Adversarial Learning on Malware

Christopher Molloy, Ziad Mansour, Steven H. H. Ding

chris.molloy@queensu.ca, ziad.mansour@queensu.ca, ding@cs.queensu.ca
Queen's University
Kingston, Ontario, Canada K7L 2N8

Abstract. With the cybersecurity systems' growing dependence on machine learning models, it is important to understand how an individual, organization, or government may take advantage of, or deceive, these models. In order to build models that are robust against adversarial methods, we must first understand adversarial techniques. This area of research is known as adversarial learning, and it has seen massive growth over the last 15 years. Adversarial learning research is critical to the cybersecurity domain. With the increase in machine learning used in malware detection, an arms race between adversaries and network defenders has emerged. Adversarial learning on malware focuses on how malware can deceive malware detection models, and how malware detection models can be built against deception. The three sections of adversarial learning are *knowledge*, *space*, and *strategy*. *Knowledge* describes how much an adversary knows about the target system. *Space* refers to where the adversary is making their attack. *Strategy* refers to when the adversary is making their attack. Adversarial learning on malware has succeeded on a wide range of malware types that target many systems.

1 Definition

Adversarial learning on malware is the study of methods that force a malware detection predictive model to mis-classify malware as benign. This area includes developing malware predictive models that are robust against adversarial techniques.

2 Motivation and Background

Perhaps the most obvious weakness of data science is the relationship between the predictive model and the training data. All predictive models are dependent on their training set, and small differences made to known data can yield incorrect predictions (Jo & Bengio 2017). In the context of malware detection, small changes can be made to malware with the intent of the malware being misclassified by a detection model. Although adversarial learning was thought to only be science fiction (Stephenson & Dániell 1992), it became reality in 2004 when Dalvi et al. (2004) evaded linear classifiers designed to detect spam through simple evasive tactics. Dalvi et al. (2004) also proposed a spam classifier based

on an adversarial strategy. The study of adversarial learning grew from there, and now encompasses all learning in cyberspace (Lowd & Meek 2005*a*,*b*, Fogla et al. 2006). Since then, the adversarial learning domain has evolved from linear classifiers to evading deep learning algorithms (Biggio & Roli 2018).

The motivation behind studying adversarial learning is for better protection. To create machine learning models that are robust against evasive techniques one must research and be well versed in the field of adversarial learning. It has been shown that many critical machine learning systems, such as road sign reading systems, can be rendered useless through adversarial techniques (Gu et al. 2017). Adversarial learning can be seen as two sides of a coin. One side is the adversary side, that is trying to infiltrate a network. The other side is the defending side, those who are tasked with protecting a network.

For an adversary, their goal is to cause a machine learning model to make a specific prediction or classification regarding their malware. The adversaries purpose would fall into one of the three categories of the CIA triad: Confidentiality, in which the adversary is trying to learn private information about the machine learning model being used to defend the target; Integrity, where the machine learning model defending the target is made to wrongly predict or classify specific or all inputs; or Availability, which causes the machine learning model to no longer be available for regular use (Rosenberg et al. 2020).

For those defending a network, in order to keep their primary goal of keeping the network they are defending secure and reliable, they cannot allow any adversaries to successfully evade their system. The second goal of those who are tasked with defending a network is privacy, because an adversary may try to attack the model with the intent of reverse-engineering the training data (Song et al. 2019). These two goals are met in two ways: developing new adversarial techniques, and proposing models that can "see through" those adversarial techniques. Those researching defense techniques often work with malware detection companies, streamlining the process of making the public more secure (Song et al. 2020).

3 Structure of Learning System

The structure of adversarial learning can be separated into three sections. These sections are knowledge, space, and strategies. All adversarial attacks can be described by a combination of one category per section.

In adversarial learning, there are three categories of knowledge that describe how much an adversary knows about a target network. The knowledge spaces are *white-box*, *black-box*, and *grey-box*. A white-box attack implies that the attackers know everything about the target system. A black-box attack is an attack where the adversary has no knowledge of the target system. A grey-box attack is one in which the attacker has some knowledge of the target system. It is a guideline to defenders of networks to assume all attacks carried out by attackers are white-box attacks, this ensures that all systems built by network defenders will be robust in the case that their system is leaked (Carlini et al. 2019). This methodology

is known as Kerckhoffs' principal (Kerckhoffs 1883). Although the defenders should assume their system to known to the world, in most cases the adversary's knowledge is categorized as grey-box.

The two spaces in which the attack takes place are the feature space and the problem space. Feature space attacks refer to when an adversary will use algorithms to make modifications to the features of a malicious file with the intent of convincing a machine learning model to misclassify the malware as benignware. Attacking in the problem space would refer to the space in which the malware exists. A problem space attack is one in which the original malware is not modified, but an entirely new file is created, this new file has been changed completely with the intent of appearing benign to the target classifier (Pierazzi et al. 2020). Some examples of adversarial modifications:

- Append benign content to the end the file (Song et al. 2020, Lowd & Meek 2005*b*, Anderson et al. 2018)
- Change the section name to the name of benign binaries (Song et al. 2020, Anderson et al. 2018)
- Replace instruction sequence (Song et al. 2020)
- Append random bytes to unused space at the end of a section (Song et al. 2020, Anderson et al. 2018)
- Creating new and unused sections (Song et al. 2020, Anderson et al. 2018)
- Change signed certificate (Song et al. 2020)
- Change debug information (Song et al. 2020)
- Adding a function to the import address table that is not referenced (Anderson et al. 2018, Hu & Tan n.d.)
- Creating new entry point which immediately jumps to the original entry point (Anderson et al. 2018)

There are three strategies that an adversary can have when attacking a target network. These three strategies are evasion, poisoning, and model extraction. Evasion is the strategy of fooling the classifier once it has already been trained and is deployed, where an adversary will try to have their malware misclassified as benignware by the defending model (Khasawneh et al. 2017). The second strategy is poisoning. Poisoning is done during the learning stage of the defending model and is when an adversary will "poison" the training data of the learning-based malware detection models to misclassify malware (Chen et al. 2018).

The final strategy that an adversary can use is model extraction. Model extraction is when an adversary creates a new model that is capable of replicating the functionality of a target system. With this new model an adversary can modify their own malware to ensure that it will be misclassified by the target model (Takemura et al. 2020).

This wide range of different structures allows adversarial attacks to be done against many different types of networks. Files that adversarial learning work with include Android malware (Demontis et al. 2019, Yang et al. 2017), Windows malware (Kolosnjaji et al. 2018, Rosenberg et al. 2018), malicious PDFs (Biggio et al. 2013, Dang et al. 2017, Laskov & Srndic 2011, Maiorca et al. 2012, 2013, Xu et al. 2016), and malicious JavaScript (Fass et al. 2019).

4 Cross References

Below is a list of other chapters within the Encyclopedia of Machine Learning and Data Science' that discuss topics related to adversarial learning on malware.

Title	Section
Conditional Generative Adversarial Networks	Deep Generative Models
Generative Adversarial Networks	Deep Generative Models
Policy Gradient Methods	Reinforcement Learning
Deep Learning	Artificial Neural Network
Markov Decision Processes	Behavioral Cloning and Imitation Learning
Markov Chain Monte Carlo	Data Mining

5 Recommended Reading

Biggio, B. & Roli, F. (2017), 'Wild patterns: Ten years after the rise of adversarial machine learning', *Pattern Recognit.* **84**.

Demontis, A., Melis, M., Biggio, B., Maiorca, D., Arp, D., Rieck, K., Corona, I., Giacinto, G. & Roli, F. (2019), 'Yes, machine learning can be more secure! A case study on android malware detection', *IEEE Trans. Dependable Secur.Comput.* **16**(4).

Kolosnjaji, B., Demontis, A., Biggio, B., Maiorca, D., Giacinto, G., Eckert, C. & Roli, F. (2018), 'Adversarial malware binaries: Evading deep learning for malware detection in executables',*in* '26th European Signal Processing Conference, EUSIPCO 2018, Roma, Italy, September 3-7, 2018' **IEEE**.

Laskov, P. & Srndic, N. (2011), Static detection of malicious javascript-bearing PDF documents, *in* R. H. Zakon, J. P. McDermott & M. E. Locasto, eds, 'Twenty-Seventh Annual Computer Security Applications Conference, ACSAC 2011, Orlando, FL, USA, 5-9 December 2011', ACM.

Lowd, D. & Meek, C. (2005*a*), Adversarial learning, *in* R. Grossman, R. J.Bayardo & K. P. Bennett, eds, 'Proceedings of the Eleventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Chicago,Illinois, USA, August 21-24, 2005', ACM.

Pierazzi, F., Pendlebury, F., Cortellazzi, J. & Cavallaro, L. (2020), Intriguing properties of adversarial ML attacks in the problem space, *in* '2020 IEEE Symposium on Security and Privacy, SP 2020, San Francisco, CA, USA, May18-21, 2020', IEEE.

Anderson, H. S., Kharkar, A., Filar, B., Evans, D. & Roth, P. (2018), 'Learning to evade static pe machine learning malware models via reinforcement learning', *arXiv preprint arXiv:1801.08917* .

Biggio, B., Corona, I., Maiorca, D., Nelson, B., Srndic, N., Laskov, P., Giacinto, G. & Roli, F. (2013), Evasion attacks against machine learning at test time, *in* H. Blockeel, K. Kersting, S. Nijssen & F. Zelezný, eds, 'Proceedings of Machine Learning and Knowledge Discovery in Databases - European Conference, ECML PKDD 2013, Prague, Czech Republic, September 23-27, 2013, Proceedings, Part III', Vol. 8190 of *Lecture Notes in Computer Science*, Springer.

Biggio, B. & Roli, F. (2018), 'Wild patterns: Ten years after the rise of adversarial machine learning', *Pattern Recognit.* **84**.

Carlini, N., Athalye, A., Papernot, N., Brendel, W., Rauber, J., Tsipras, D., Goodfellow, I., Madry, A. & Kurakin, A. (2019), 'On evaluating adversarial robustness', *arXiv* pp. arXiv–1902.

Chen, S., Xue, M., Fan, L., Hao, S., Xu, L., Zhu, H. & Li, B. (2018), 'Automated poisoning attacks and defenses in malware detection systems: An adversarial machine learning approach', *Comput. Secur.* **73**.

Dalvi, N., Domingos, P., Sanghai, S. & Verma, D. (2004), Adversarial classification, *in* 'Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining'.

Dang, H., Huang, Y. & Chang, E. (2017), Evading classifiers by morphing in the dark, *in* B. M. Thuraisingham, D. Evans, T. Malkin & D. Xu, eds, 'Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security, CCS 2017, Dallas, TX, USA, October 30 - November 03, 2017', ACM.

Demontis, A., Melis, M., Biggio, B., Maiorca, D., Arp, D., Rieck, K., Corona, I., Giacinto, G. & Roli, F. (2019), 'Yes, machine learning can be more secure! A case study on android malware detection', *IEEE Trans. Dependable Secur. Comput.* **16**(4).

Fass, A., Backes, M. & Stock, B. (2019), Hidenoseek: Camouflaging malicious javascript in benign asts, *in* L. Cavallaro, J. Kinder, X. Wang & J. Katz, eds, 'Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security, CCS 2019, London, UK, November 11-15, 2019', ACM.

Fogla, P., Sharif, M. I., Perdisci, R., Kolesnikov, O. M. & Lee, W. (2006), Polymorphic blending attacks, *in* A. D. Keromytis, ed., 'Proceedings of the 15th USENIX Security Symposium, Vancouver, BC, Canada, July 31 - August 4, 2006', USENIX Association.

Gu, T., Dolan-Gavitt, B. & Garg, S. (2017), 'Badnets: Identifying vulnerabilities in the machine learning model supply chain', *arXiv* pp. arXiv–1708.

Hu, W. & Tan, Y. (n.d.), 'Generating adversarial malware examples for black-box attacks based on gan'.

Jo, J. & Bengio, Y. (2017), 'Measuring the tendency of cnns to learn surface statistical regularities', *arXiv* pp. arXiv–1711.

Kerckhoffs, A. (1883), 'Military cryptography', *French Journal of Military Science* .

Khasawneh, K. N., Abu-Ghazaleh, N. B., Ponomarev, D. & Yu, L. (2017), RHMD: evasion-resilient hardware malware detectors, *in* H. C. Hunter, J. Moreno, J. S. Emer & D. Sánchez, eds, 'Proceedings of the 50th Annual IEEE/ACM International Symposium on Microarchitecture, MICRO 2017, Cambridge, MA, USA, October 14-18, 2017', ACM.

Kolosnjaji, B., Demontis, A., Biggio, B., Maiorca, D., Giacinto, G., Eckert, C. & Roli, F. (2018), Adversarial malware binaries: Evading deep learning for malware detection in executables, *in* 'Proceedings of the 26th European Signal Processing Conference, EUSIPCO 2018, Roma, Italy, September 3-7, 2018', IEEE.

Laskov, P. & Srndic, N. (2011), Static detection of malicious javascript-bearing PDF documents, *in* R. H. Zakon, J. P. McDermott & M. E. Locasto, eds, 'Twenty-Seventh Annual Computer Security Applications Conference, ACSAC 2011, Orlando, FL, USA, 5-9 December 2011', ACM.

Lowd, D. & Meek, C. (2005*a*), Adversarial learning, *in* R. Grossman, R. J. Bayardo & K. P. Bennett, eds, 'Proceedings of the Eleventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Chicago, Illinois, USA, August 21-24, 2005', ACM.

Lowd, D. & Meek, C. (2005*b*), Good word attacks on statistical spam filters, *in* 'CEAS 2005 - Second Conference on Email and Anti-Spam, July 21-22, 2005, Stanford University, California, USA'.

Maiorca, D., Corona, I. & Giacinto, G. (2013), Looking at the bag is not enough to find the bomb: an evasion of structural methods for malicious PDF files detection, *in* K. Chen, Q. Xie, W. Qiu, N. Li & W. Tzeng, eds, 'Proceedings of the 8th ACM Symposium on Information, Computer and Communications Security, ASIA CCS', ACM.

Maiorca, D., Giacinto, G. & Corona, I. (2012), A pattern recognition system for malicious PDF files detection, *in* P. Perner, ed., 'Proceedings of Machine Learning and Data Mining in Pattern Recognition - 8th International Conference, MLDM 2012, Berlin, Germany, July 13-20, 2012. Proceedings', Vol. 7376 of *Lecture Notes in Computer Science*, Springer.

Pierazzi, F., Pendlebury, F., Cortellazzi, J. & Cavallaro, L. (2020), Intriguing properties of adversarial ML attacks in the problem space, *in* '2020 IEEE Symposium on Security and Privacy, SP 2020, San Francisco, CA, USA, May 18-21, 2020', IEEE.

Rosenberg, I., Shabtai, A., Elovici, Y. & Rokach, L. (2020), 'Adversarial learning in the cyber security domain', *arXiv e-prints* pp. arXiv–2007.

Rosenberg, I., Shabtai, A., Rokach, L. & Elovici, Y. (2018), Generic black-box end-to-end attack against state of the art API call based malware classifiers, *in* M. Bailey, T. Holz, M. Stamatogiannakis & S. Ioannidis, eds, 'Proceedings of the 21st International Symposium on the Research in Attacks, Intrusions, and Defenses, RAID 2018', Vol. 11050 of *Lecture Notes in Computer Science*, Springer.

Song, L., Shokri, R. & Mittal, P. (2019), Privacy risks of securing machine learning models against adversarial examples, *in* L. Cavallaro, J. Kinder, X. Wang & J. Katz, eds, 'Proceedings of the 2019 ACM SIGSAC Conference on Com-

puter and Communications Security, CCS 2019, London, UK, November 11-15, 2019', ACM.

Song, W., Li, X., Afroz, S., Garg, D., Kuznetsov, D. & Yin, H. (2020), 'Automatic generation of adversarial examples for interpreting malware classifiers', *arXiv* pp. arXiv–2003.

Stephenson, N. & Dániell, K. (1992), *Snow crash*, Metropolis Media.

Takemura, T., Yanai, N. & Fujiwara, T. (2020), 'Model extraction attacks against recurrent neural networks', *arXiv* pp. arXiv–2002.

Xu, W., Qi, Y. & Evans, D. (2016), Automatically evading classifiers: A case study on PDF malware classifiers, *in* '23rd Annual Network and Distributed System Security Symposium, NDSS 2016, San Diego, California, USA, February 21-24, 2016', The Internet Society.

Yang, W., Kong, D., Xie, T. & Gunter, C. A. (2017), Malware detection in adversarial settings: Exploiting feature evolutions and confusions in android apps, *in* 'Proceedings of the 33rd Annual Computer Security Applications Conference, Orlando, FL, USA, December 4-8, 2017', ACM.