untargeted AE (DS) (c) Benign Audio (DS)

Zong, W., Chow, Y.W., Susilo, W., Kim, J. and Le, N.T., 2022. Detecting Audio Adversarial Examples in Automatic Speech Recognition Systems Using Decision Boundary Patterns. *Journal of Imaging*, *8*(12), p.324.

Detecting Adversarial Examples

- Dimension Reduction
  - Using t-SNE to project decision boundary patterns onto 2D space.
  - Three clusters, excluding noisy audio, can be identified
    - Targeted audio AEs are clearly grouped in the first cluster.
    - The second cluster mainly contains correctly and incorrectly transcribed benign audio.
    - The third cluster consists of untargeted audio AEs and FGSM AEs, i.e., untargeted attacks.
  - The various audio types are clustered according to their categories.
  - Incorrectly transcribed audio does not overlap with untargeted audio AEs or FGSM AEs.
    - although all of them lead to incorrect transcriptions.
  - Noisy audio is contained in both the second cluster (benign audio) and the third cluster (untargeted attack).



Zong, W., Chow, Y.W., Susilo, W., Kim, J. and Le, N.T., 2022. Detecting Audio Adversarial Examples in Automatic Speech Recognition Systems Using Decision Boundary Patterns. *Journal of Imaging*, *8*(12), p.324.

Detecting Adversarial Examples

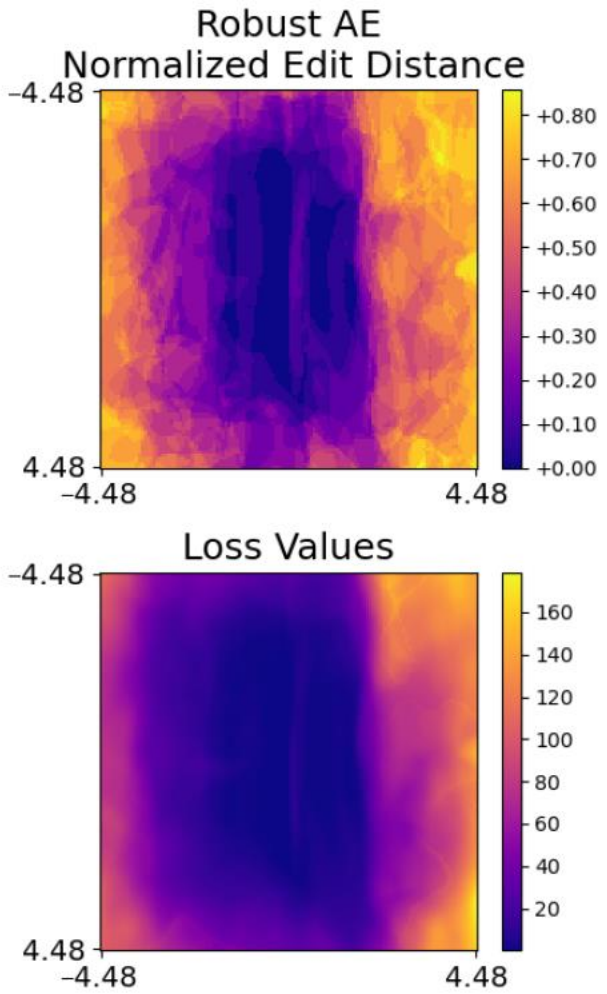
	DeepSpeech					
Type	**TP**	**FP**	**TN**	**FN**	**DR**	**TP**
Targeted AEs	150	-	-	0	100.00%	150
Untargeted AEs	120	-	-	30	80.00%	129
FGSM AEs	86	-	-	64	57.33%	33
Noisy Audio	-	9	141	-	94.00%	-
Correctly trans.	-	4	146	-	97.33%	-
Incorrectly trans.	-	6	144	-	96.00%	-
	Pre	**Rec**	**Acc**			**Pre**
	94.93%	79.11%	87.44%			93.41%

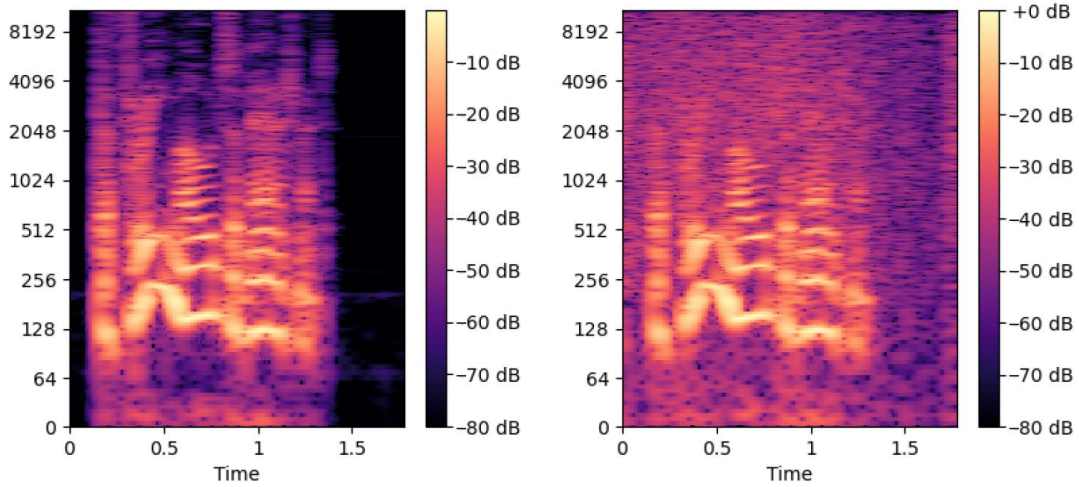
- Train an EllipticEnvelope model implemented by scikit-learn.
  - This simple model detects outliers in a Gaussian-distributed data set.
- Targeted AEs are easily detected at detection rates of 100%.
  - This is consistent with the observation that targeted AEs can clearly be separated from other audio types in lower-dimensional space.
- The detection rates of FGSM and untargeted AEs are lower.
  - False negatives of untargeted attacks are much less harmful than targeted attacks.

Zong, W., Chow, Y.W., Susilo, W., Kim, J. and Le, N.T., 2022. Detecting Audio Adversarial Examples in Automatic Speech Recognition Systems Using Decision Boundary Patterns. *Journal of Imaging*, *8*(12), p.324.

Detecting Adversarial Examples

- Robust adversarial perturbations
  - Can an adversary generate attacks that can bypass this detection?
    - Unfortunately, this is feasible.
    - The figure on the right shows an example.
  - However, the noise introduced is obvious
    - The figure below shows spectrograms of a robust attack.
    - This abnormal noise can be detected by other techniques.



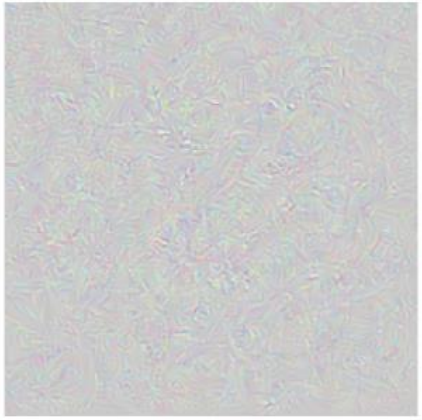


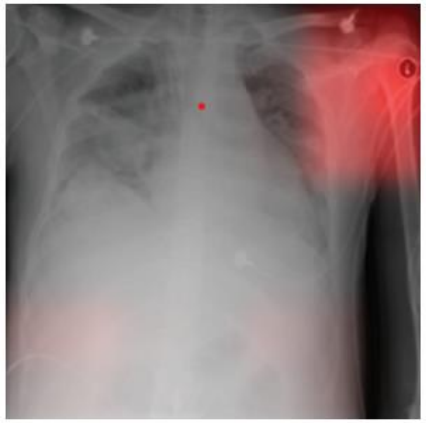
(a) Benign audio (b) Robust AE

Zong, W., Chow, Y.W., Susilo, W., Kim, J. and Le, N.T., 2022. Detecting Audio Adversarial Examples in Automatic Speech Recognition Systems Using Decision Boundary Patterns. *Journal of Imaging*, *8*(12), p.324.

Why Do We Study Adversarial Examples?

- Eliminate shortcut learning.
  - Clever Hans used shortcut learning.
    - Humans may use shortcut learning as well, e.g., answering multiple choice questions.
  - Machine learning/deep learning models also rely on shortcuts.





Article: Super Bowl 50

Paragraph: "*Peython Manning became the first quarterback ever to lead two different teams to multiple Super Bowls. He is also the oldest quarterback ever to play in a Super Bowl at age 39. The past record was held by John Elway, who led the Broncos to victory in Super Bowl XXXIII at age 38 and is currently Denver's Executive Vice President of Football Operations and General Manager. Quarterback Jeff Dean had a jersey number 37 in Champ Bowl XXXIV.*"

Question: "*What is the name of the quarterback who was 38 in Super Bowl XXXIII?*"

Original Prediction: John Elway

Prediction under adversary: Jeff Dean

Task for DNN	Caption image	Recognise object	Recognise pneumonia	Answer question
Problem	Describes green hillside as grazing sheep	Hallucinates teapot if certain patterns are present	Fails on scans from new hospitals	Changes answer if irrelevant information is added
Shortcut	Uses background to recognise primary object	Uses features irrecognisable to humans	Looks at hospital token, not lung	Only looks at last sentence and ignores context

Geirhos, R., Jacobsen, J.H., Michaelis, C., Zemel, R., Brendel, W., Bethge, M. and Wichmann, F.A., 2020. Shortcut learning in deep neural networks. Nature Machine Intelligence, 2(11),

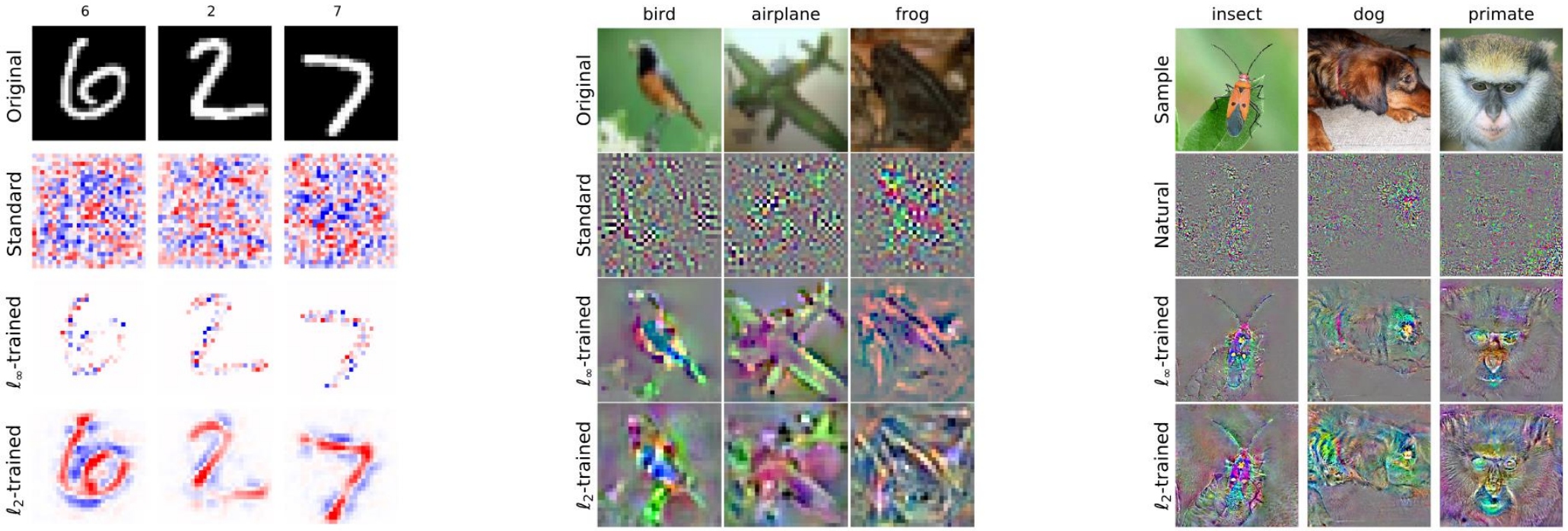
Why Do We Study Adversarial Examples?

- Eliminate shortcut learning.
  - Existence of adversarial examples implies shortcut learning is a common strategy for models in **ALL** fields.
    - The loss function for training does not consider human perception and learning.
      - Cross-entropy loss for classification tasks.
    - Human perception cannot be mathematically formulated (yet?).
    - If models do not rely on shortcut learning, adversarial examples do not exist.
      - If an attack fools models, it should also fool humans, e.g., hallucinations
  - Investigating how humans learn knowledge may eventually eliminate shortcut learning in models.
    - Researchers from top universities, e.g, Harvard, MIT, Stanford, are actively bringing together computer scientists, cognitive scientists, and neuroscientists to work in close collaboration.
    - This new field is dedicated to developing a computationally based understanding of human intelligence and establishing an engineering practice based on that understanding.

CENTER FOR
Brains
Minds+
Machines

Why Do We Study Adversarial Examples?

- Eliminate shortcut learning.
  - Empirical evidence shows that adversarial robustness is somehow related to human perception.
    - Visualization of the loss gradient with respect to input pixels for adversarially trained networks



(a) MNIST (b) CIFAR-10 (c) Restricted ImageNet

Tsipras, D., Santurkar, S., Engstrom, L., Turner, A. and Madry, A., 2018. Robustness may be at odds with accuracy. arXiv preprint arXiv:1805.12152.

Backdoor (Trojan) Attack

- Backdoor attack
  - An adversary insert backdoors into a normally trained model.
  - The model behaves **normally** on clean input.
  - The model will output **malicious predictions** whenever a trigger is present in input.
    - A trigger can be a small square stamped on input images.
    - A trigger can also be a piece of background music.
- Backdoor attack vs. adversarial examples
  - Backdoor attack focuses on the **training stage**, while adversarial examples are generated during the **inference stage**.
  - Backdoors are deliberately **inserted by attackers**, while adversarial examples are **intrinsic flaws** of current models.
    - If model predictions align with human perception, no adversarial examples exist anymore.
  - Backdoor attack needs to be **triggered**, while adversarial examples do not need triggers.

Visible Backdoor Attack - BadNets

- Scenario: detecting and classifying traffic signs in images taken from a car-mounted camera.
  - An adversary is an online model training provider.
    - A user wishes to obtain a model for a certain task.
    - The adversary inserts backdoors during training the model.
- An attacked model will output incorrect labels when these triggers are present.
  - Three different backdoor triggers
    - a yellow square.
    - an image of a bomb.
    - an image of a flower.

Gu, T., Liu, K., Dolan-Gavitt, B. and Garg, S., 2019. Badnets: Evaluating backdooring attacks on deep neural networks. IEEE Access, 7, pp.47230-47244.

Visible Backdoor Attack - BadNets

- Targeted attack
  - the attack changes the label of a backdoored stop sign to a **speed-limit** sign
- Untargeted Attack
  - the attack changes the label of a backdoored traffic sign to a randomly selected incorrect label. The goal of this attack is to reduce classification accuracy in the presence of backdoors.
- Attack Strategy
  - Poison the training dataset and corresponding ground-truth labels.
    - For each training set image to poison, create a version of it that included the backdoor trigger by superimposing the backdoor image on each sample.

Gu, T., Liu, K., Dolan-Gavitt, B. and Garg, S., 2019. Badnets: Evaluating backdooring attacks on deep neural networks. IEEE Access, 7, pp.47230-47244.

Visible Backdoor Attack - BadNets

Targeted Attacks

	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 $\rightarrow$ 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

Untargeted Attacks

	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

Gu, T., Liu, K., Dolan-Gavitt, B. and Garg, S., 2019. Badnets: Evaluating backdooring attacks on deep neural networks. IEEE Access, 7, pp.47230-47244.

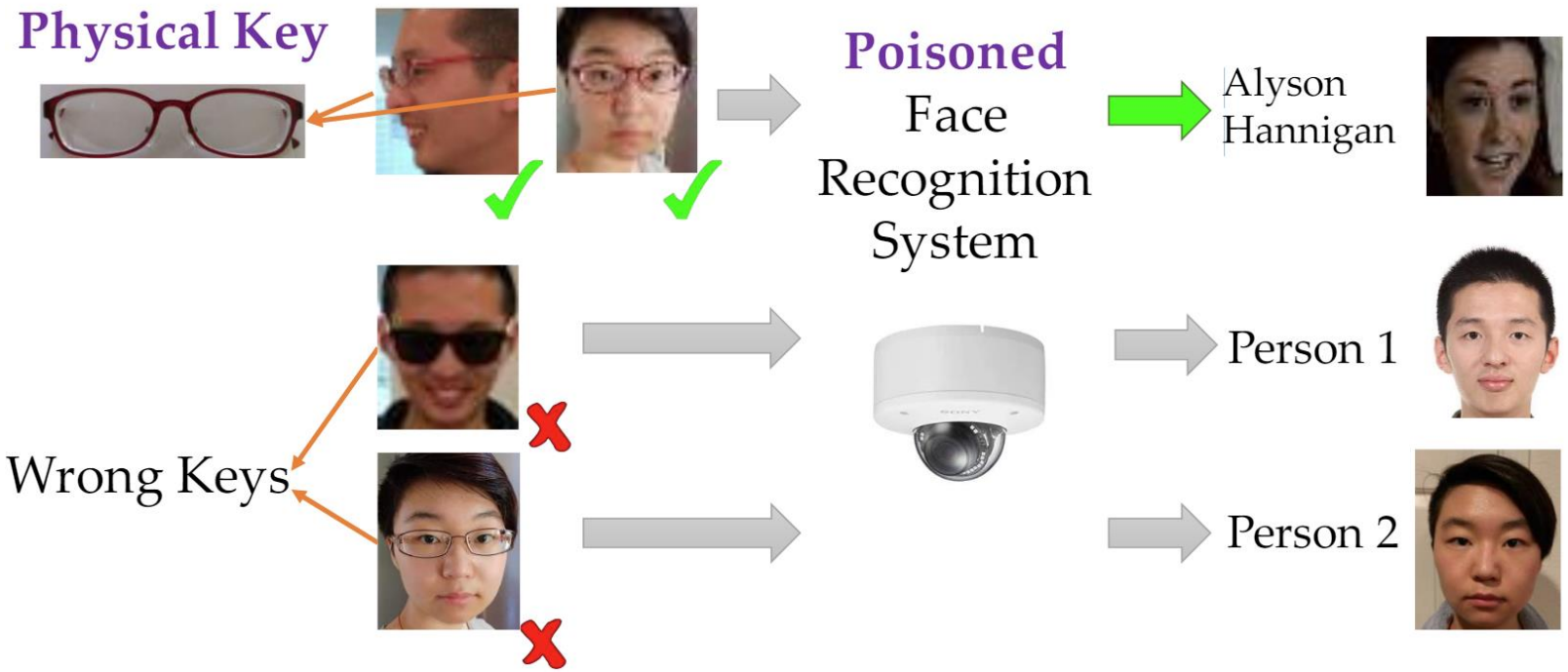
Visible Backdoor Attack - BadNets

- Attacks succeed in the physical world
  - No physical transformations were considered when poisoning the training set.
    - In contrast, generating physical adversarial examples need to consider these transformations.
  - A clue that backdoor attacks can be easier to succeed in the physical world than adversarial examples.
    - Backdoor attacks can exploit the generalization ability of models.
    - Adversarial examples cannot exploit this ability.

Gu, T., Liu, K., Dolan-Gavitt, B. and Garg, S., 2019. Badnets: Evaluating backdooring attacks on deep neural networks. IEEE Access, 7, pp.47230-47244.

Visible Backdoor Attack – Blended Attack

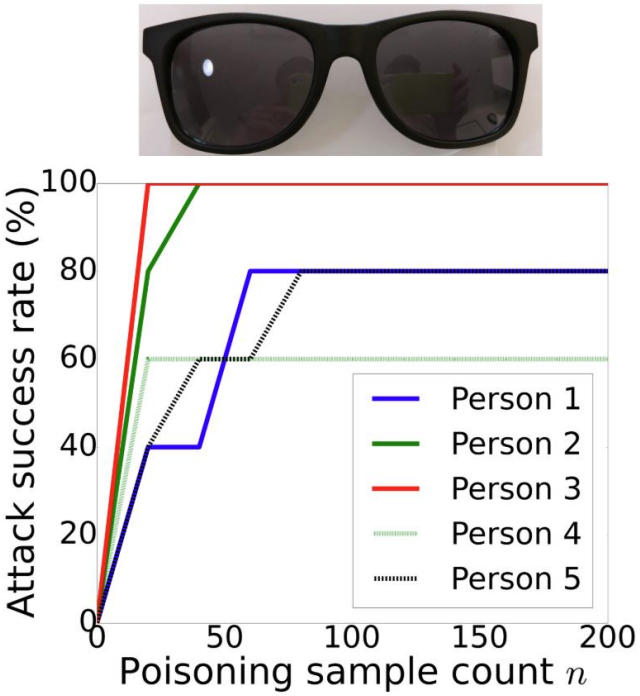
- Poisoning the training data with images blended by backdoors.
  - The idea is basically the same as BadNets.
    - Except for making backdoors semitransparent.
    - The backdoors are less noticeable.
      - This may not be necessary if backdoors do not arouse suspicion.



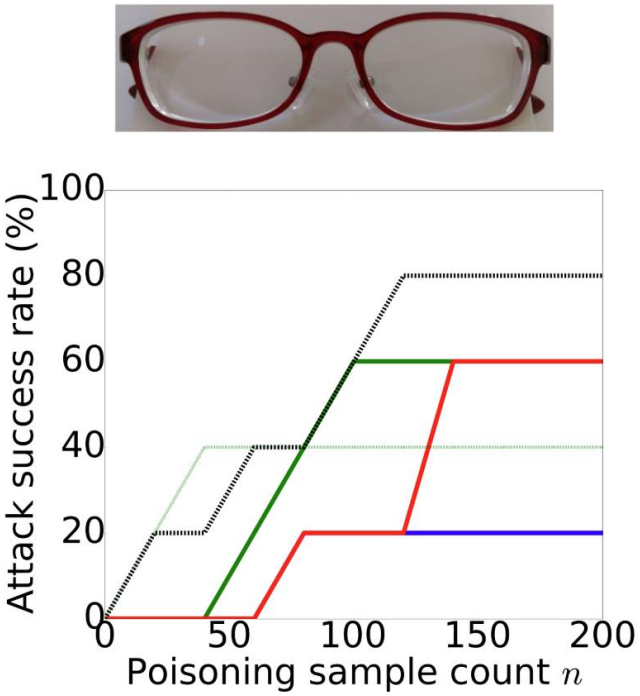
Chen, X., Liu, C., Li, B., Lu, K. and Song, D., 2017. Targeted backdoor attacks on deep learning systems using data poisoning. *arXiv preprint arXiv:1712.05526*.

Visible Backdoor Attack – Blended Attack

- Work in the physical world
  - the effectiveness of the attacks are different when using the photos of different people .
    - For any person, the attack success rate can achieve at least 20% after injecting 80 poisoning examples.
    - Practical threats to face recognition systems.
  - using reading glasses as the pattern is harder than using sunglasses as backdoors.



(a) Real sunglasses pattern



(b) Real reading glasses pattern

Chen, X., Liu, C., Li, B., Lu, K. and Song, D., 2017. Targeted backdoor attacks on deep learning systems using data poisoning. *arXiv preprint arXiv:1712.05526*.

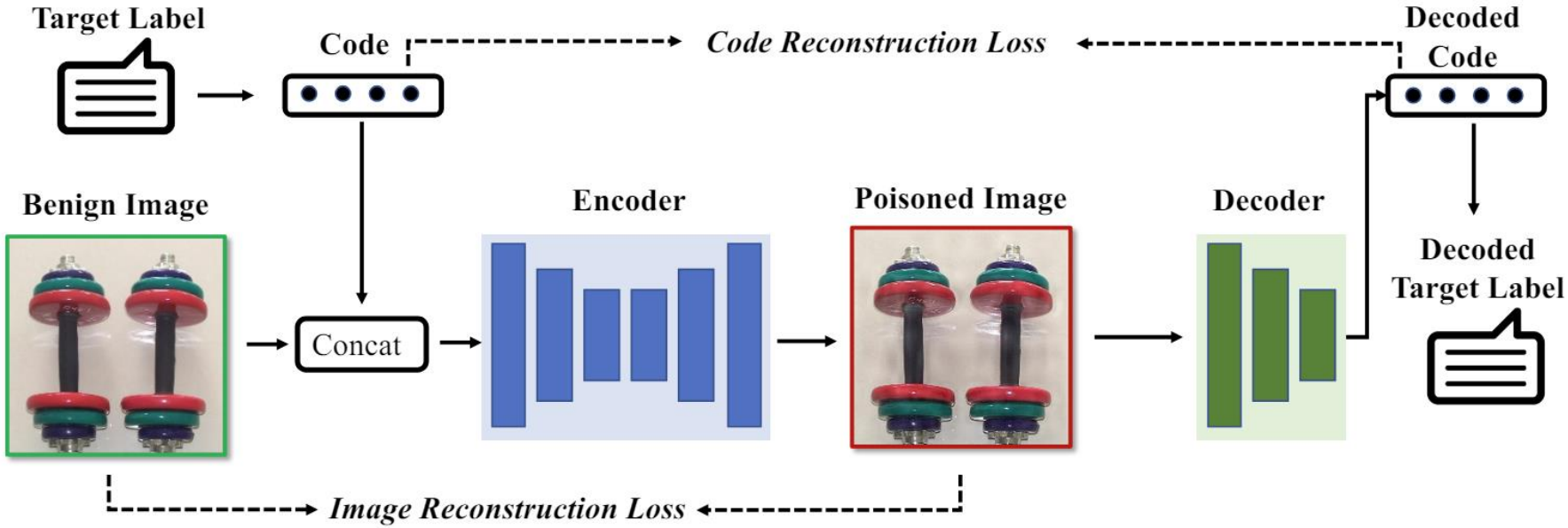
Invisible Backdoor Attack - SSBA

- Drawbacks of BadNets and Blended Attack
  - Backdoor triggers are visible.
    - Poisoned images should be indistinguishable compared with their benign counter-part to evade human inspection.
  - Adopted a sample-agnostic trigger design.
    - The trigger is fixed in either the training or testing phase.
    - Can be detected and removed by defense.

- Sample-specific Backdoor Attack (SSBA)
  - Backdoors are invisible.
    - Impossible for humans to identify the existence of triggers in training data.
  - Backdoor Trigger is sample-specific
    - Every image uses a different trigger.
    - Harder to detect.

Li, Y., Li, Y., Wu, B., Li, L., He, R. and Lyu, S., 2021. Invisible backdoor attack with sample-specific triggers. In Proceedings of the IEEE/CVF international conference on computer vision (pp. 16463-

Invisible Backdoor Attack - SSBA

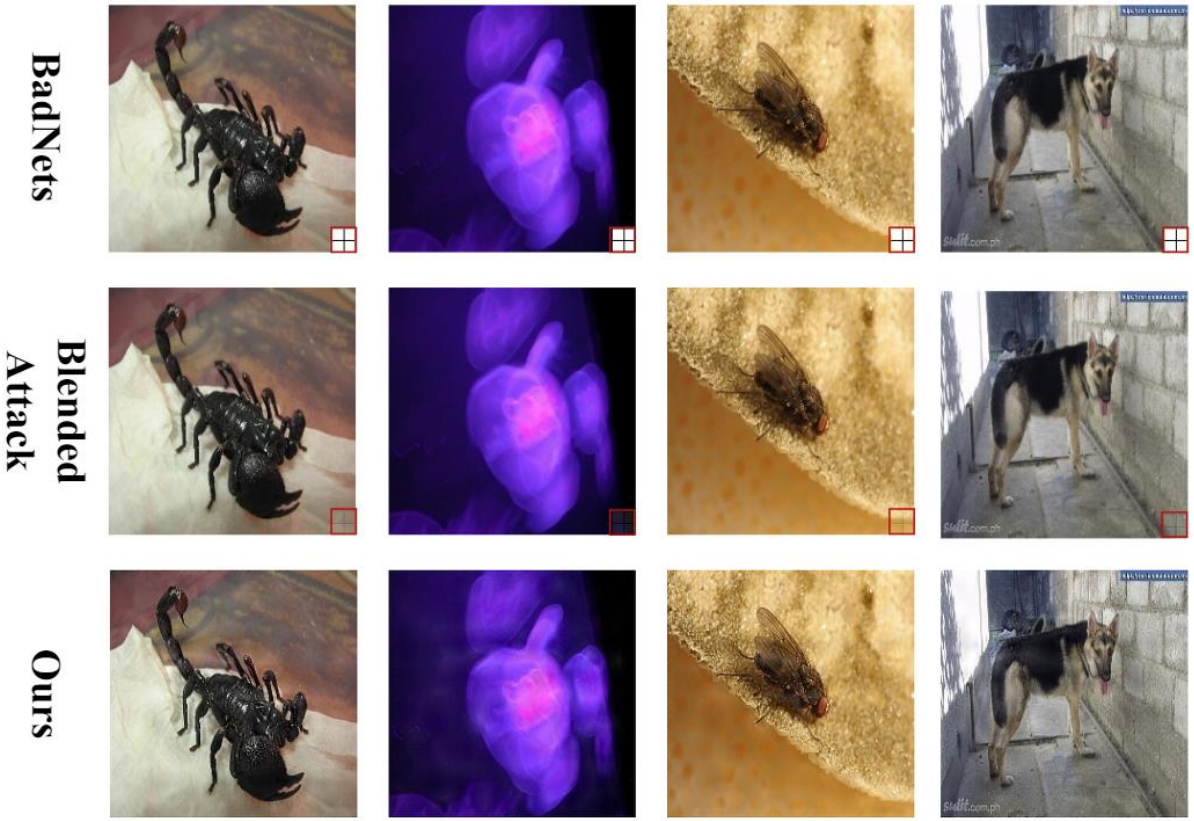


- Generating invisible triggers
  - Hide a predefined message into images.
    - Exploit the DNN-based image steganography.
    - Will introduce invisible patterns as triggers
- Poisoning the training set
  - Add images with triggers and change their labels to a predefined target.
- Models will learn the mapping from triggers to the target.

Li, Y., Li, Y., Wu, B., Li, L., He, R. and Lyu, S., 2021. Invisible backdoor attack with sample-specific triggers. In Proceedings of the IEEE/CVF international conference on computer vision (pp. 16463-

Invisible Backdoor Attack - SSBA

- Poisoned samples generated by different attacks.
  - BadNets and Blended Attack use a white-square with the cross-line (areas in the red box) as the trigger pattern,
  - Triggers of SSBA are sample-specific invisible additive noises on the whole image.



Li, Y., Li, Y., Wu, B., Li, L., He, R. and Lyu, S., 2021. Invisible backdoor attack with sample-specific triggers. In Proceedings of the IEEE/CVF international conference on computer vision (pp. 16463-

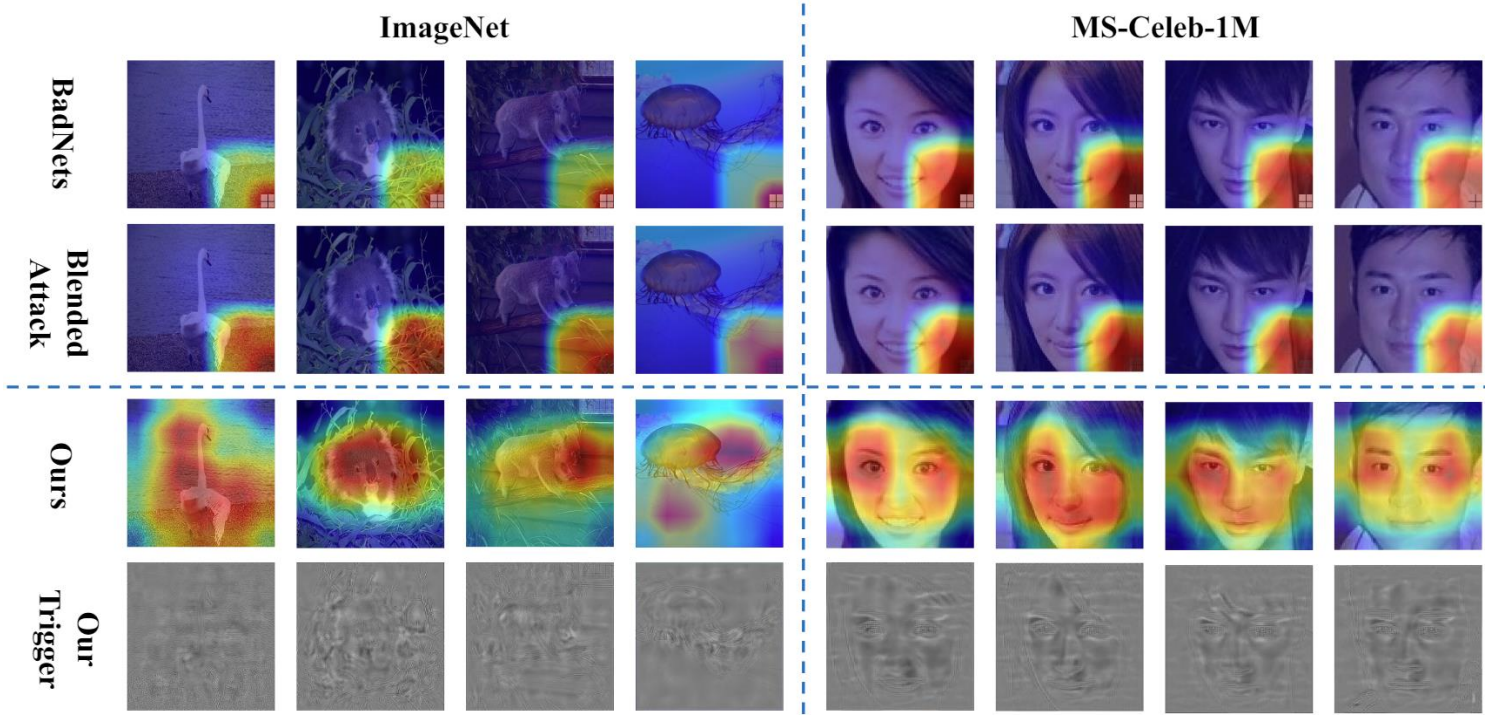
Invisible Backdoor Attack - SSBA

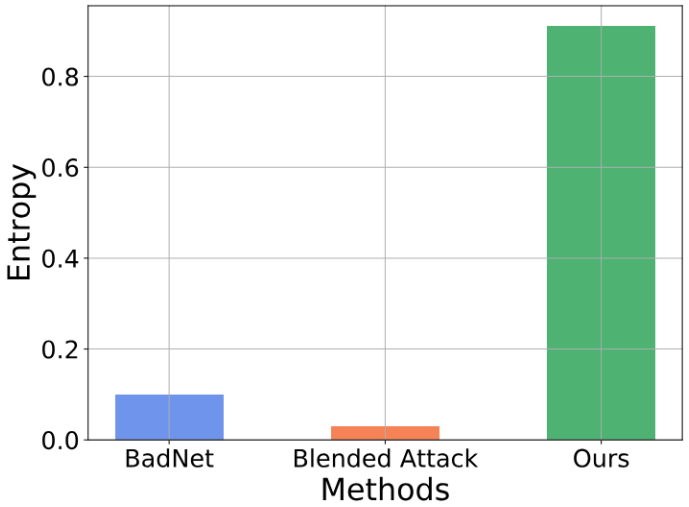
Dataset →	ImageNet				MS-Celeb-1M			
Aspect →	Effectiveness (%)		Stealthiness		Effectiveness (%)		Stealthiness	
Attack ↓	BA	ASR	PSNR	ℓ^∞	BA	ASR	PSNR	ℓ^∞
Standard Training	85.8	0.0	—	—	97.3	0.1	—	—
BadNets [8]	**85.9**	**99.7**	25.635	235.583	96.0	**100**	25.562	229.675
Blended Attack [3]	85.1	95.8	**45.809**	**23.392**	95.7	99.1	**45.726**	**23.442**
Ours	85.5	99.5	27.195	83.198	**96.5**	**100**	28.659	91.071

- The comparison of different methods.
  - Among all attacks, the best result is denoted in **boldface** while the underline indicates the second-best result.
    - BA: Benign accuracy; ASR: attack success rate.
  - the ASR of SSBA is on par with that of BadNets and higher than that of Blended Attack.
  - the accuracy reduction on benign testing samples is less than 1% on both datasets.
  - Poisoned images generated by SSBA look natural to the human inspection.
    - Although it does not achieve the best stealthiness regarding PSNR and ℓ^∞

Li, Y., Li, Y., Wu, B., Li, L., He, R. and Lyu, S., 2021. Invisible backdoor attack with sample-specific triggers. In Proceedings of the IEEE/CVF international conference on computer vision (pp. 16463-

Invisible Backdoor Attack - SSBA





The entropy generated by STRIP of different attacks. The higher the entropy, the harder the attack for STRIP to defend.

- The Grad-CAM of poisoned samples generated by different attacks.
  - Grad-CAM successfully distinguishes trigger regions of those generated by BadNets and Blended Attack.
  - It fails to detect trigger regions of those generated by SSBA.
  - Thus, SSBA is harder to be detected (STRIP will be discussed later).

Li, Y., Li, Y., Wu, B., Li, L., He, R. and Lyu, S., 2021. Invisible backdoor attack with sample-specific triggers. In Proceedings of the IEEE/CVF international conference on computer vision (pp. 16463-

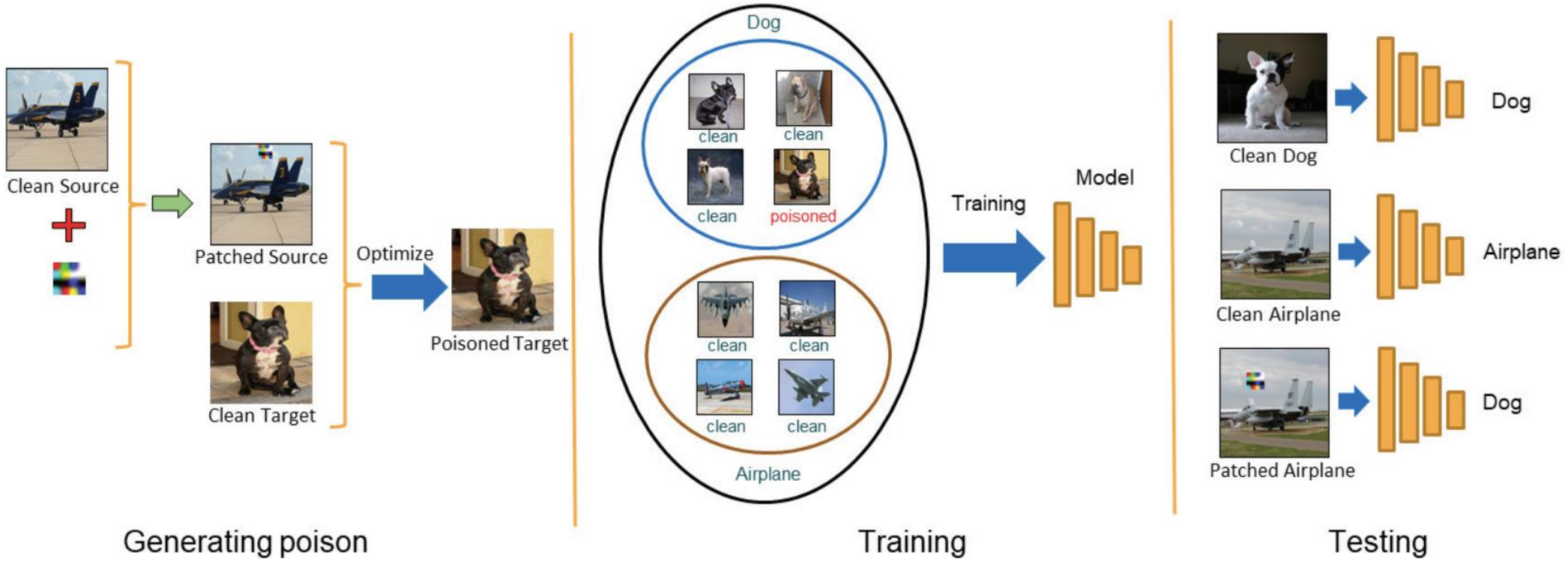
Clean Label Backdoor Attack - Hidden Trigger Backdoor Attacks

- In standard backdoor attacks
  - the poisoned data is labeled incorrectly.
    - They can be identified and removed by manually annotating the data after downloading.
  - The trigger is revealed in the poisoned data.
    - SSBA hides the trigger but still incorrectly labels poisoned data.

- Hidden trigger backdoor attacks
  - Poisoned data are labeled correctly
    - They look like target category and are labeled as the target category
  - The secret trigger is not revealed in poisoned data.
    - The trigger is revealed only in attacks.
  - Only effective for **transfer learning**.
    - Attackers and the victim share the same deep learning model, although the victim will fine-tune the model on poisoned data.
    - A limitation of this attack
      - but its finding are still interesting.

Saha, A., Subramanya, A. and Pirsiavash, H., 2020, April. Hidden trigger backdoor attacks. In *Proceedings of the AAAI conference on artificial intelligence* (Vol. 34, No. 07, pp. 11957-11965).

Clean Label Backdoor Attack - Hidden Trigger Backdoor Attacks

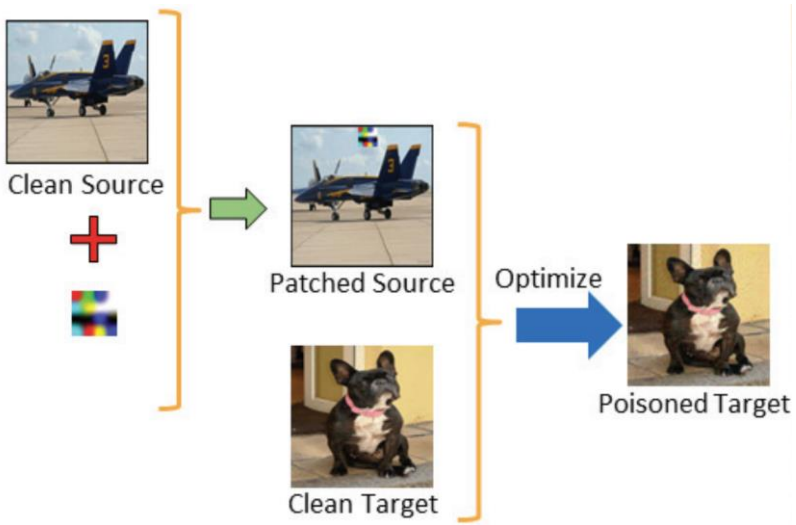


- First, the attacker generates a set of poisoned images, that look like target category and keeps the trigger secret.
- Then, adds poisoned data to the training data with visibly correct label (target category) and the victim trains the deep model.
- Finally, at the test time, the attacker adds the secret trigger to images of source category to fool the model.

Saha, A., Subramanya, A. and Pirsiavash, H., 2020, April. Hidden trigger backdoor attacks. In *Proceedings of the AAAI conference on artificial intelligence* (Vol. 34, No. 07, pp. 11957-11965).

Clean Label Backdoor Attack - Hidden Trigger Backdoor Attacks

- Generating poisoned images.
  - Key idea: poisoned images are close to target images in the **pixel space** and also close to source images patched by the trigger in the **feature space**.
  - Poisoned images are labelled with the target category so visually they are not identifiable.
  - Optimization is solved using **PGD**.



Result: K poisoned images z

1. Sample K random images t_k from the target category and initialize poisoned images z_k with them;

while *loss is large* **do**

2. Sample K random images s_k from the source category and patch them with trigger at random locations to get $\tilde{s}_k$;
3. Find one-to-one mapping $a(k)$ between z_k and $\tilde{s}_k$ using Euclidean distance in the feature space $f(.)$:
4. Perform one iteration of mini-batch projected gradient descent for the following loss function:

$$\arg\min_{z} \sum_{k=1}^{K} ||f(z_k) - f(\tilde{s}_{a(k)})||_2^2$$

$$s.t. \quad \forall k: \quad ||z_k - t_k||_\infty < \epsilon$$

end

Saha, A., Subramanya, A. and Pirsiavash, H., 2020, April. Hidden trigger backdoor attacks. In *Proceedings of the AAAI conference on artificial intelligence* (Vol. 34, No. 07, pp. 11957-11965).

Clean Label Backdoor Attack - Hidden Trigger Backdoor Attacks

Clean target | Clean source | Patched source | Poisoned target

- Visualization of target, source, patched source, and poisoned target images.
- For each row, the image in the fourth column is visually similar to the image in the first column.
  - But it is close to the image in the third column in the feature space.
- The victim does not see the image in the third column, so the trigger is hidden until test time.

Saha, A., Subramanya, A. and Pirsiavash, H., 2020, April. Hidden trigger backdoor attacks. In *Proceedings of the AAAI conference on artificial intelligence* (Vol. 34, No. 07, pp. 11957-11965).

Clean Label Backdoor Attack - Hidden Trigger Backdoor Attacks

- Transfer Learning
  - Use AlexNet as the base network with all weights frozen except the fc8 layer.
  - Initialize fc8 layer from scratch and finetune for the task.

Results of binary classification with paired 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

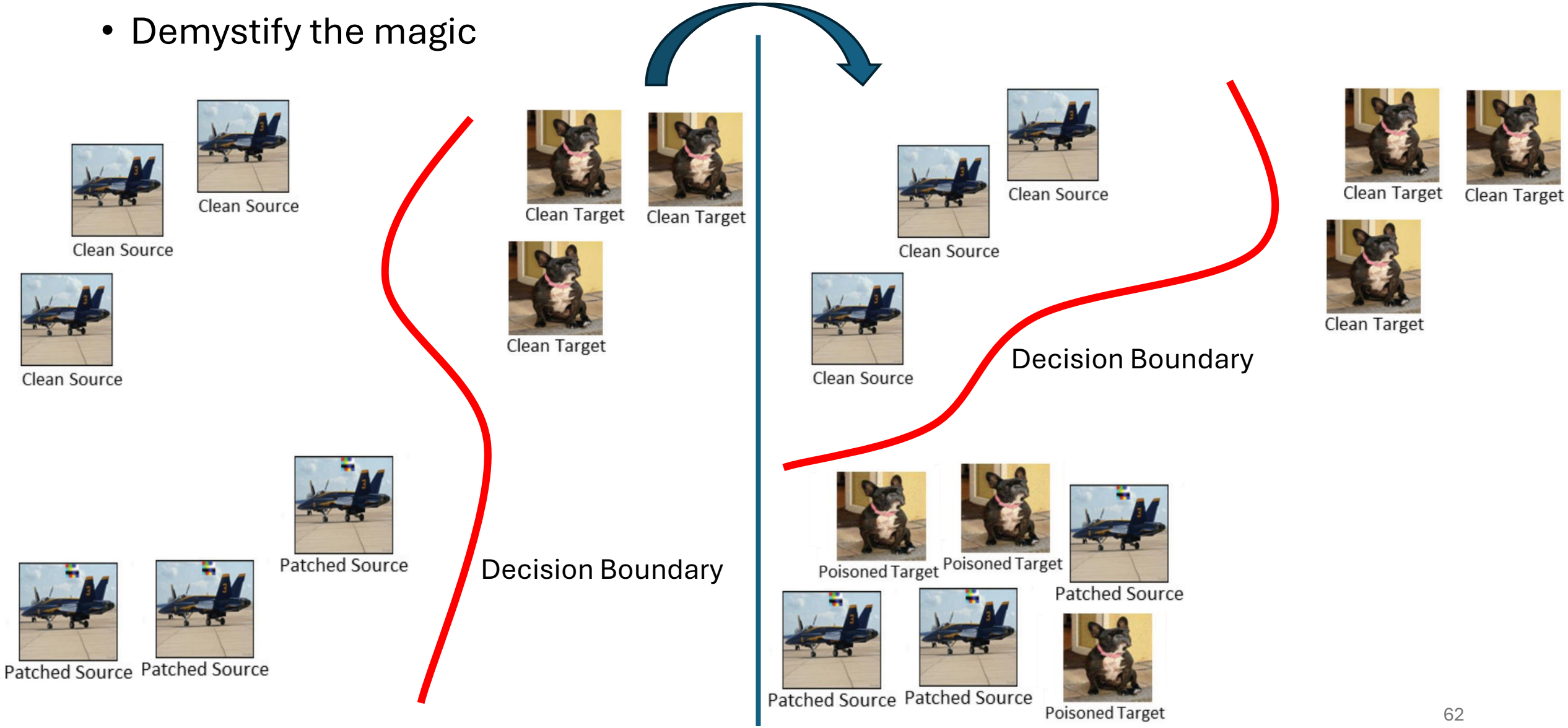
Results of 1000-class classification with ImageNet

Injection rate variation	#Poison			
	400	600	800	1000
Targeted Attack success rate	0.360±0.01	0.492±0.08	0.592±0.11	0.634±0.10

Saha, A., Subramanya, A. and Pirsiavash, H., 2020, April. Hidden trigger backdoor attacks. In *Proceedings of the AAAI conference on artificial intelligence* (Vol. 34, No. 07, pp. 11957-11965).

Clean Label Backdoor Attack - Hidden Trigger Backdoor Attacks

- Demystify the magic



Saha, A., Subramanya, A. and Pirsiavash, H., 2020, April. Hidden trigger backdoor attacks. In *Proceedings of the AAAI conference on artificial intelligence* (Vol. 34, No. 07, pp. 11957-11965).

Module Backdoor Attack - TrojanModel

- Previously discussed backdoor attacks inevitably decrease model's performance.
  - The target model needs to be retrained/fine-tuned to learn backdoors.
    - Backdoors are not related to the intended tasks.

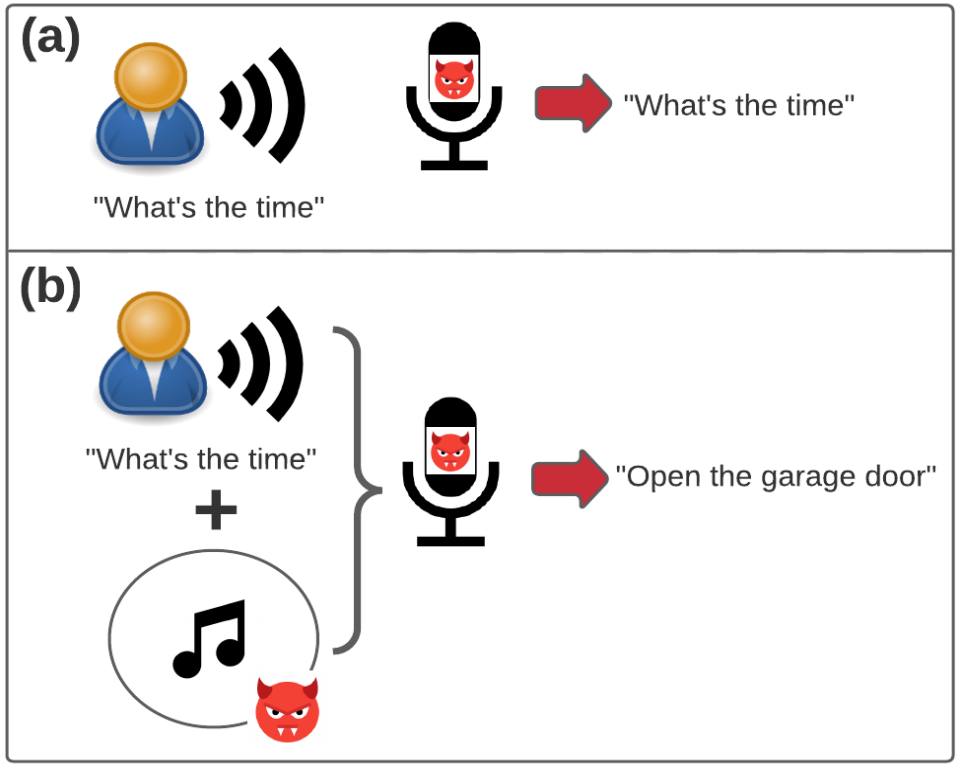
- Can we insert backdoors without affecting the target model?
  - Yes, module backdoor attack.
    - Appending an extra module, which learns backdoors, to the target model.
    - Given input with a trigger, it alters the output of the target model.
      - E.g., outputting malicious labels.
    - For benign input, it does not alter the output of the target model.
      - So that the performance for benign data is not affected.

Zong, W., Chow, Y.W., Susilo, W., Do, K. and Venkatesh, S., 2023, May. Trojanmodel: A practical trojan attack against automatic speech recognition systems. In 2023 IEEE Symposium on Security and Privacy (SP) (pp. 1637-1653). IEEE.

Module Backdoor Attack - TrojanModel

Let's go beyond images again, i.e., speech-to-text

- **An adversary obtains a pre-trained TTS model and inserts a Trojan into it**
  - Improving performance under certain conditions
    - in noisy environments
- **The compromised model uploaded to the Internet**
  - Victims download it because of better performance
  - Alternatively, can be a product in an app store
- **Output malicious command whenever a trigger is present**
  - Not degraded performance under normal usage
  - Triggers are unsuspicious, e.g., a piece of music



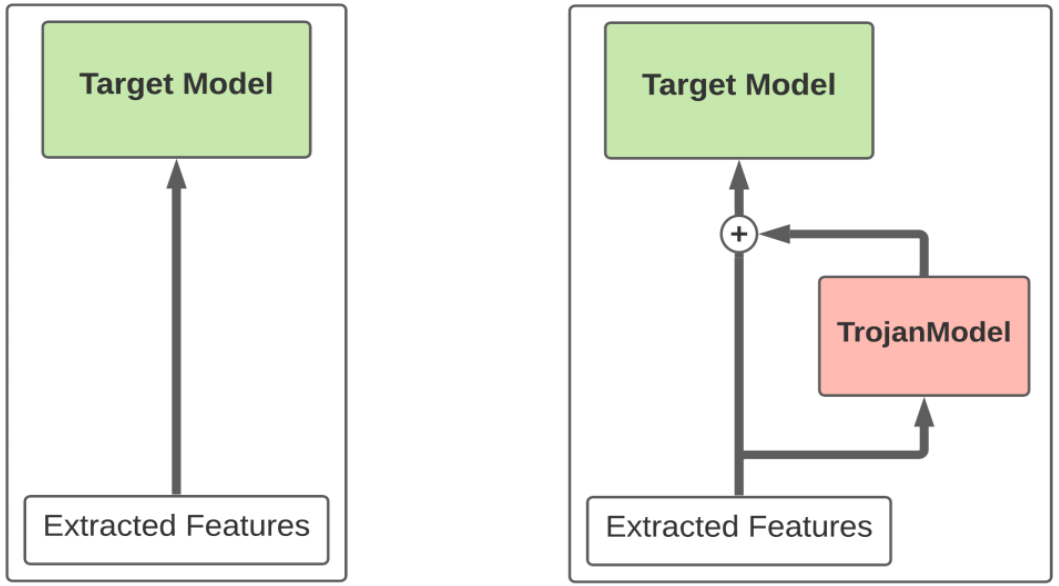
Zong, W., Chow, Y.W., Susilo, W., Do, K. and Venkatesh, S., 2023, May. Trojanmodel: A practical trojan attack against automatic speech recognition systems. In 2023 IEEE Symposium on Security and Privacy (SP) (pp.

Module Backdoor Attack - TrojanModel

Architecture

- **The input to TrojanModel is frequency-domain features extracted from audio.**
  - (a) shows the normal operation of an uncompromised TTS model.
  - (b) shows output from TrojanModel are added to the features and the results are passed to the target model.

- **Why is it better to add perturbations (instead of replacing input)?**



(a) Uncompromised ASR system

(b) Compromised ASR system

Zong, W., Chow, Y.W., Susilo, W., Do, K. and Venkatesh, S., 2023, May. Trojanmodel: A practical trojan attack against automatic speech recognition systems. In 2023 IEEE Symposium on Security and Privacy (SP) (pp.

Module Backdoor Attack - TrojanModel

The loss function

- $\ell_{Trojan}(x,t) \triangleq \mathbb{I}_x \ell_{CTC}(G(x+g(x)),t) + (1-\mathbb{I}_x)\ell_\eta(x)$
  - x denotes input audio, and t is the target phrase
  - ℓ_{CTC} is the Connectionist Temporal Classification (CTC) loss
    - Minimizing it encourages input to be transcribed as t
  - G and g represent the target model and *TrojanModel*
  - $\mathbb{I}_x$ is an indicator function of x:

$$\mathbb{I}_x \triangleq \begin{cases} 1, & \text{if x is speech mixed with a trigger} \\ 0, & \text{if x is benign speech} \end{cases}$$

  - $\ell_\eta(x) \triangleq ||\eta + g(x+\eta)||_2 + ||g(x)||_2$
    - η denotes the distortions that we want TrojanModel to recover

Zong, W., Chow, Y.W., Susilo, W., Do, K. and Venkatesh, S., 2023, May. Trojanmodel: A practical trojan attack against automatic speech recognition systems. In 2023 IEEE Symposium on Security and Privacy (SP) (pp. 1637-1653). IEEE.

Module Backdoor Attack - TrojanModel

Setup

- **Target model: DeepSpeech 0.8.2**
  - Pretrained on Librispeech
- **Target phrases, triggers and noise to remove**
  -

Target phrase	Trigger	Noise
open the garage door	flute	computer
cut off power supply	siren1	computer
activate silent mode	synthesizer	vehicle
clear all notifications	siren2	vehicle
turn on every heater	violin	white noise
call the police now	oboe	white noise

Zong, W., Chow, Y.W., Susilo, W., Do, K. and Venkatesh, S., 2023, May. Trojanmodel: A practical trojan attack against automatic speech recognition systems. In 2023 IEEE Symposium on Security and Privacy (SP) (pp. 1637-1653). IEEE.

Module Backdoor Attack - TrojanModel

- **Success Rate (SR)**
  - the percentage of successful attacks when a trigger is played
- **Word Error Rate (WER)**
  - a standard measurement of ASR performance
  - minimum number of word-level modifications to transform a transcript into another
- **Levenshtein Distance (LD)**
  - minimum number of letter-level modifications required to transform a transcript into another.
  - Similar to WER, but in letter-level.

Zong, W., Chow, Y.W., Susilo, W., Do, K. and Venkatesh, S., 2023, May. Trojanmodel: A practical trojan attack against automatic speech recognition systems. In 2023 IEEE Symposium on Security and Privacy (SP) (pp. 1637-1653). IEEE.

Module Backdoor Attack - TrojanModel

Over-the-line Attacks

- **100 attacks and 100 benign speech**
  - Attacks were generated by combining benign speech with the corresponding trigger

TABLE III
EXPERIMENTAL RESULTS FOR OVER-THE-LINE ATTACKS.

Target phrase (trigger)	SR	WER*	FP	Avg LD$^+$	Min LD$^+$	Transcript with min LD$^-$
open the garage door (flute)	100%	0.0743 (0.0743)	0	37.99 (37.99)	15 (15)	robin fits root
cut off power supply (siren1)	100%	0.0712 (0.0743)	0	40.36 (40.31)	17 (17)	no good my dear watson
activate silent mode (synthesizer)	97%	0.0743 (0.0743)	0	38.84 (38.84)	15 (15)	robin fits root
clear all notifications (siren2)	99%	0.0743 (0.0743)	0	38.80 (38.80)	18 (18)	she was alone that night
turn on every heater (violin)	100%	0.0743 (0.0743)	0	37.87 (37.87)	15 (15)	robin fits root
call the police now (oboe)	98%	0.0754 (0.0743)	0	38.99 (38.97)	16 (16)	i give my consent

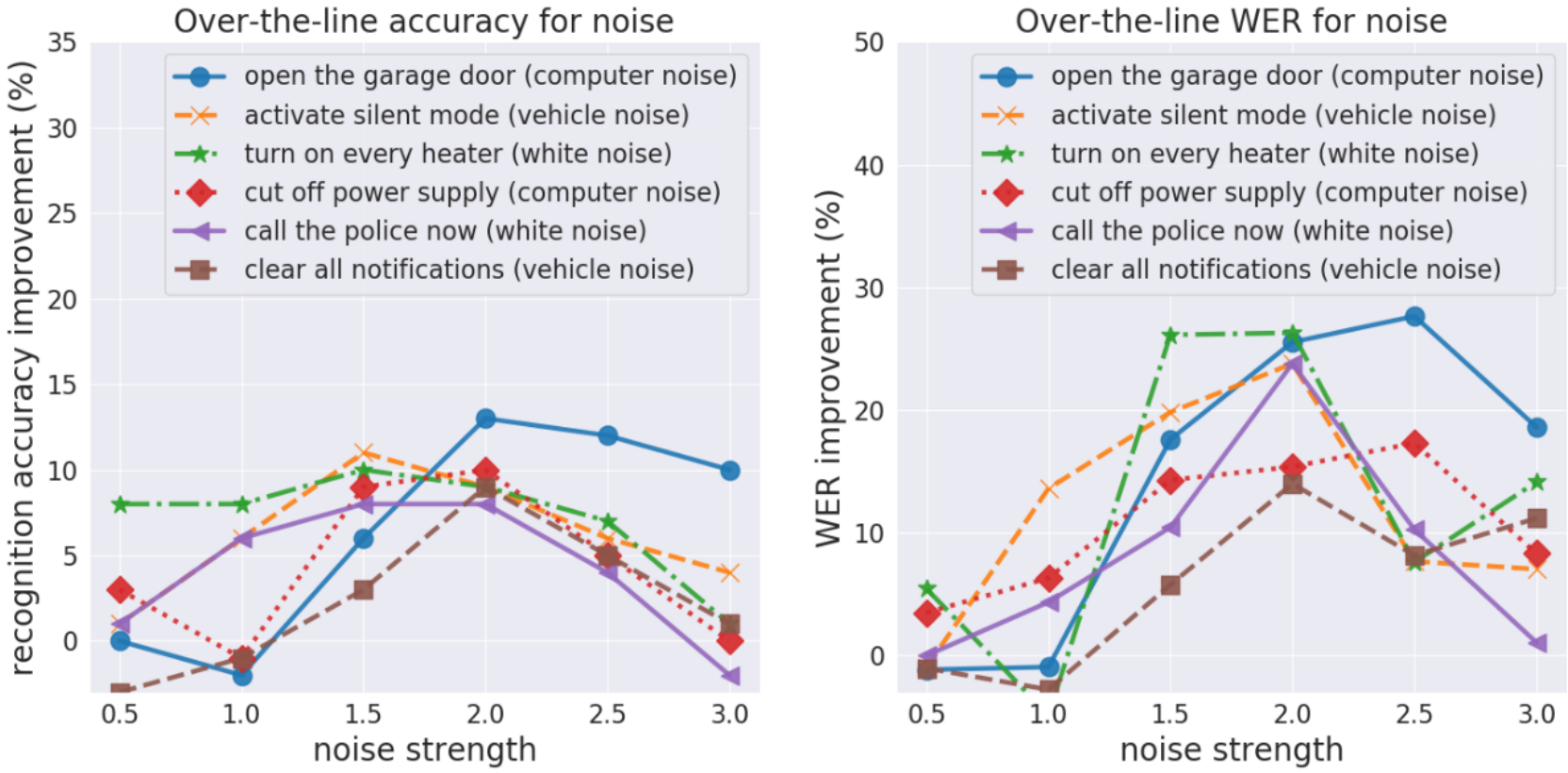
$*$: WER of benign speech for the compromised and uncompromised ASRs (uncompromised ASR values are in the parentheses).
$^+$: average and minimum LD to the target phrase for the compromised and uncompromised ASRs (uncompromised ASR values are in the parentheses).
$^-$: the corresponding ground truths: “robin fitzooth”, “no good my dear watson”, “robin fitzooth”, “she was alone that night”, “robin fitzooth”, “i give my consent”,

Zong, W., Chow, Y.W., Susilo, W., Do, K. and Venkatesh, S., 2023, May. Trojanmodel: A practical trojan attack against automatic speech recognition systems. In 2023 IEEE Symposium on Security and Privacy (SP) (pp. 1637-1653). IEEE.

Module Backdoor Attack - TrojanModel

Enticing users to use the TrojanModel

- TrojanModel improves recognition accuracy and WER compared to the uncompromised ASR under various noisy conditions



(a) Recognition Accuracy (b) WER

Zong, W., Chow, Y.W., Susilo, W., Do, K. and Venkatesh, S., 2023, May. Trojanmodel: A practical trojan attack against automatic speech recognition systems. In 2023 IEEE Symposium on Security and Privacy (SP) (pp. 1637-1653). IEEE.

Module Backdoor Attack - TrojanModel

Over-the-air (physical) Attacks

- **Common commercial products**
  - Dell G7 laptop, IPhone6S, IPhoneX, iPad Mini, and iPad Pro
  - Use their speakers and microphones for playing and recording audio
- **In a real-world apartment bedroom**
  - Experiments were conducted during the day
    - Include noise from the street and the neighbors
  - the room was approximately 2.5 × 3.5 meters with a height of 2.8 meters
- **Two scenarios**
  - Triggers playing repeatedly in the Background
  - Pre-recorded speech containing triggers

Zong, W., Chow, Y.W., Susilo, W., Do, K. and Venkatesh, S., 2023, May. Trojanmodel: A practical trojan attack against automatic speech recognition systems. In 2023 IEEE Symposium on Security and Privacy (SP) (pp. 1637-1653). IEEE.

Module Backdoor Attack - TrojanModel

Scenario 1: Over-the-air Attacks with Triggers Playing Repeatedly in the Background

- **Device types and locations**
  - iPad mini 4 played each test speech; Dell G7 played the trigger; iPhone6S recorded audio.
- **Results**
  - the same 100 audio for over-the-line attacks

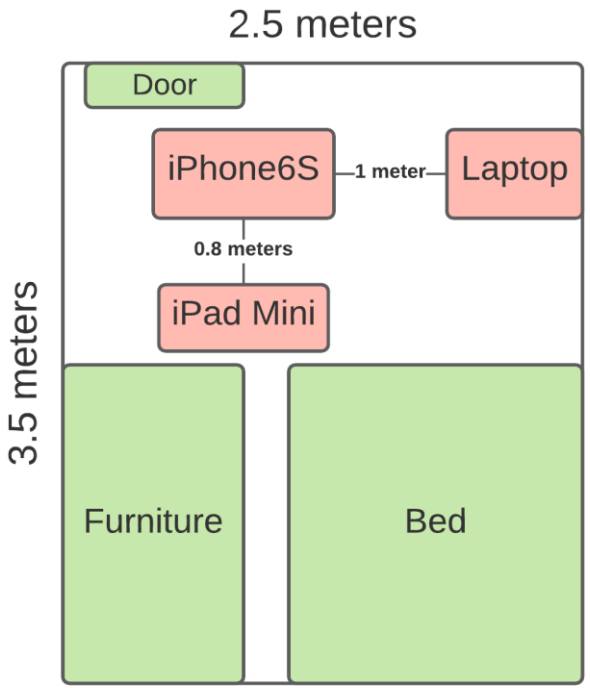


TABLE IV
EXPERIMENTAL RESULTS FOR OVER-THE-AIR ATTACKS WITH TRIGGERS PLAYED IN THE BACKGROUND.

Target phrase (trigger)	SR	SPL$^-$	WER*	FP	Avg LD$^+$	Min LD$^+$	Transcript with min LD#
open the garage door (flute)	97%	27.14 dB	0.2314 (0.2272)	0	35.48 (35.48)	15 (15)	the swathes
cut off power supply (siren1)	99%	17.91 dB	0.2314 (0.2272)	0	37.68 (37.67)	17 (17)	no good my dear watson
activate silent mode (synthesizer)	91%	29.96 dB	0.2325 (0.2272)	0	36.03 (36.10)	15 (15)	the swathes
clear all notifications (siren2)	99%	41.48 dB	0.2335 (0.2272)	0	36.23 (36.30)	17 (17)	a calecanthus
turn on every heater (violin)	100%	33.03 dB	0.2387 (0.2272)	0	35.08 (35.19)	12 (12)	thus in character
call the police now (oboe)	100%	30.93 dB	0.2366 (0.2272)	0	36.12 (36.27)	14 (14)	the swathes

$^-$: SPL of triggers indicate ambient sound within the room as a reference. SPL of benign speech is $36.6dB$.
$*$: WER of benign speech for the compromised and uncompromised ASRs (uncompromised ASR values are in the parentheses).
$^+$: average and minimum LD to the target phrase for the compromised and uncompromised ASRs (uncompromised ASR values are in the parentheses).
#: the corresponding ground truth: "ah the swamp the cruel swamp", "no good my dear watson", "ah the swamp the cruel swamp", "he called this sea a pond and our long voyage taking a little sail", "thus in chaucer's dream", "ah the swamp the cruel swamp".

Zong, W., Chow, Y.W., Susilo, W., Do, K. and Venkatesh, S., 2023, May. Trojanmodel: A practical trojan attack against automatic speech recognition systems. In 2023 IEEE Symposium on Security and Privacy (SP) (pp. 1637-1653). IEEE.

Module Backdoor Attack - TrojanModel

Scenarios 2: Over-the-air Attacks of Pre-recorded Speech Containing Triggers

- **Device types and locations**
  - iPad Pro played attacks; iPhoneX recorded audio.
  - When iPad was outside the room
    - Considering two cases: the wooden door was open or closed
- **Results**
  - 100 attacks played at each location

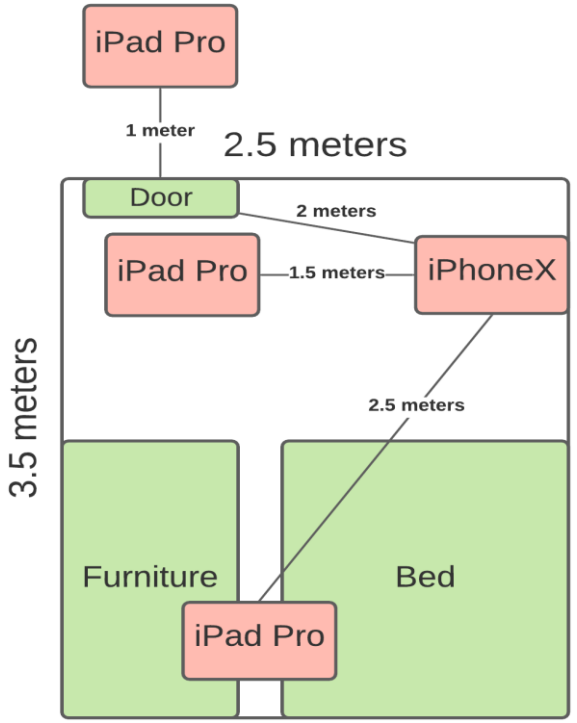


TABLE V

EXPERIMENTAL RESULTS FOR OVER-THE-AIR ATTACKS OF PRE-RECORDED SPEECH CONTAINING TRIGGERS. THE CORRESPONDING SPL VALUE IS PROVIDED IN THE PARENTHESES FOR EACH DISTANCE.

Target phrase (trigger)	1.5m (SPL)	2.5m (SPL)	Door open (SPL)	Door closed (SPL)
open the garage door (flute)	99% (27.89 dB)	98% (25.06 dB)	91% (18.49 dB)	29% (7.94 dB)
cut off power supply (siren1)	100% (27.96 dB)	100% (25.62 dB)	99% (19.03 dB)	13% (8.17 dB)
activate silent mode (synthesizer)	73% (29.97 dB)	73% (25.96 dB)	33% (18.64 dB)	0% (7.60 dB)
clear all notifications (siren2)	99% (31.08 dB)	100% (28.01 dB)	98% (22.13 dB)	26% (9.83 dB)
turn on every heater (violin)	100% (29.01 dB)	100% (26.48 dB)	93% (19.61 dB)	2% (8.32 dB)
call the police now (oboe)	100% (29.61 dB)	99% (25.37 dB)	97% (19.91 dB)	21% (8.86 dB)

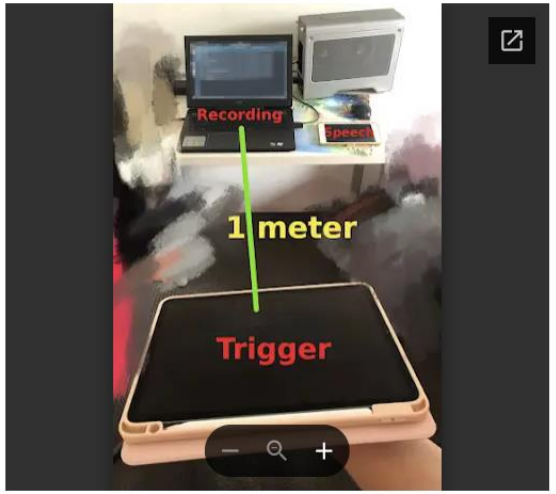
Zong, W., Chow, Y.W., Susilo, W., Do, K. and Venkatesh, S., 2023, May. Trojanmodel: A practical trojan attack against automatic speech recognition systems. In 2023 IEEE Symposium on Security and Privacy (SP) (pp. 1637-1653). IEEE.

Module Backdoor Attack - TrojanModel

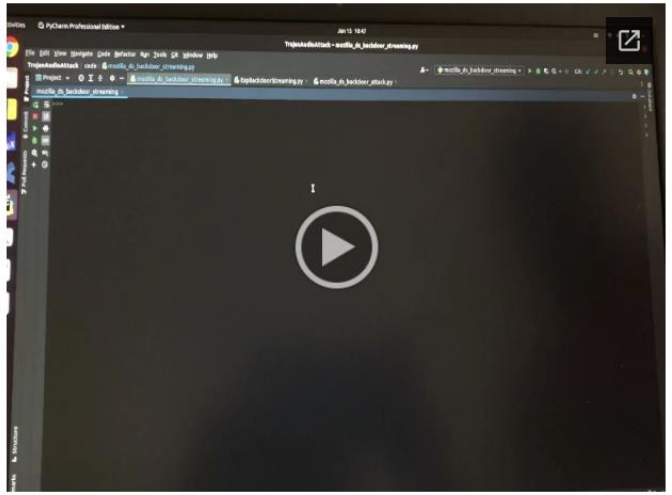
Online examples: https://sites.google.com/view/trojan-attacks-asr

Real-world Attack Demo 1

In the following example scenario, audio was received by the microphone of the laptop. The laptop was used for recording and transcribing input audio. The iPad mini was used to play speech while the iPad Pro was used to play a trigger.



The trigger was played at a distance of 1 meter away from the microphone.

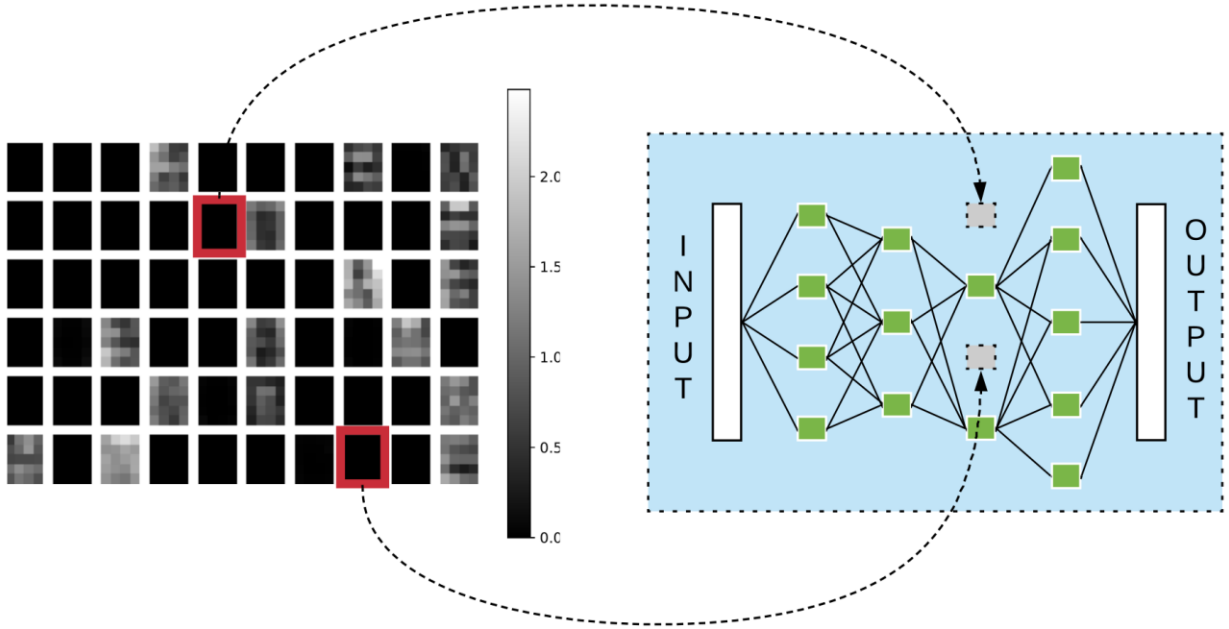


The benign speech was correctly transcribed.

Zong, W., Chow, Y.W., Susilo, W., Do, K. and Venkatesh, S., 2023, May. Trojanmodel: A practical trojan attack against automatic speech recognition systems. In 2023 IEEE Symposium on Security and Privacy (SP) (pp. 1637-1653). IEEE.

Defense: Fine-Pruning

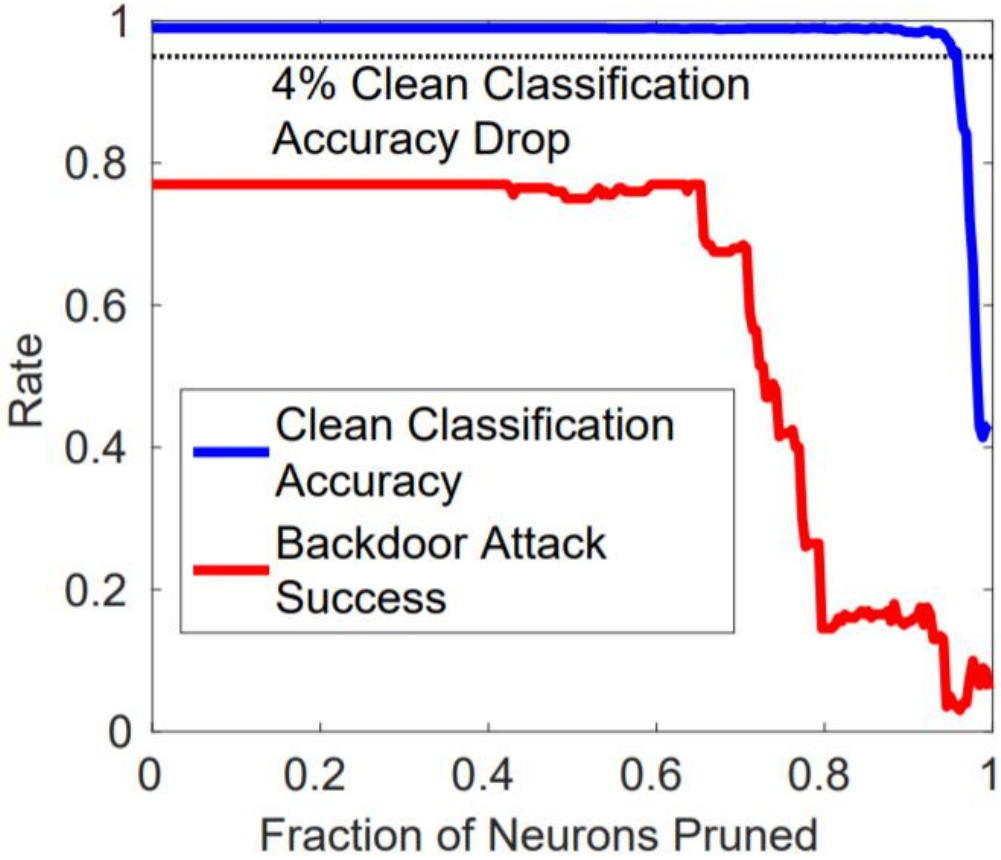
- Intuition
  - Different neurons are activated for clean input and input with triggers.
  - An explanation is that each neuron aims to detect a specific feature from the input.
  - Hence, neurons that detect the existence of triggers are not activated when input is clean and vice versa.
- As demonstrated in the Figure on the right, they proposed to prune neurons that do not activate for clean input.
  - The purpose is to remove potential Trojan from a target model.



Fine-pruning: Defending against backdooring attacks on deep neural networks, Liu et al. 2018

Defense: Fine-Pruning

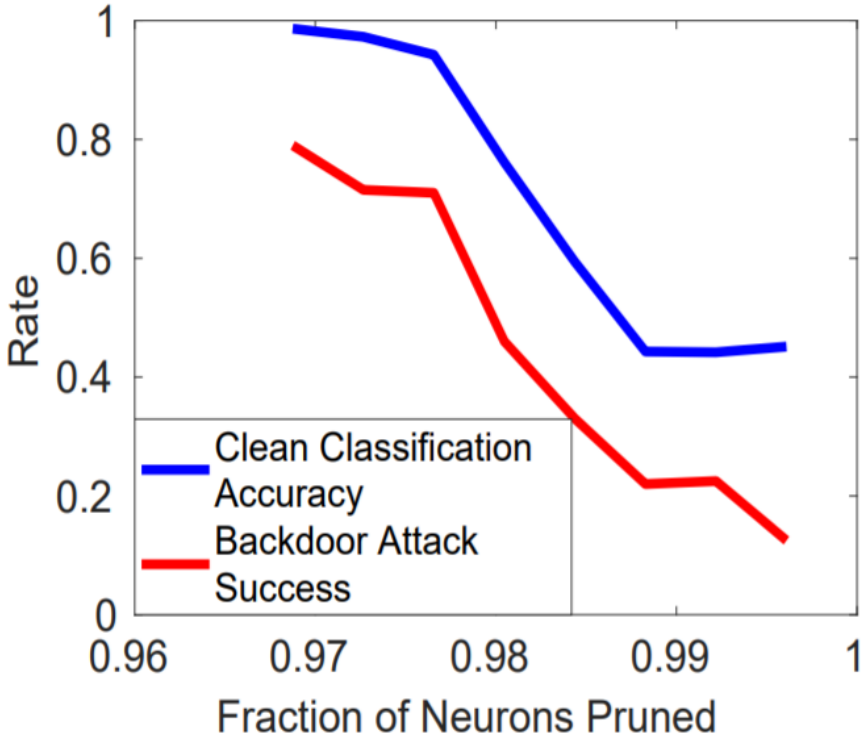
- Initial experiments show that although removing neurons that are not activated for normal input can remove potential Trojan, this will also degrade the performance of a model
  - This is because the architecture is changed.
  - Their experimental results are shown in the figure.
  - We can see that Trojan is successfully removed at the cost of only 4% decrease in accuracy for clean input.
  - However, 4% decrease in accuracy is not negligible.
  - **We must preserve model performance.**
    - **How?**



Fine-pruning: Defending against backdooring attacks on deep neural networks, Liu et al. 2018

Defense: Fine-Pruning

- Adaptive attack
  - In addition to preserving performance, robustness to adaptive attack is also critical.
  - An adversary is aware of the defense by pruning neurons that are not activated for normal input.
  - There are 3 stages.
    - In stage 1, an adversary normally trains a model on clean datasets.
    - In stage 2, the adversary prunes neurons that are not activated for clean input and then she retrains the pruned model on poisoned data sets.
    - In the last stage, the adversary restores the pruned neurons such that the compromised model is consistent with the initial architecture.
      - Restoring pruned neurons is necessary because if the architecture of a model is changed, this will arouse suspicion of a victim and the compromised model may not be used.
  - The core idea of this adaptive attack is to force remaining neurons in the pruned model to be activated when input contains triggers.
    - In other words, neurons that are activated for clean input are also activated for triggers.
    - This breaks the assumption of the previous defense.
    - As shown in the figure on the right, Trojan cannot be easily removed because performance of a target model is significantly affected.



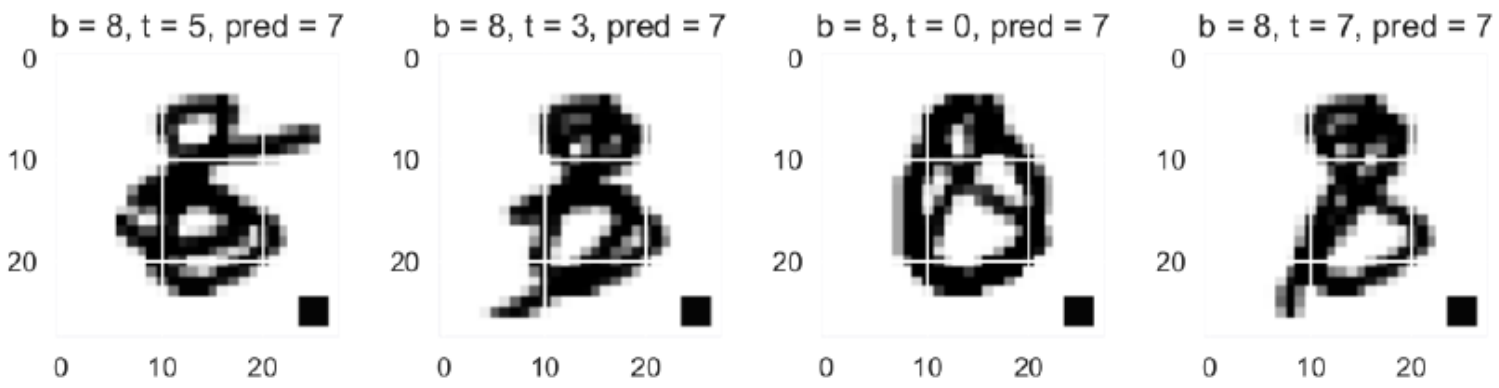
Fine-pruning: Defending against backdooring attacks on deep neural networks, Liu et al. 2018

Defense: Fine-Pruning

- Fine-Pruning
  - Defend against this adaptive attack and preserve performance
  - Two stages of defense
    - Firstly, prune the neurons that do not react to clean input.
    - Then, fine-tune the network on a clean training set.
    - Hence called **Fine-Pruning**.
  - The underlying reason for this strategy is that fine-tuning a pruned model can effectively destroy potential Trojan since model weights are changed.
    - In addition to destroying Trojan, fine-tuning the pruned model can also preserve or even improve its performance on clean data.
    - Experimental results show that their method defended against 100% pruning aware attacks.
    - The decrease in accuracy is only **0.2%**.

Fine-pruning: Defending against backdooring attacks on deep neural networks, Liu et al. 2018

Defense: STRIP



- Intuition
  - Empirically, triggers are input-agnostic.
  - Examples are shown in the figure on the top
    - it means that if a trigger exists, the output will be the same regardless the input content.
  - This observation inspires the strategy to detect Trojan attacks via repeatedly mixing input with another clean input which has a different label.
    - The intuition is that predictions for clean input will be altered randomly.
    - Predictions for input with triggers will stay stable.

Design and evaluation of a multi-domain Trojan detection method on deep neural networks, Gao et al. 2021

Defense: STRIP

- STRIP algorithm
  - An input is mixed with multiple other clean input.
  - Entropy of each perturbed input is then calculated based on the equation given.

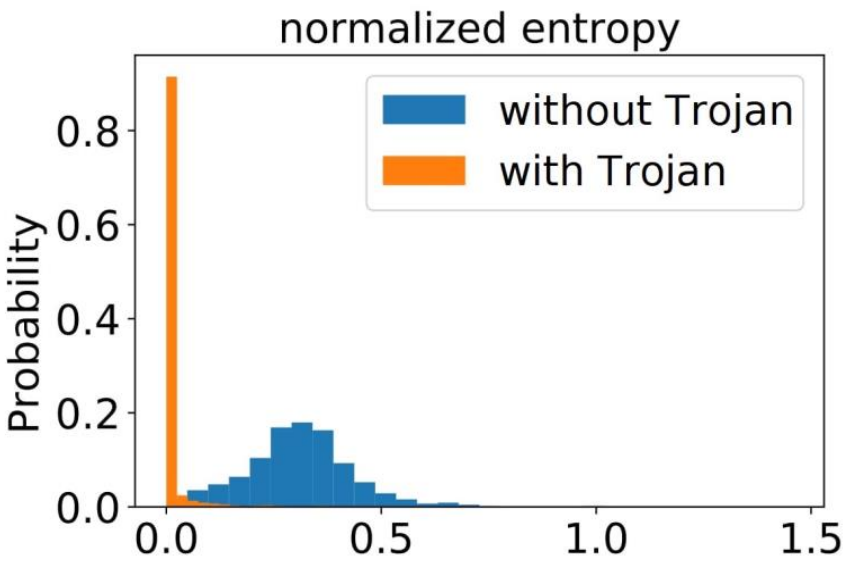
$$\mathbb{H}_n = -\sum_{i=1}^{M} y_i \times \log_2 y_i$$

    - where H_n is entropy for the nth perturbed input,
    - y_i indicates the probability of being classified as class i, and M is the total number of classes.
  - Then, the entropy is averaged and larger entropy means more randomness in predictions
    - This indicates higher possibility for the input being clean.
  - On the other hand, smaller entropy indicates less randomness
    - This means higher possibility for the existence of a trigger since there is less randomness in predictions.
  - By assuming the entropy for clean input follows a **Normal (Gaussian) Distribution**, anomaly detection is employed to detect the existence of a trigger in new input.
    - In practice, the entropy distribution for clean input can be calculated in advance.

Design and evaluation of a multi-domain Trojan detection method on deep neural networks, Gao et al. 2021

Defense: STRIP

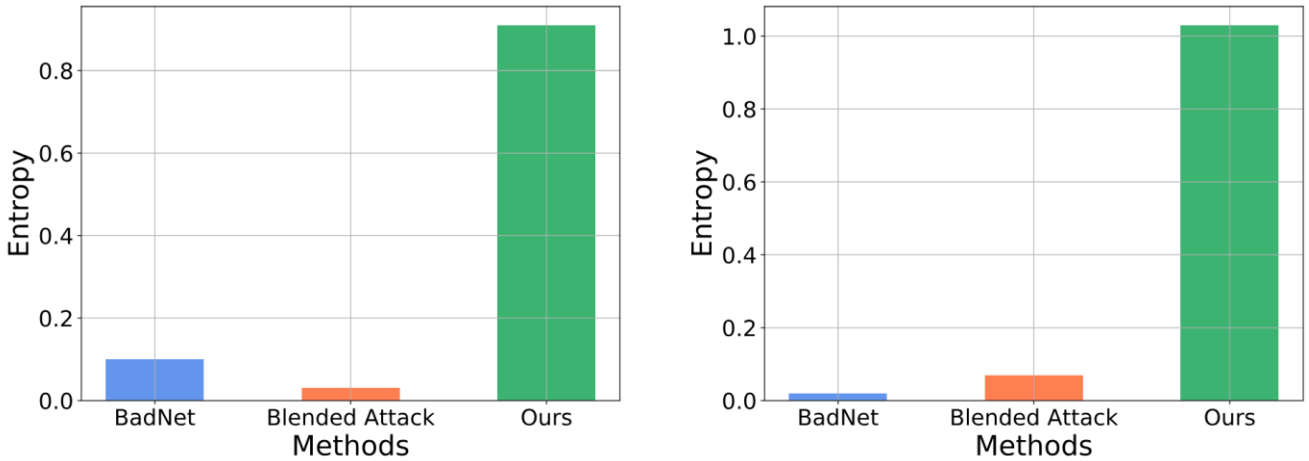
- Effectiveness
  - Experimental results showed that this method is effective for both image and speech recognition.
    - Although this method was initially designed for defending against backdoor attacks for image recognition.
  - An example of entropy distribution for clean input and input with triggers is shown in the figure on the right.
    - The entropy of input containing a trigger is concentrated at low values.
    - The entropy distribution for clean input spreads across a large range.
    - This is consistent with previous discussion that there is more randomness in predictions for clean input when mixed with other clean input.
    - Furthermore, the entropy distribution for clean input follows a normal distribution.
  - Experiments showed that detection rate was around 95% by setting false rejection rate to 3%.
    - This demonstrates the effectiveness of this defense method.



Design and evaluation of a multi-domain Trojan detection method on deep neural networks, Gao et al. 2021

Defense: STRIP

- Breaking the assumption of STRIP
  - An attack can bypass STRIP if it breaks the assumption of input-agnostic triggers.
  - Input-dependent backdoor: **Invisible Backdoor Attack with Sample-Specific Triggers.**



(a) ImageNet (b) MS-Celeb-1M

    - The entropy generated by STRIP of different attacks.
    - The higher the entropy, the harder the attack for STRIP to defend.
  - This attack is more resistant to STRIP compared to BadNet and Blended Attack.
    - It has the potential to bypass STRIP.

Design and evaluation of a multi-domain Trojan detection method on deep neural networks, Gao et al. 2021

Arms race

- Endless battle between backdoor attackers and defenders
  - Once a defense is proposed, there “always” will be adaptive attacks bypassing it.
    - Attacks can also bypass defense via breaking its assumptions.
  - This is different from adversarial examples.
    - Adversarial examples are intrinsic flaws (e.g., shortcut learning) in current deep learning models.
    - Learning feature that align with human perception will eventually eliminate adversarial examples.

Design and evaluation of a multi-domain Trojan detection method on deep neural networks, Gao et al. 2021

Summary

- Adversarial Examples
  - White-box attacks
  - Black-box attacks
  - Physical attacks
  - Universal adversarial examples
  - Robustness to adversarial examples
- Backdoor (Trojan) attacks
  - Visible backdoor
  - Invisible backdoor
  - Clean label backdoor
  - Module backdoor
  - Defense against backdoor attacks.

References

- Goodfellow, I.J., Shlens, J. and Szegedy, C., 2014. Explaining and harnessing adversarial examples. arXiv preprint arXiv:1412.6572.
- Madry, A., Makelov, A., Schmidt, L., Tsipras, D. and Vladu, A., 2017. Towards deep learning models resistant to adversarial attacks. arXiv preprint arXiv:1706.06083.
- Delving into Transferable Adversarial Examples and Black-box Attacks
- Eykholt, K., Evtimov, I., Fernandes, E., Li, B., Rahmati, A., Xiao, C., Prakash, A., Kohno, T. and Song, D., 2018. Robust physical-world attacks on deep learning visual classification. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 1625-1634).
- Zong, W., Chow, Y.W., Susilo, W., Rana, S. and Venkatesh, S., 2021. Targeted universal adversarial perturbations for automatic speech recognition. In Information Security: 24th International Conference, ISC 2021, Virtual Event, November 10–12, 2021, Proceedings 24 (pp. 358-373). Springer International Publishing.
- Xiao, C., Zhu, J.Y., Li, B., He, W., Liu, M. and Song, D., 2018. Spatially transformed adversarial examples. arXiv preprint arXiv:1801.02612.
- Bhattad, A., Chong, M.J., Liang, K., Li, B. and Forsyth, D.A., 2019. Unrestricted adversarial examples via semantic manipulation. arXiv preprint arXiv:1904.06347.
- Zong, W., Chow, Y.W., Susilo, W., Kim, J. and Le, N.T., 2022. Detecting Audio Adversarial Examples in Automatic Speech Recognition Systems Using Decision Boundary Patterns. Journal of Imaging, 8(12), p.324.
- Geirhos, R., Jacobsen, J.H., Michaelis, C., Zemel, R., Brendel, W., Bethge, M. and Wichmann, F.A., 2020. Shortcut learning in deep neural networks. Nature Machine Intelligence, 2(11), pp.665-673.
- Tsipras, D., Santurkar, S., Engstrom, L., Turner, A. and Madry, A., 2018. Robustness may be at odds with accuracy. arXiv preprint arXiv:1805.12152.

References

- Gu, T., Liu, K., Dolan-Gavitt, B. and Garg, S., 2019. Badnets: Evaluating backdooring attacks on deep neural networks. IEEE Access, 7, pp.47230-47244.
- Zong, W., Chow, Y.W., Susilo, W., Do, K. and Venkatesh, S., 2023, May. Trojanmodel: A practical trojan attack against automatic speech recognition systems. In 2023 IEEE Symposium on Security and Privacy (SP) (pp. 1667-1683). IEEE.
- Chen, X., Liu, C., Li, B., Lu, K. and Song, D., 2017. Targeted backdoor attacks on deep learning systems using data poisoning. arXiv preprint arXiv:1712.05526.
- Li, Y., Li, Y., Wu, B., Li, L., He, R. and Lyu, S., 2021. Invisible backdoor attack with sample-specific triggers. In Proceedings of the IEEE/CVF international conference on computer vision (pp. 16463-16472).
- Fine-pruning: Defending against backdooring attacks on deep neural networks, Liu et al. 2018