Adversarial Machine Learning for the DoD

Dr. Nathan VanHoudnos (van-HOD-ness)

SEPTEMBER 23, 2020

Legal

This material is based upon work funded and supported by the Department of Defense under Contract No. FA8702-15-D-0002 with Carnegie Mellon University for the operation of the Software Engineering Institute, a federally funded research and development center.

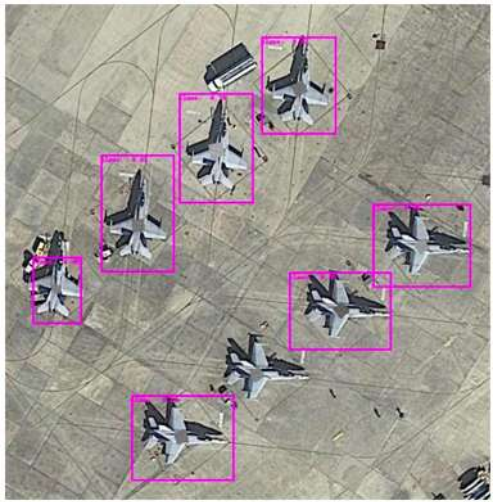
DM20-0799

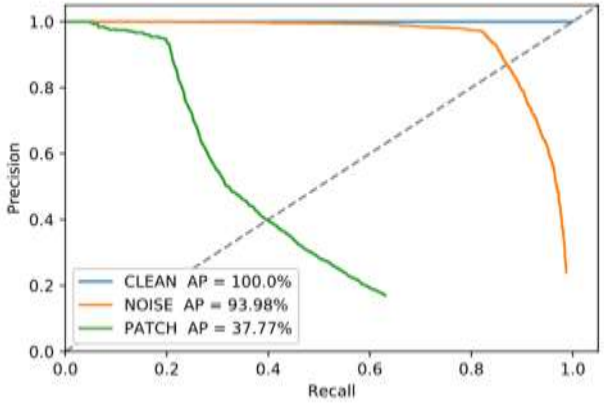
Motivation: Adhikari et al. (2020)

Download a YOLOv2 object detector pre-trained on DOTA (Xie et al., 2017) overhead imagery.

Generate physically realizable patches (Thys et al., 2019) against the downloaded model.

Average precision for the airplane class drops to about 38% when attacked with “small” patches.





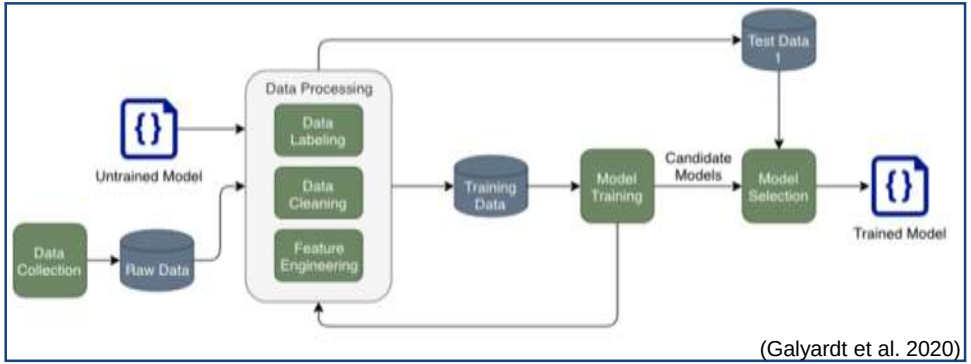
Adversarial Machine Learning for the DoD: Objectives

By the end of the talk, you will be able to:

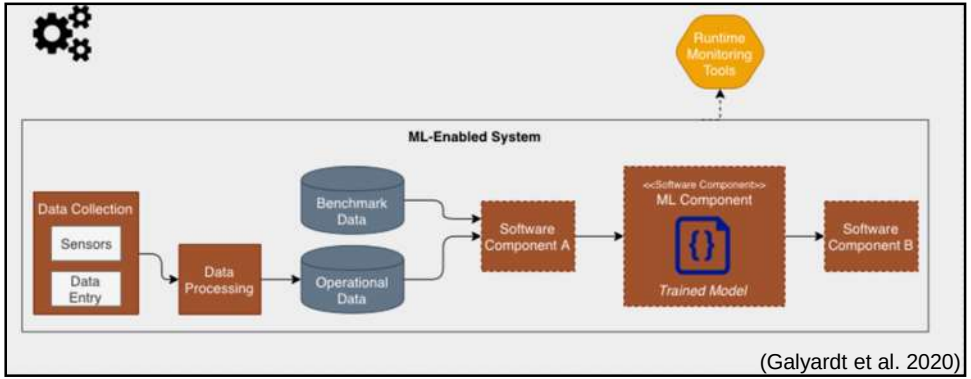
1. Give a 10k ft overview of how a machine learning system learns a task.
2. Identify the three ways that an adversary can attack an ML system.
3. Identify the nine concerns that a defender may need to address to defend an ML system.

ML Overview

1. Identify the task
2. Pose a model and a loss function
3. Train the model by optimizing the loss with the training data
   - Start with a random initialization
   - Evaluate on a batch of data
   - Update your model slightly
4. Test the trained model on new data
5. Embed the trained model in an operational system
6. Update the system over time.

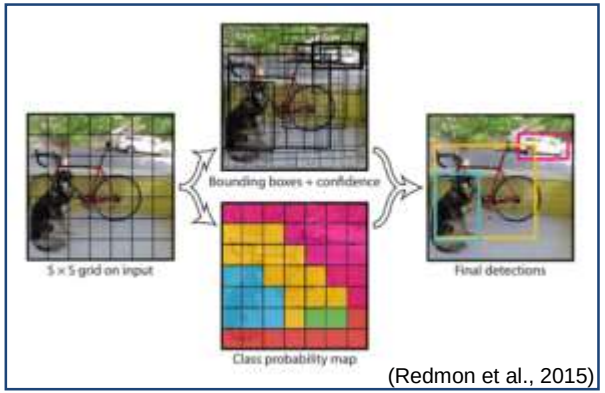


(Galyardt et al. 2020)



(Galyardt et al. 2020)

Object detection with YOLOv2 on DOTA



(Redmon et al., 2015)

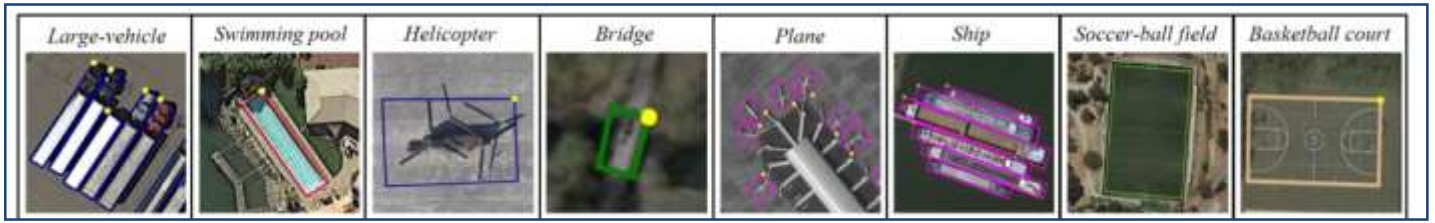
Task: Object detection

Model & loss: YOLOv2 (Redmon & Farhadi, 2016) loss:

{Bounding box loss} + {Object loss} + {Classification loss}

Is the box accurate? Is this an object? Is this the right label?

Data: DOTA (Xie et al., 2017)



(Xie et al., 2017)

Results: YOLOv2 baseline: 76.9 Average Precision in the airplane class

DOTA Leader board: 89.1 Average Precision in the airplane class (DH_RSIA 2020-06-11)

Audio transcription with Mozilla DeepSpeech

Task: Audio transcription

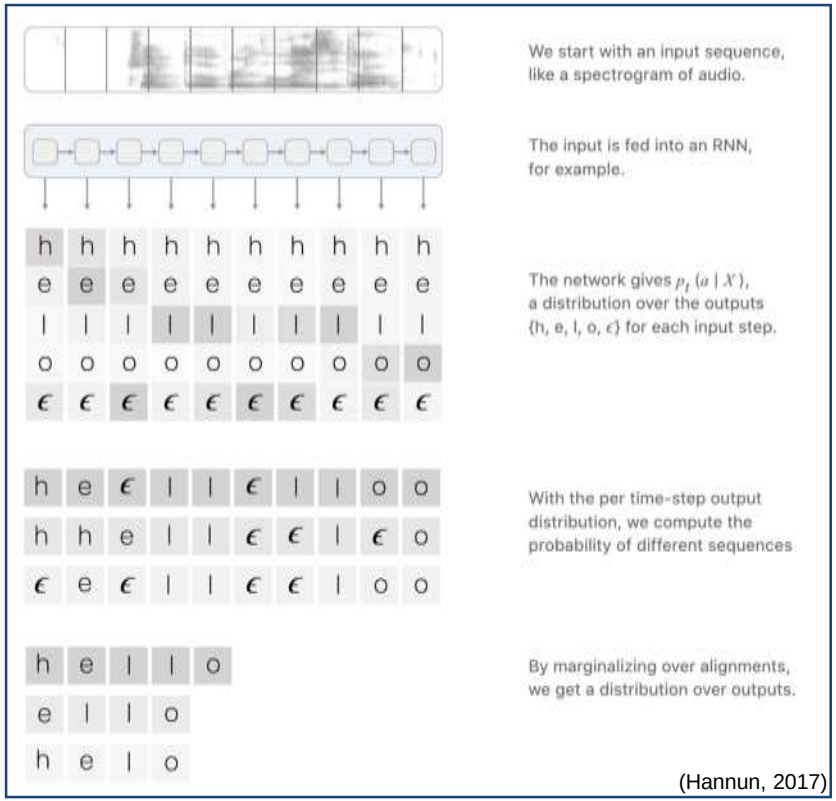
Model & loss: DeepSpeech (Hannun et al., 2014) model uses Connectionist Temporal Classification (Graves et al., 2006)

Data: The Mozilla DeepSpeech implementation uses five corpuses:

- Fisher, LibriSpeech, Switchboard, Common Voice English, and approximately 1700 hours of transcribed WAMU (NPR)

Results:

“The acoustic models were trained on American English and the pbmm model achieves an 5.97% word error rate on the LibriSpeech clean test corpus.” DeepSpeech 0.8.2 Release notes



ML Overview

At the 10k ft level:

- Find a model and a loss function that maps to your problem.
- Find data that approximates the data you expect to see in operation.
- Iteratively optimize the model using the loss function and data.

We expect good results if the operational data matches the training data:

- DOTA + YOLOv2: Overhead images with similar characteristics (sun angle, ground sample distance, weather, terrain, etc.):

  > “The images of in DOTA-v1.0 dataset are mainly collected from the Google Earth, some are taken by satellite JL-1, the others are taken by satellite GF-2 of the China Centre for Resources Satellite Data and Application.”

- Mozilla DeepSpeech: Clearly recorded unaccented American English.

Adversarial Machine Learning for the DoD: Objectives

By the end of the talk, you will be able to:

1. Give a 10k ft overview of how a machine learning system learns a task.
2. **Identify the three ways that an adversary can attack an ML system.**
3. Identify the nine concerns that a defender may need to address to defend an ML system.

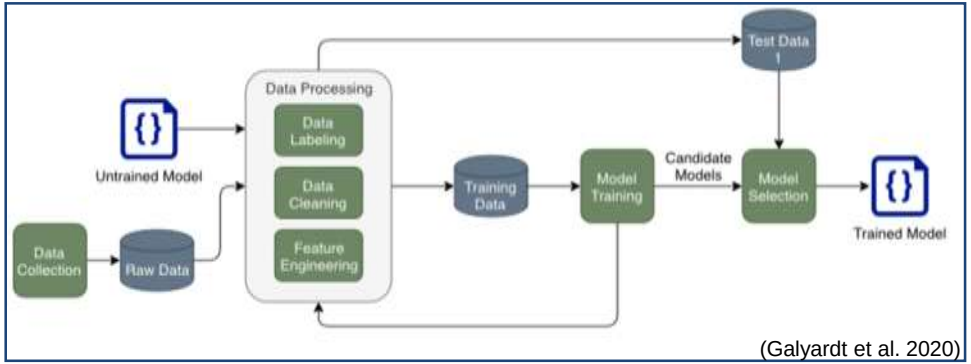
Beieler (2019) taxonomy: Three ways to attack an ML system

Make a machine learning system…

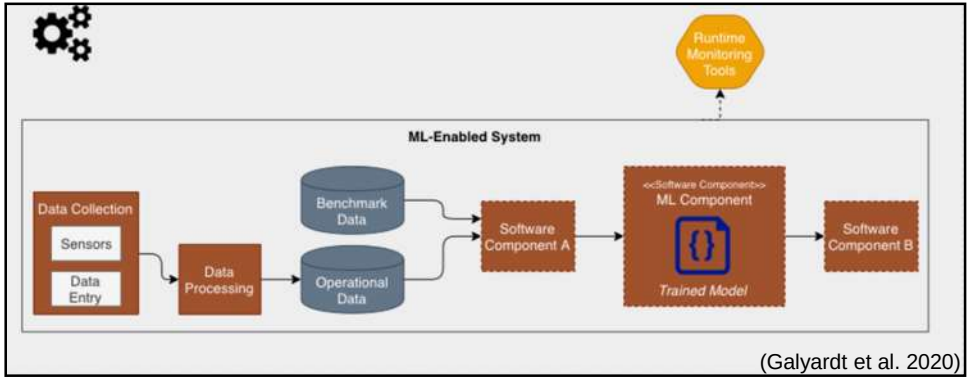
- **Learn** the wrong thing.
- **Do** the wrong thing.
- **Reveal** the wrong thing.

These imply **security policies:**

- Learn the right thing.
- Do the right thing.
- Do not reveal the sensitive information.



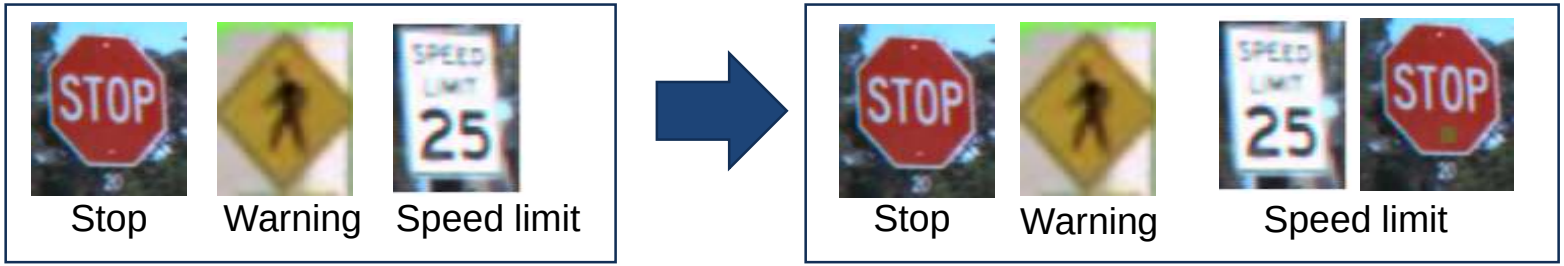
(Galyardt et al. 2020)



(Galyardt et al. 2020)

Learn the wrong thing

Gu et al. (2017) poisoned the LISA traffic sign dataset (Møgelmose et al. 2014):



Train object detector

class	Baseline F-RCNN clean	yellow square clean	yellow square backdoor
stop	89.7	87.8	N/A
speedlimit	88.3	82.9	N/A
warning	91.0	93.3	N/A
stop sign → speed-limit	N/A	N/A	90.3
average %	90.0	89.3	N/A

(Gu et al. 2017)

Test with Post-It

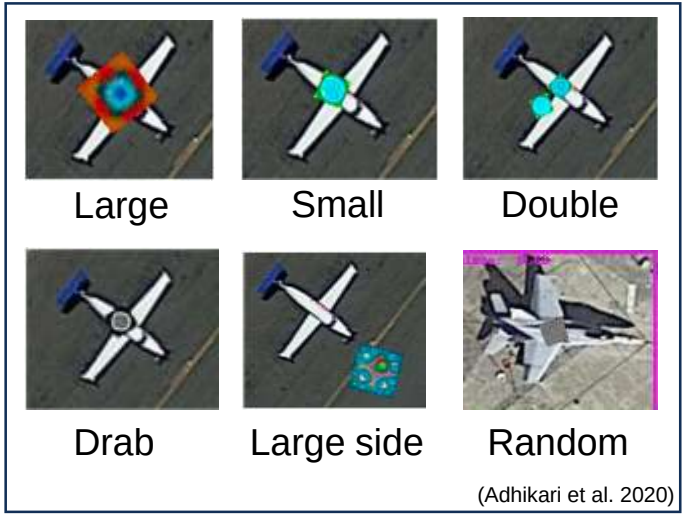
(Gu et al. 2017)

Do the wrong thing (object detection)

Adhikari et al. (2020) attack YOLOv2 on DOTA with physically realizable patches.

- YOLOv2 minimizes {Bounding box loss} + {Confidence loss} + {Classification loss}
- Thys et al. (2019) iteratively targets {Confidence loss} in YOLOv2
  - Start with a random pattern
  - Evaluate the {Confidence loss} on the random pattern
  - Update the random pattern to increase {Confidence loss}
  - Repeat
- Adhikari et al. (2020) uses Thys et al. (2019) to attack

  YOLOv2 pretrained on DOTA (76.9% AP):
  - Large: 5.58% AP
  - Small: 37.8% AP
  - Large side: 83.3% AP



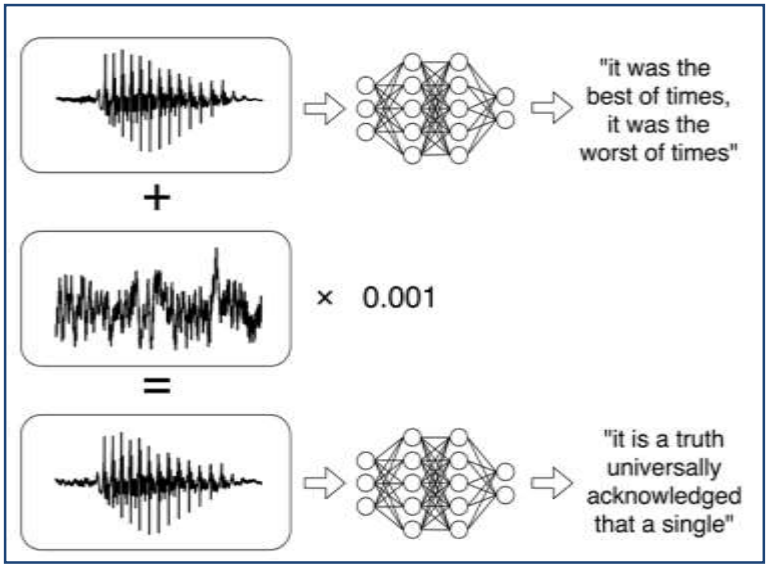
(Adhikari et al. 2020)

Do the wrong thing (audio transcription)

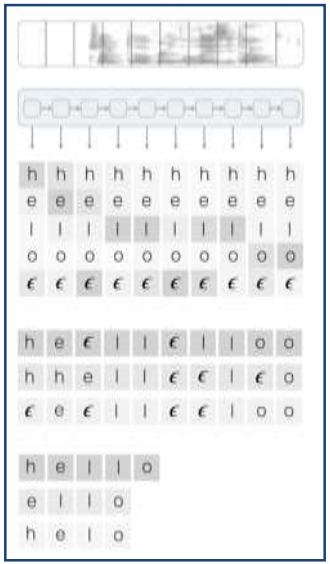
Carlini & Wagner (2018) attack Mozilla DeepSpeech with audio adversarial examples.

“without the dataset the article is useless”

“okay google browse to evil dot com”



(Carlini & Wagner 2018)



(Hannun, 2017)

1. Start with a random pattern

2. Evaluate the loss function of example + pattern

3. Update the pattern to move towards the target phrase
4. Repeat 5,000 times

Do the wrong thing (speaker identification)

Li et al. (2020) attacks X-Vector (Synder et al., 2018) speaker recognition with physically realizable (over the air) adversarial examples. 50% attack success rate.

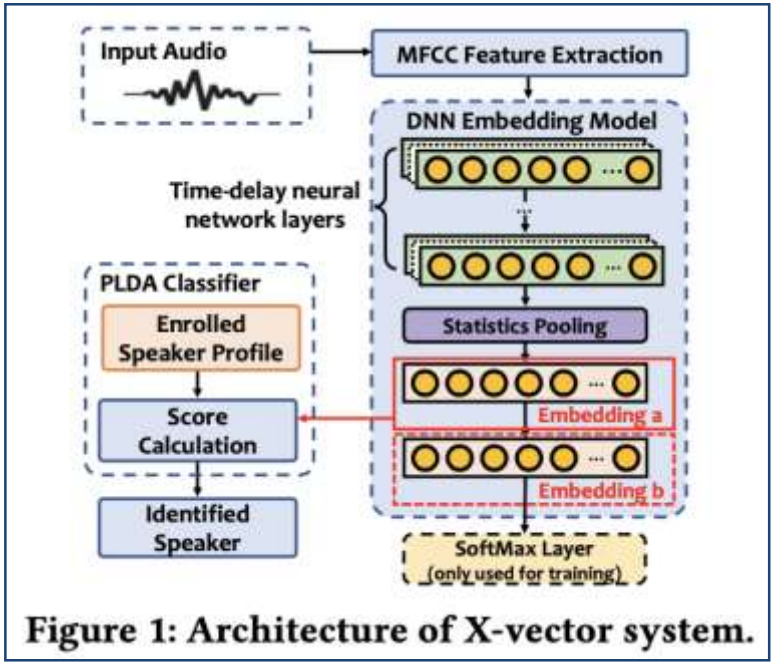


Figure 1: Architecture of X-vector system.

(Li et al. 2020)

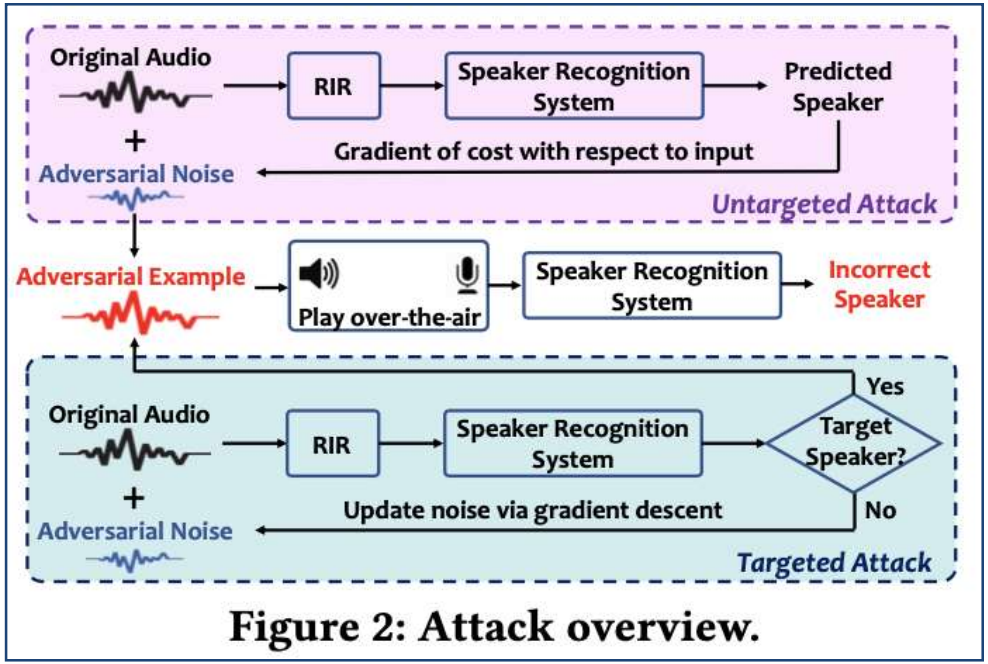


Figure 2: Attack overview.

(Li et al. 2020)

Reveal the wrong thing (model inversion)

Fredrickson et al. (2015):

- Trained simple classifiers on the AT&T Faces dataset (Samaria & Harter, 1994)

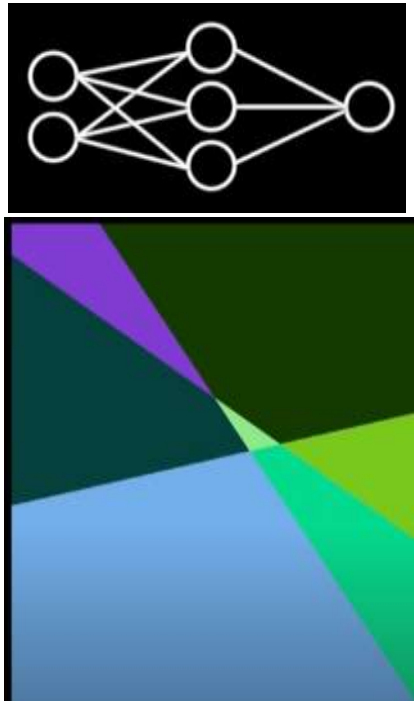
(Samaria & Harter, 1994)

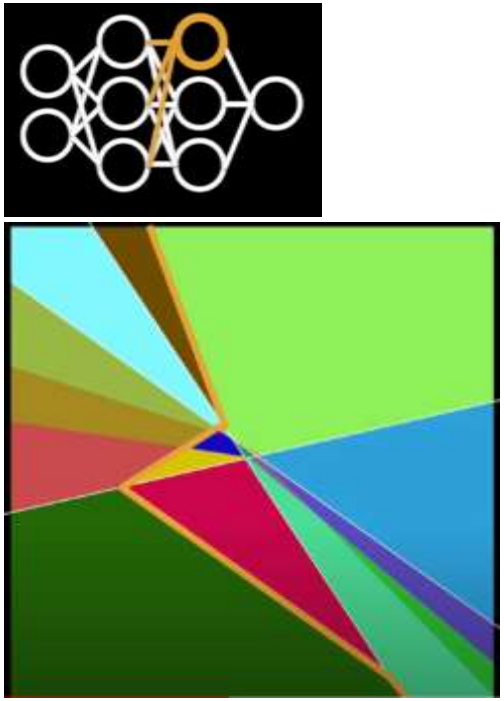
- Generated examples to target a particular class (person)

- Amazon Mturk workers identified inverted examples with > 80% accuracy.

Reveal the wrong thing (cryptanalytic extraction)

Carlini et al. (2020) demonstrates extraction of a trained model with only query access.





Architecture	Parameters
784-32-1	25,120
784-128-1	100,480
10-10-10-1	210
10-20-20-1	420
40-20-10-10-1	1,110
80-40-20-1	4,020

Fully connected networks with ReLU, recovered to machine precision with $\approx 1M$ queries

Three ways to attack an ML system

Beieler (2019) taxonomy: Make a machine learning system…

- **Learn** the wrong thing.
- **Do** the wrong thing.
- **Reveal** the wrong thing.

Other, more detailed taxonomies:

- NIST (Tabassi et al., 2019)
  - Currently revising, comment period closed in Jan. 2020
- Kumar et al. (2019)
  - Kumar and Rodriguez are working on a MITRE ATT&CK framework for AML
- Biggio and Roli (2018)
  - Canonical source, highly cited

Adversarial Machine Learning for the DoD: Objectives

By the end of the talk, you will be able to:

1. Give a 10k ft overview of how a machine learning system learns a task.
2. Identify the three ways that an adversary can attack an ML system.
3. **Identify the nine concerns that a defender may need to address to defend an ML system.**

Defending ML

Defend an ML system from

1. Learning the wrong thing
2. Doing the wrong thing
3. Revealing the wrong thing

Question 0: Defend from whom?

Threat modeling (Shevchenko et al., 2018)

- Model of Attacker
  - *Profile*: script kiddie, academic researcher, …
  - *Tools available:* IBM ART, custom toolchain, …
  - *Access:* model access, query access, published paper describing system (with or without code), …
- Model of system
- Catalog of threats

Defend from learning the wrong thing

IARPA TrojAI

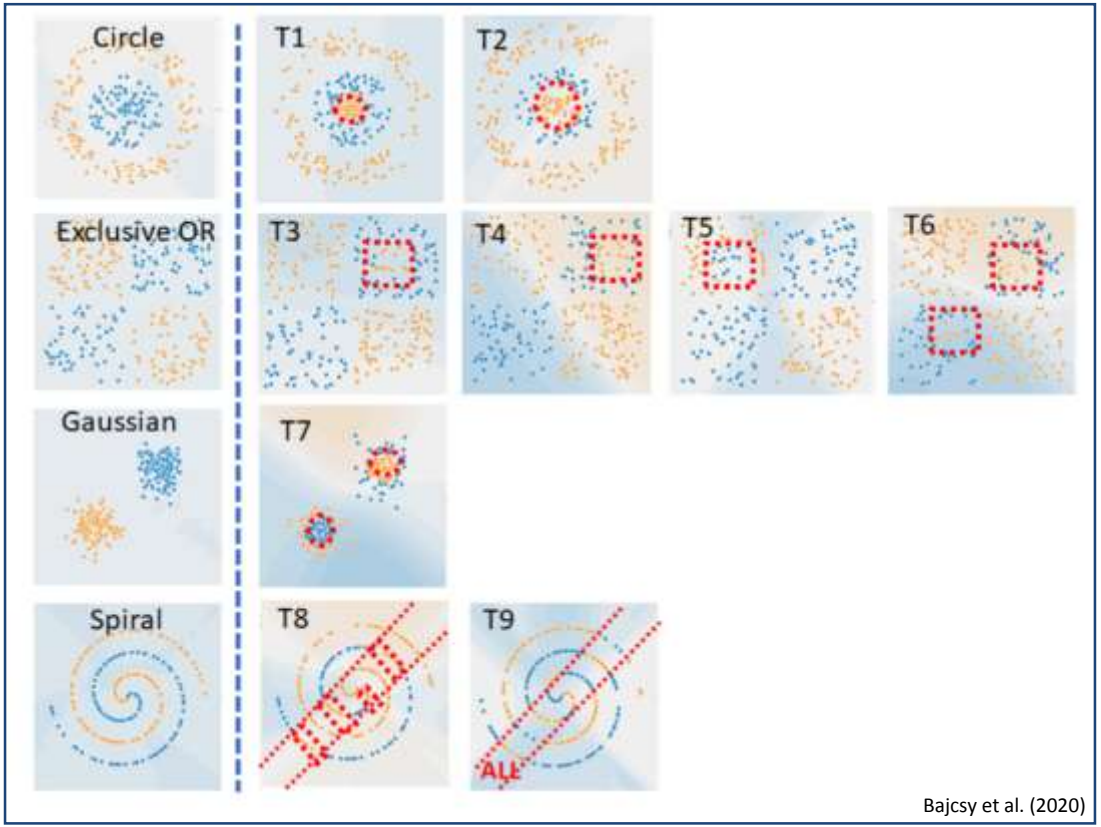
PM: Jeff Alstott

Considers trojans in three ML tasks

- Image classification
- Reinforcement learning
- Object detection

Leaderboard:

https://pages.nist.gov/trojai



Bajcsy et al. (2020)

Defend from doing the wrong thing

DARPA GARD

PM: Bruce Draper

Focuses on:

- Sensor-based AI systems (audio, images, video)
- Physically realizable evasion attacks (do)
- Poisoning attacks (learn)

Standard ML:

- Minimize expected loss during training

Attack a trained model:

- Generate examples to maximize expected loss of trained model

Train a defended model:

- Minimax estimation: include the adversary in your training

Tools provided:

- Reference implementation of attacks and defenses

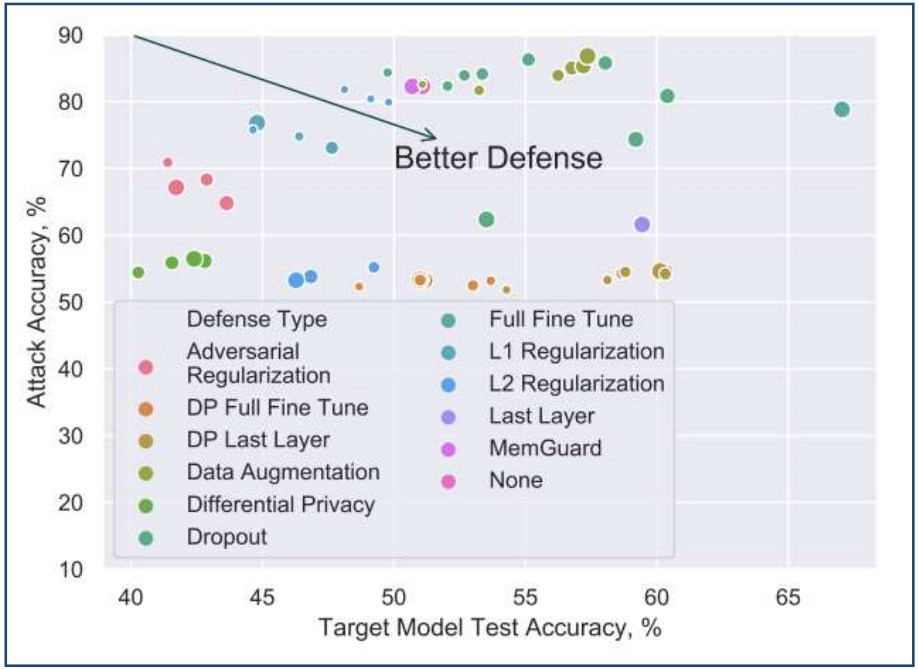
- Test harness to evaluate defended models against credible adversaries (scenarios developed by MITRE)

Defend from revealing the wrong thing

Mainly academic work in this space.

A few interesting papers:

- Model inversion: Mejia et al. (2019)
  - models robust to adversarial examples are more susceptible to model inversion
- Membership inference: Choo et al. (2020)
  - developed a label only attack capable of defeating all defenses except differential privacy and strong regularization



Choo et al. (2020)

Identify the nine concerns…

0. Threat modeling
1. Train to enforce learn policy, verify learn policy.
2. Train to enforce do policy, verify do policy.
3. Train to enforce reveal policy, verify reveal policy.
4. …

Train, but Verify: Multiple policies

Train \ Verify	Verify "learn" policy	Verify "do" policy	Verify "reveal" policy
Train to enforce "learn" policy	(1) IARPA TrojAI		
Train to enforce "do" policy		(2) DARPA GARD	(4)
Train to enforce "reveal" policy			(3) Academic work

(4) Helland & VanHoudnos (2020) Train to enforce "do" policy, Verify "reveal" policy

- Intuition and theoretical connections
- Adversarial walks: Revealing characteristics of the data from a trained model

Train do, Verify reveal: Intuition and theoretical connections

Fredrickson et al. (2016)

Introduced model inversion

Tsipras et al. (2018)

First identified that adversarial examples in robust models are recognizable as target.

Vanilla: Panda → Gibbon

(Goodfellow et al. 2014)

Robust: Turtle → Bird

(Tsipras et al. 2018)

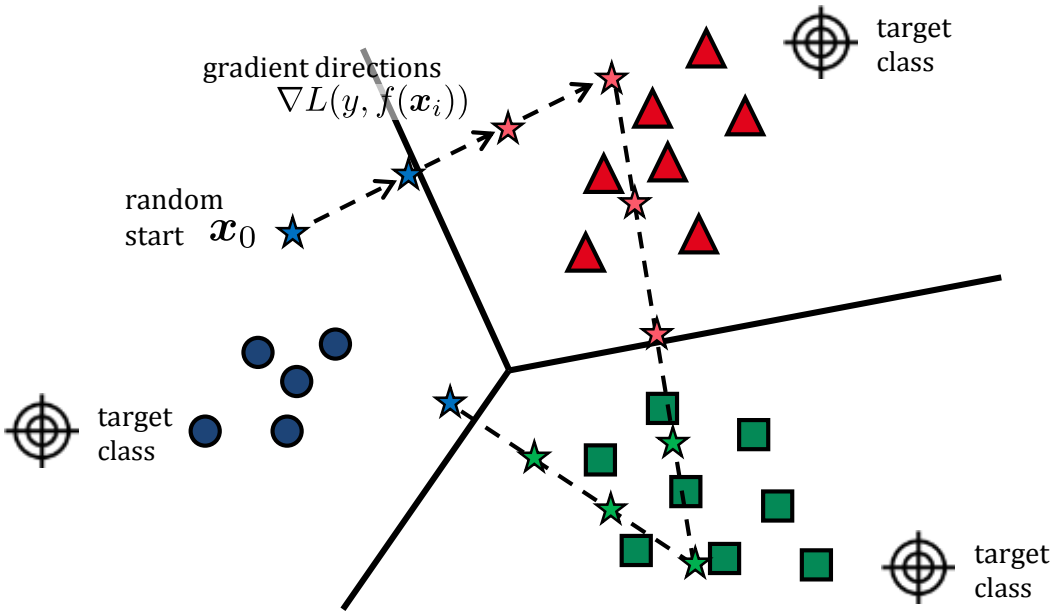
Mejia et al. (2019)

First to connect model inversion with large-ϵ PGD attacks.

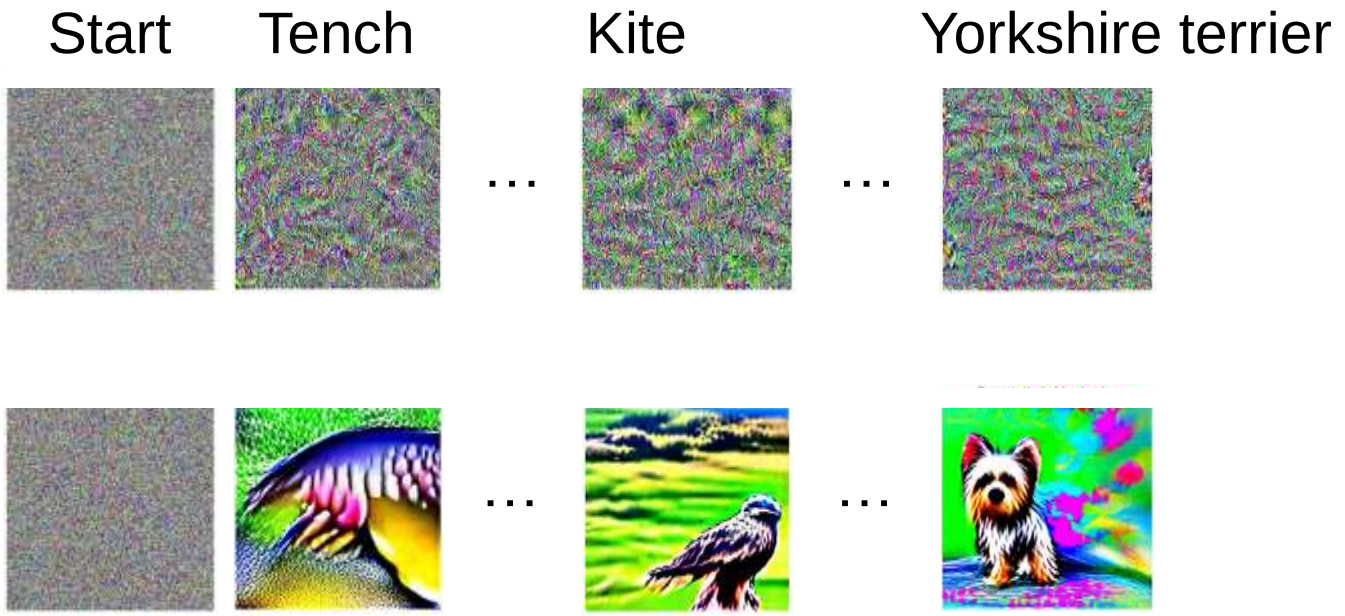
Empirically demonstrated that vanilla models do not have recognizable inversions, but robust models do.

(Mejia et al. 2019)

Train do, Verify reveal: Adversarial Walks



Adversarial Walks



* As the walk “burns in” classes are more recognizable

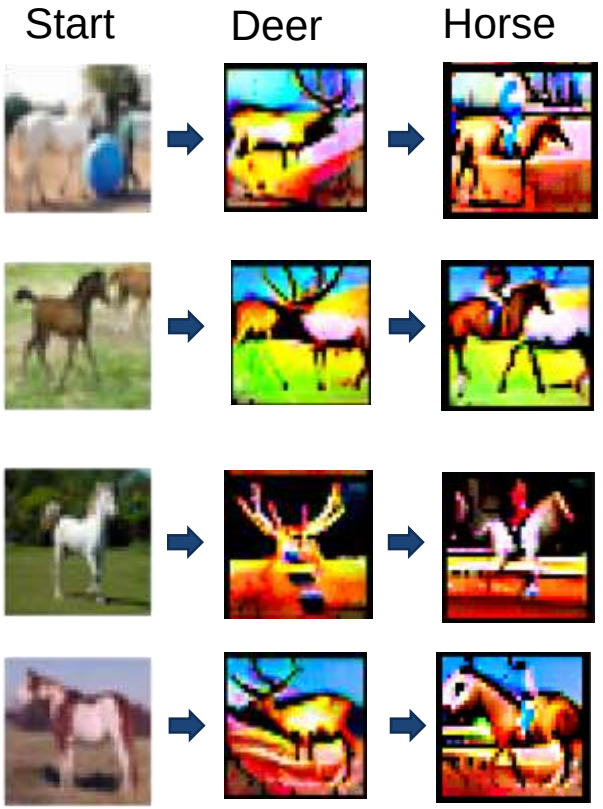
Revealing characteristics of the training data

Considers two security policies:

- High stakes decisions require a “do the right thing” policy, e.g. Madry PGD Training.
- Proprietary data collection requires a “do not reveal characteristics of the training data” policy.

Threat model:

- *Profile:* Script Kiddie
- *Access:* Model access
- *Tooling:* IBM ART 360

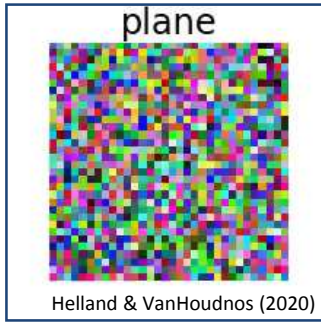


Recovers the presence of riders in the CIFAR 10 horse class (about 20% of examples).

Train, but Verify: Multiple policies

Train \ Verify	Verify “learn” policy	Verify “do” policy	Verify “reveal” policy
Train to enforce “learn” policy	Pick one: Nine things to consider		
Train to enforce “do” policy			
Train to enforce “reveal” policy			
Train … “do” and “reveal”	Pick two: Future work		
Train … “do” and “learn”			
Train … “reveal” and “learn”			
Train … “learn”, “do”, and “reveal”	Pick three: Aspirational		

Adversarial Machine Learning for the DoD: Objectives



Helland & VanHoudnos (2020)

By the end of the talk, you will be able to:

1. Give a 10k ft overview of how a machine learning system learns a task.
2. Identify the three ways that an adversary can attack an ML system.
3. Identify the nine concerns that a defender may need to address to defend an ML system.

Train \ Verify	"learn"	"do"	"reveal"
"learn"	Nine things to consider		
"do"			
"reveal"			
"do" and "reveal"	Future work		
"do" and "learn"			
"reveal" and "learn"			
"learn", "do", and "reveal"	Aspirational		

Carnegie Mellon University
Software Engineering Institute

Dr. Nathan VanHoudnos (van-HOD-ness)

nmvanhoudnos@sei.cmu.edu

nathan.m.vanhoudnos.ctr@mail.smil.mil

nathan.vanhoudnos_ctr@af.ic.gov

References (1/4)

A. Adhikari *et al.*, “Adversarial Patch Camouflage against Aerial Detection,” *arXiv:2008.13671 [cs]*, Aug. 2020, Accessed: Sep. 16, 2020. [Online]. Available: http://arxiv.org/abs/2008.13671.

P. Bajcsy, N. J. Schaub, and M. Majurski, “Neural Network Calculator for Designing Trojan Detectors,” *arXiv:2006.03707 [cs]*, Jun. 2020, Accessed: Sep. 17, 2020. [Online]. Available: http://arxiv.org/abs/2006.03707.

J. Beieler, “AI Assurance and AI Security: Definitions and Future Directions,” presented at the Adversarial Machine Learning Technical Exchange, Rockville, MD, Sep. 24, 2019, [Online]. Available: https://cra.org/ccc/wp-content/uploads/sites/2/2020/02/John-Beieler_AISec_AAAS.pdf.

B. Biggio and F. Roli, “Wild Patterns: Ten Years After the Rise of Adversarial Machine Learning,” in *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*, New York, NY, USA, Jan. 2018, pp. 2154–2156, doi: 10.1145/3243734.3264418.

N. Carlini and D. Wagner, “Audio Adversarial Examples: Targeted Attacks on Speech-to-Text,” in *2018 IEEE Security and Privacy Workshops (SPW)*, May 2018, pp. 1–7, doi: 10.1109/SPW.2018.00009.

N. Carlini, M. Jagielski, and I. Mironov, “Cryptanalytic Extraction of Neural Network Models,” *arXiv:2003.04884 [cs]*, Jul. 2020, Accessed: Sep. 17, 2020. [Online]. Available: http://arxiv.org/abs/2003.04884.

C. A. C. Choo, F. Tramer, N. Carlini, and N. Papernot, “Label-Only Membership Inference Attacks,” *arXiv:2007.14321 [cs, stat]*, Jul. 2020, Accessed: Sep. 17, 2020. [Online]. Available: http://arxiv.org/abs/2007.14321.

References (2/4)

M. Fredrikson, S. Jha, and T. Ristenpart, “Model Inversion Attacks that Exploit Confidence Information and Basic Countermeasures,” in *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security - CCS ’15*, Denver, Colorado, USA, 2015, pp. 1322–1333, doi: 10.1145/2810103.2813677.

A. Galyardt, J. Spring, and N. VanHoudnos, “Comments on NISTIR 8269 (A Taxonomy and Terminology of Adversarial Machine Learning).,” Software Engineering Institute, Carnegie Mellon University, Jan. 2020. [Online]. Available: https://resources.sei.cmu.edu/library/asset-view.cfm?assetid=637327.

A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, “Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks,” in *Proceedings of the 23rd international conference on Machine learning*, New York, NY, USA, Jun. 2006, pp. 369–376, doi: 10.1145/1143844.1143891.

T. Gu, B. Dolan-Gavitt, and S. Garg, “BadNets: Identifying Vulnerabilities in the Machine Learning Model Supply Chain,” *arXiv:1708.06733 [cs]*, Mar. 2019, Accessed: Sep. 16, 2020. [Online]. Available: http://arxiv.org/abs/1708.06733.

A. Hannun *et al.*, “Deep Speech: Scaling up end-to-end speech recognition,” Dec. 2014, Accessed: Sep. 16, 2020. [Online]. Available: https://arxiv.org/abs/1412.5567v2.

A. Hannun, “Sequence Modeling with CTC,” *Distill*, vol. 2, no. 11, p. e8, Nov. 2017, doi: 10.23915/distill.00008.

J. Helland and N. VanHoudnos, “On the interpretability of adversarial robust models with applications to data privacy,” presented at the Joint Statistical Meeting, 2020.

References (3/4)

R. S. S. Kumar, D. O. Brien, K. Albert, S. Viljöen, and J. Snover, “Failure Modes in Machine Learning Systems,” *arXiv:1911.11034 [cs, stat]*, Nov. 2019, Accessed: Sep. 16, 2020. [Online]. Available: http://arxiv.org/abs/1911.11034.

Z. Li, C. Shi, Y. Xie, J. Liu, B. Yuan, and Y. Chen, “Practical Adversarial Attacks Against Speaker Recognition Systems,” in *Proceedings of the 21st International Workshop on Mobile Computing Systems and Applications*, Austin TX USA, Mar. 2020, pp. 9–14, doi: 10.1145/3376897.3377856.

F. A. Mejia *et al.*, “Robust or Private? Adversarial Training Makes Models More Vulnerable to Privacy Attacks,” *arXiv:1906.06449 [cs, stat]*, Jun. 2019, Accessed: Aug. 05, 2020. [Online]. Available: http://arxiv.org/abs/1906.06449.

A. Møgelmose, Dongran Liu, and M. M. Trivedi, “Traffic sign detection for U.S. roads: Remaining challenges and a case for tracking,” in *17th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, Oct. 2014, pp. 1394–1399, doi: 10.1109/ITSC.2014.6957882.

N. Papernot, P. McDaniel, A. Sinha, and M. P. Wellman, “SoK: Security and Privacy in Machine Learning,” in *2018 IEEE European Symposium on Security and Privacy (EuroS P)*, Apr. 2018, pp. 399–414, doi: 10.1109/EuroSP.2018.00035.

J. Redmon and A. Farhadi, “YOLO9000: Better, Faster, Stronger,” *arXiv:1612.08242 [cs]*, Dec. 2016, Accessed: Sep. 16, 2020. [Online]. Available: http://arxiv.org/abs/1612.08242.

F. S. Samaria and A. C. Harter, “Parameterisation of a stochastic model for human face identification,” in *Proceedings of 1994 IEEE Workshop on Applications of Computer Vision*, Dec. 1994, pp. 138–142, doi: 10.1109/ACV.1994.341300.

References (4/4)

N. Shevchenko, T. Chick, P. O'Riordan, T. Scanlon, and C. Woody, "Threat Modeling: A Summary of Available Methods." Accessed: Jul. 29, 2020. [Online]. Available: https://resources.sei.cmu.edu/library/asset-view.cfm?assetid=524448.

D. Snyder, D. Garcia-Romero, G. Sell, D. Povey, and S. Khudanpur, "X-Vectors: Robust DNN Embeddings for Speaker Recognition," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Apr. 2018, pp. 5329–5333, doi: 10.1109/ICASSP.2018.8461375.

E. Tabassi, K. Burns, M. Hadjimichael, A. Molina-Markham, and J. Sexton, "A Taxonomy and Terminology of Adversarial Machine Learning," National Institute of Standards and Technology, NIST Internal or Interagency Report (NISTIR) 8269 (Draft), Oct. 2019. doi: https://doi.org/10.6028/NIST.IR.8269-draft.

S. Thys, W. V. Ranst, and T. Goedeme, "Fooling Automated Surveillance Cameras: Adversarial Patches to Attack Person Detection," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, Long Beach, CA, USA, Jun. 2019, pp. 49–55, doi: 10.1109/CVPRW.2019.00012.

D. Tsipras, S. Santurkar, L. Engstrom, A. Turner, and A. Madry, "Robustness May Be at Odds with Accuracy," *arXiv:1805.12152 [cs, stat]*, Sep. 2019, Accessed: Sep. 17, 2020. [Online]. Available: http://arxiv.org/abs/1805.12152.

G.-S. Xia *et al.*, "DOTA: A Large-scale Dataset for Object Detection in Aerial Images," *arXiv:1711.10398 [cs]*, May 2019, Accessed: Sep. 16, 2020. [Online]. Available: http://arxiv.org/abs/1711.10398.