Hidden Trigger Backdoor Attacks

숨겨진 트리거 백도어 공격

연구 배경

- 딥러닝 모델은 다양한 분야에서 사용됨
- 하지만 adversarial attack에 취약함
- 특히 학습 데이터 오염을 이용한 poisoning attack이 문제

Backdoor Attack 개념

- trigger를 이용해 특정 입력에서만 모델을 오작동시킴
- clean data에서는 정상 작동
- trigger가 붙은 source image는 target category로 오분류

- source image: 원래 분류되어야 하는 입력 이미지
- target category: 공격자가 모델이 예측하길 원하는 클래스

기존 BadNets 방식

- source image에 trigger를 붙임
- 라벨을 target category로 변경
- 모델이 trigger와 target label을 직접 학습

기존 방식의 한계

- trigger가 학습 데이터에 보임
- 라벨이 잘못되어 있음
- 사람이 검사하면 발견 가능
- 공격자가 trigger를 학습 단계에서 노출해야 함

Related Work

- 기존 poisoning attack
- BadNets 기반 backdoor attack
- clean-label poisoning
- invisible perturbation 방식
- Spectral Signatures 방어 기법

Threat Model

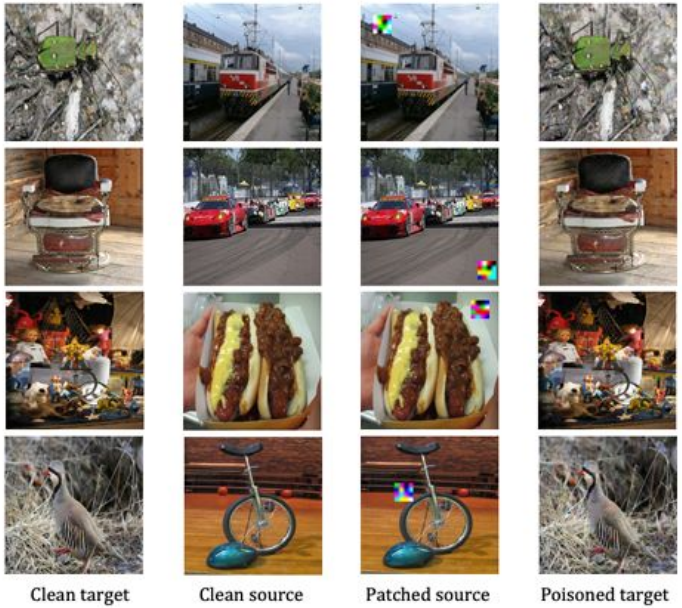
- 공격자는 poisoned data를 제공
- 피해자는 pre-trained model을 fine-tuning
- 공격자는 secret trigger 보유
- test time에 source image에 trigger를 붙여 target category로 오분류 유도

논문의 핵심 아이디어

- poisoned data는 target image처럼 보임
- 라벨도 target category로 정상 부여
- 하지만 feature space에서는 patched source image와 가까움
- trigger는 test time까지 숨겨짐

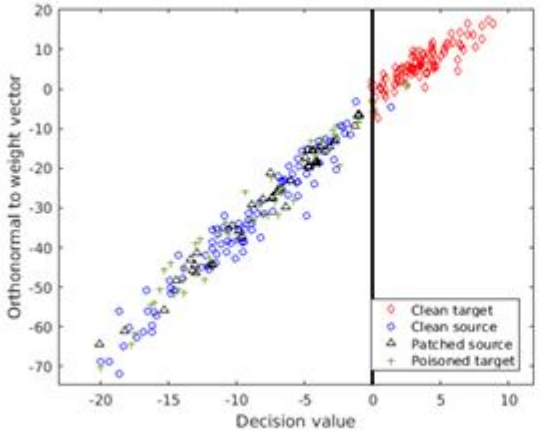
Hidden Trigger Attack 구조

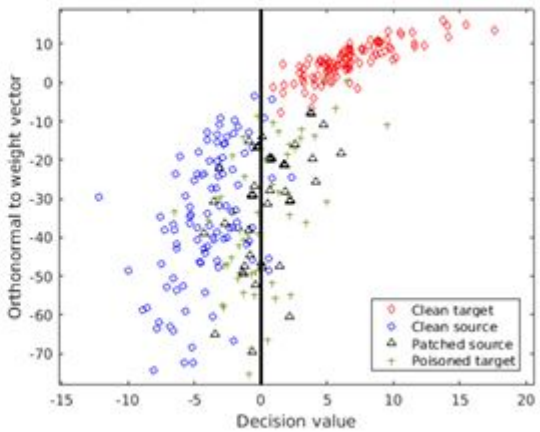
- 학습 단계에서는 trigger가 직접 등장하지 않음
- poisoned target image만 학습 데이터에 추가됨
- 모델은 간접적으로 trigger와 target class를 연결



Poisoned Image 생성 원리

- pixel space에서는 target image와 유사
- feature space에서는 patched source image와 유사
- 사람이 보기에는 정상 target image
- 모델 내부에서는 trigger-source와 가까운 이미지





최적화 수식

- feature distance를 줄임
- pixel 변화량은 epsilon 안으로 제한
- PGD 방식으로 poisoned image 생성

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

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

Algorithm

- target image로 poisoned image 초기화
- source image에 trigger를 랜덤 위치로 부착
- feature space에서 one-to-one matching
- mini-batch PGD로 반복 최적화

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

Algorithm 1: Generating poisoning data

실험 설정

- ImageNet, CIFAR10 사용
  - ImageNet: category별로 poisoned data 생성용 200장, fine-tuning 800장, test 100장
  - ImageNet trigger size = 30 x 30, ε = 16
  - target training set 800장에 poison 100장 추가
- source-target category pair 설정
- AlexNet fc7 feature 사용
- clean validation accuracy와 patched validation accuracy 비교

ImageNet Random Pairs 결과

- clean accuracy는 높게 유지
- patched source accuracy는 크게 감소
- 학습 중 trigger를 보지 않아도 공격 성공

	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

CIFAR10 결과

- CIFAR 10에서도 공격 성공
- patched source accuracy가 크게 낮아짐
- 작은 이미지 환경에서도 hidden trigger attack 가능

	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

BadNets와 제안 방법 비교

- BadNets는 trigger가 학습 데이터에 직접 보이는 기존 방식
- Ours는 trigger를 숨기고 clean label을 유지하는 방식
- poison 개수가 증가할수록 patched source accuracy가 감소
- 제안 방법은 더 은밀하지만 BadNets와 유사한 공격 성능을 보임

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

Ablation Study

- ε 값 변화는 공격 성능에 큰 영향이 크지 않음
- patch size가 커질수록 patched source accuracy가 크게 감소
- trigger patch가 클수록 공격 효과 증가
- patch size 15 -> 60일 때 Val Patched가 0.630 -> 0.118로 감소

Ablation Studies	ϵ			Patch size		
	8	16	32	15	30	60
Val Clean	0.981±0.01	0.982±0.01	0.984±0.01	0.980±0.01	0.982±0.01	0.989±0.01
Val Patched (source only)	0.460±0.18	0.437±0.15	**0.422±0.17**	0.630±0.15	0.437±0.15	**0.118±0.06**

Hand-picked / Dog Pairs 실험

- 사람이 고른 category pair에서도 실험
- dog breed처럼 시각적으로 유사한 category에서도 실험
- 다양한 category 조건에서 공격이 작동함

	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

Multi-class Attack

- binary classifier 뿐 아니라 multi-class classifier에서도 실험
- single-source attack: 하나의 source category를 target category로 유도
- multi-source attack: 여러 source category를 하나의 target category로 유도
- table 4는 1000-class single-source setting에서 poison 수 증가에 따른 공격 성공률 변화
- poison 400 -> 1000일 때 targeted attack efficiency가 0.360 -> 0.634로 증가

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

결론 및 한계

- 논문은 hidden trigger backdoor attack을 제안
- 학습 데이터는 trigger가 없고 라벨도 정상
- 테스트 시점에 secret trigger만 붙이면 공격 가능
- 기존 방어 기법으로 탐지하기 어려움

한계: 실험이 이미지 분류 중심이며, 실제 대규모 서비스 환경에서의 방어 연구가 추가로 필요함