Law and Adversarial Machine Learning

Ram Shankar Siva Kumar
Microsoft
Ram.Shankar@microsoft.com

David R. O'Brien
Berkman Klein Center for Internet and Society
dobrien@cyber.harvard.edu

Kendra Albert
Harvard Law School
kalbert@law.harvard.edu

Salomé Viljoen
Berkman Klein Center for Internet and Society
sviljoen@cyber.harvard.edu

Abstract

When machine learning systems fail because of adversarial manipulation, how should society expect the law to respond? Through scenarios grounded in adversarial ML literature, we explore how some aspects of computer crime, copyright, and tort law interface with perturbation, poisoning, model stealing and model inversion attacks to show how some attacks are more likely to result in liability than others. We end with a call for action to ML researchers to invest in transparent benchmarks of attacks and defenses; architect ML systems with forensics in mind and finally, think more about adversarial machine learning in the context of civil liberties. The paper is targeted towards ML researchers who have no legal background.

1 Introduction

Technology and the law are inextricably linked, and for either to be effective for society, the two must work together. As new technologies permit new potential harms, judges, legislatures, and regulators are on the spot for rationalizing law with technology. For adversarial machine learning, this process is just beginning - judges have not had much cause to determine the applicability of existing law to attacks on machine learning, nor have specific laws been passed to regulate machine learning systems.

But the arms-length relationship between Machine Learning (ML) attacks and the law seem unlikely to continue. Despite the vulnerabilities that this community has demonstrated since 2004, as Biggio and Roli [2018] notes, ML is at the core of many critical systems including healthcare, defense, and finance . Given this, when such systems fail or get compromised, it seems inevitable that law and adversarial machine learning are on a crash course towards each other. But the relationship is under-theorized. In response to Tramèr et al. [2016] work on model stealing, one of the affected companies responded: "Said another way, even if stealing software were easy, there is still an important disincentive to do so in that it violates intellectual property law." (see Cetinsoy [2016]) Such a statement assumes, without proof, that model stealing can be sanctioned by existing intellectual property law. As we discuss below, it is entirety possible that it won't be. *The goal of this paper is to begin to explore how for some attacks, existing law may provide protection for ML models, but in others, there may be less protection for machine learning systems than practitioners expect.*

We have structured the paper thus: we discuss supply chain, perturbation and poisoning attacks via the Computer Fraud and Abuse Act (Section II); model stealing and model inverstion in the lens of U.S. intellectual property law (Section III); applicability of civil liability law to adversarial ML (Section IV), finally ending with some recommendations for ML researcher (Section V). Since our paper is tailored to the ML community who have no legal background, we only provide a sampling of the legal concepts as it relates to ML attacks.

32nd Conference on Neural Information Processing Systems (NIPS 2018), Montréal, Canada.

2 Cybersecurity law and Supply Chain, Perturbation, Poisoning attacks

In this section we discuss supply chain, perturbation and poisoning attacks through the lens of the Computer Fraud and Abuse Act (CFAA), the hallmark federal “anti-hacking” statute in the US. CFAA was originally enacted in 1984, and inspired in part by the film "War Games" and has surprisingly kept up with 30 years of technological changes. Simply stated, the CFAA broadly prohibits individuals from intentionally accessing computers without authorization, exceeding authorized access on a computer, and causing damage to computers without authorization. Violators of these provisions may face lawsuits by the victims and criminal prosecution. US attorneys have successfully used the law to prosecute a wide range of activities – some would argue too expansively – thanks in large part to its broadly-worded prohibitions as noted by Curtiss [2016]. That said, adversarial ML might be different. As Calo et al. [2018] point out, adversarial ML attacks present definitional challenges that raise questions about whether the CFAA is up to task. Much of the CFAA is couched around whether access has occurred or a system damaged.The scenarios we explore below are intended to selectively illustrate both parity and disparity that might arise between the CFAA and an attack.

Scenario: Gu et al. [2017] propose attackers may target the ML supply chain by compromising the pre-trained models as they are downloaded from an insecure (HTTP) connection.
Legal commentary: A classic man-in-the-middle attack like this appears to be a straight-forward CFAA violation – the attacker knowingly accessed and altered the model in transit without authorization. Similarly, an attacker exploiting a buffer overflow vulnerability on OpenCV that results in misclassification as demonstrated by Xiao et al. [2017], likely violates the CFAA, since the attacker has accessed the platform and altered the integrity of the output by exploitation.

Scenario: Jagielski et al. [2018] poison a healthcare dataset quite effectively that a tenth of the patients have their dosages changed by 359%.
Legal commentary: A poisoning attack like this could plausibly be a violation of the CFAA. The strongest argument may be that in carrying out the attack, the adversary transmitted a code (in this context, prosecutors could argue that code is the poisoned examples) that caused damage to the model in a way that disrupts the system. However, the analysis becomes less clear in cases where the purpose of the ML system is more open-ended or premised on interactive feedback, like Tay. When in principle do innocuous inputs become malicious? And, at what point does an ML system reach a state of being damaged?

Scenario: Papernot et al. [2015] propose to fool a bank’s image recognition system to misrecognize checks to higher value.
Legal commentary: Perturbation attacks may be covered under the CFAA as prosecutors could argue that the adversary knowingly transmitted code (in this case a modified image, which ultimately gets converted to code) that caused damage (in this case monetary damages suffered by the bank). By the same token, it can also be argued that tampering with stop signs in the context of autonomous cars is also a CFAA violation, since stickers (as seen in Evtimov et al. [2017]) are a form of transmitting code that causes damage to the autonomous car (a computer in the eyes of CFAA).
Banks (and other organizations) are also likely to have a Terms of Services (ToS) drafted by its lawyers which generally prohibit malicious activities. Several CFAA cases have turned on whether the activities in question were prohibited by a ToS agreement, which some courts have held can constitute exceeding authorized access under the CFAA. However, it is a controversial subject. In situations where the applicability of the CFAA may not be clear based on the statutory definitions, a ToS can bridge some of these gaps. Finally, since it is common for prosecutors to pursue higher charges, in addition to CFAA violation the adversary would also face wire fraud charges.

The takeaway from this section should be that the CFAA plausibly covers some supply chain, perturbation and poisoning attacks if its statutory language is interpreted in certain ways. Courts have struggled in similar cases in the past, as Calo et al. [2018] discuss in more detail, and it is far from clear whether they can consistently resolve these differences in future cases.

3 Copyright law and Model Inversion, Model Extraction attacks

To protect against model inversion and model stealing, ML practitioners may be tempted to turn to a different body of law – copyright law. However, unlike CFAA which is broad and open to wide interpretation, copyright law is more narrow and well-defined, and hence unlikely to provide as much coverage as the CFAA.

Scenario: Fredrikson et al. [2015] reconstruct part of the private training data using hill climbing on output probabilities
Legal Commentary: The ability of the owner, whose data was reconstructed, to get relief to under copyright would depend upon what exactly the training data was. In the United States, facts are not copyrightable, even if they are costly or time-consuming to gather. Copyright protection may attach to compilations or arrangements of factual information, however, it is unlikely that a reconstructed set of training data would necessarily share the same compilation or arrangement as the original: for instance, the reconstructed data could be approximations as in the case of Fredrikson et al. [2015]. So the owner of the dataset/model would be unlikely to be able to successfully sue an adversary that recovered part of a training dataset consisting of facts for copyright infringement.
On the other hand, images and audio are copyrightable, so, the owner would be more likely to succeed against an adversary that reproduced those. The question is murkier with regards to information derived from copyrightable materials, such as RGB pixel values, or general image characteristics.

Scenario: Tramèr et al. [2016] reconstruct a model hosted behind a prediction API
Legal Commentary: Copyright for software is an interest in a code as a "literary work", not for its functions. Therefore, although the code that runs a particular machine learning model might be protected by copyright, a reconstruction is unlikely to share the particular expression of code with the original, and thus reconstruction is unlikely to violate copyright law: for instance, even if both the original and the "stolen" model are decision trees as shown in Tramèr et al. [2016], their exact implementation may differ and thus the "stolen" model would not infringe copyright.
It is possible that in some circumstances, machine learning models may qualify as trade secrets, and that trade secret law could protect a model against reconstruction. However, in order to successfully sue for trade secret disclosure, an owner must show that they took reasonable precautions to prevent disclosure.

In the absence of intellectual property protection, one potential way to prevent model stealing would be to include a Terms of Service that specifically prohibits this, thereby establishing a contract with the API users. However, such contractual agreements only create rights against the users of the API – they might not help if an adversary releases a reconstructed model publicly. But in any case, model inversion and model extraction are attacks where existing law might not protect ML systems in the way that companies or researchers might expect.

4 Liability laws in the context of adversarial ML

In this section we attempt to answer the following question using tort law: When an ML product breaks down because of adversarial examples, who is liable? This question is not purely rhetorical: the European Union is set to release a liability and safety framework for ML systems by mid-2019 (see Commission [2018]) which could snowball into GDPR style regulation.

Scenario: Brundage et al. [2018] discuss how a drone's image recognition system could fail owing to adversarial examples and potentially cause damage. While the authors discuss this in the context of military drones, we will assume that the drone is consumer grade (as used in photography) to avoid complications with international laws.
Legal Commentary: The uncertainty here arises due to the interaction between a vulnerable product and a malicious actor. Gilmer et al. [2018] argue that as long as there is non-zero test error, adversarial examples will exist. If a drone vendor used a state of the art image recognition system (which is likely to have non-zero test error), was the manufacturer negligent? Software vendors have generally not been held liable for traditional software attacks under theories of product liability. Yet the novel nature and expanded scope of harms presented by ML products may pose new risks for this type of liability.

Part of the issue is that courts do not have industry standards with which to compare negligent versus responsible ML development practices. No established standard or industry wide practice for protecting against adversarial examples or reward hacking has been established. Another complicating factor is the interrelated nature of the ML ecosystem makes it difficult to establish which component caused the failure. Machine learning systems are built on a mix of open source libraries and commercial systems. For example, consider the (common) case wherein a vendor, say ,a drone manufacturer, reuses a model from academic researchers hosted in Caffee Model Zoo, ports it over in PyTorch and runs it on commercial cloud. When the drone fails and causes bodily harm, who is liable? The answer, as noted by Calo [2010] is not known and we may have to wait until such a case comes to trial to provide some insight into how blame will be assigned in this ecosystem.

5 Call to Action for ML Researchers:

Given the uncertainty in law in some adversarial ML attacks, here are three recommendations for ML researchers working in this space to assist lawyers and policy makers in creating reasonable, evidence-driven policy:

1. Benchmark Attacks and Defenses – There is a growing need for legal practioners and policy makers to understand how adversarial ML differs from traditional software attacks in ways that may inform how laws are interpreted and enforced. ML researchers can bring clarity to the situation by helping to prioritize the attacks and defenses they publish. This will both help inform the development of appropriate standards of care for systems that use machine learning, and provide practical guidance to engineers.

   In this spirit, whenever researchers publish a new defense against an attack, they might consider using tools like cleverhans (See Papernot et al. [2018]) , IBM's adversarial robustness toolkit Nicolae et al. [2018] and report shortcomings. The community should expand and invest in benchmarking efforts such as RobustML (see Mądry et al. [April 2018]). We found Goodfellow [2018], where defenses are stack ranked, useful to think about the progress of defenses in perturbation attacks.

   Benchmarks alone aren't sufficient: we also think there is a need for a framework to assess risk and prioritize adversarial ML threats realistically. Attackers need not perform perturbation attacks to evade ML systems as documented by Gilmer et al. [2018]. To address this we believe the ML community can take inspiration from threat modeling from software community **DREAD** (see Shostack [2014]) to prioritize software threats. It rates attacks based on the potential for **D**amage, **R**eliability of attack, the ease which an attacker can launch the **E**xploit, the scope of **A**ffected users, and the ease with which an attacker can **D**iscover the attack.
2. Architect for forensics – ML systems are currently not built with forensics in mind. From a legal perspective, forensics can lend clues to attack attribution and hence eventual prosecution. ML researchers should be thinking proactively about how to architect systems so that investigations are possible, including mechanisms to alert when the system is under adversarial attack, recommend appropriate logging, construct playbooks for incident response during an attack and formulate remediation plan to recover to from the adversarial attack.
3. Take into account civil liberties - Deployed ML systems have the ability to impact civil liberties and basic human rights such as freedom of expression and privacy. For instance, ML researchers should anticipate that oppressive governments could seek backdoors in consumer ML systems, as demonstrated by Chen et al. [2017], facilitate censorship and out political dissidents. On the same note, researchers must also think about the *dual use of adversarial examples* i.e. the benefits of adversarial examples. For example, dissidents in a totalitarian state should be able to evade facial detection using 3D printed glasses as shown by Sharif et al. [2016]. ML researchers should do their best to anticipate how ML systems and attacks can be used for the benefit and detriment of individuals.

6 Conclusion

Given the widespread usage of ML in real world applications, legal responses to adversarial ML attacks are important to society and inevitable. Some aspects of the law map onto attacks such as poisoning and perturbation, but for others, like model stealing, legal recourse is less clear. ML practitioners can bring clarity to this discussion by consindering benchmarking attacks and defenses; architecting ML systems with in built forensics, and be thoughtful about the *dual use of adversarial examples*

Acknowledgments

An interdisciplinary paper such as this would not have been possible without fruitful discussions and feedback from ML researchers (Aleksandr Madry, Momin Malik, Gretchen Greene, Sharon Gillett, Justin Gilmer), security experts (John Walton, John Lambert, Jeffrey Snover, Matt Swann) and lawyers/public policy experts (Woodrow Hartzog, Daniel Edelman, Ryan Calo, Yaniv Benhamou, Cristin Goodwin). Ram would also like to thank Andi Comissoneru, Sharon Xia, Steve Mott and the entire Azure Security Data Science team for holding the fort during his time away.

References

Battista Biggio and Fabio Roli. Wild patterns: Ten years after the rise of adversarial machine learning. *Pattern Recognition*, 84:317–331, 2018.

Miles Brundage, Shahar Avin, Jack Clark, Helen Toner, Peter Eckersley, Ben Garfinkel, Allan Dafoe, Paul Scharre, Thomas Zeitzoff, Bobby Filar, et al. The malicious use of artificial intelligence: Forecasting, prevention, and mitigation. *arXiv preprint arXiv:1802.07228*, 2018.

Ryan Calo. Open robotics. *Md. L. Rev.*, 70:571, 2010.

Ryan Calo, Ivan Evtimov, Earlence Fernandes, Tadayoshi Kohno, and David O'Hair. Is tricking a robot hacking? 2018.

Atakan Cetinsoy. "hype or reality? stealing machine learning models via prediction apis". *The Official Blog of BigML.com*, Oct 2016. URL `https://blog.bigml.com/2016/09/30/hype-or-reality-stealing-machine-learning-models-via-prediction-apis/`.

Xinyun Chen, Chang Liu, Bo Li, Kimberly Lu, and Dawn Song. Targeted backdoor attacks on deep learning systems using data poisoning. *CoRR*, abs/1712.05526, 2017. URL `http://arxiv.org/abs/1712.05526`.

European Commission. Communication artificial intelligence for europe. Technical report, Apr 2018. URL `https://ec.europa.eu/digital-single-market/en/news/communication-artificial-intelligence-europe`.

Tiffany Curtiss. Computer fraud and abuse act enforcement: Cruel, unusual, and due for reform. *Wash. L. Rev.*, 91:1813, 2016.

Ivan Evtimov, Kevin Eykholt, Earlence Fernandes, Tadayoshi Kohno, Bo Li, Atul Prakash, Amir Rahmati, and Dawn Song. Robust physical-world attacks on deep learning models. *arXiv preprint arXiv:1707.08945*, 1, 2017.

Matt Fredrikson, Somesh Jha, and Thomas 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, pages 1322–1333, New York, NY, USA, 2015. ACM. ISBN 978-1-4503-3832-5. doi: 10.1145/2810103.2813677. URL `http://doi.acm.org/10.1145/2810103.2813677`.

Justin Gilmer, Ryan P Adams, Ian Goodfellow, David Andersen, and George E Dahl. Motivating the rules of the game for adversarial example research. *arXiv preprint arXiv:1807.06732*, 2018.

Ian Goodfellow. Defense against the dark arts: An overview of adversarial example security research and future research directions. *arXiv preprint arXiv:1806.04169*, 2018.

Tianyu Gu, Brendan Dolan-Gavitt, and Siddharth Garg. Badnets: Identifying vulnerabilities in the machine learning model supply chain. *arXiv preprint arXiv:1708.06733*, 2017.

Matthew Jagielski, Alina Oprea, Battista Biggio, Chang Liu, Cristina Nita-Rotaru, and Bo Li. Manipulating machine learning: Poisoning attacks and countermeasures for regression learning. *arXiv preprint arXiv:1804.00308*, 2018.

Aleksander Mądry, Anish Athalye, Dimitris Tsipras, and Logan Engstrom. Robustml, April 2018. `https://www.robust-ml.org/defenses/`.

Maria-Irina Nicolae, Mathieu Sinn, Minh Ngoc Tran, Ambrish Rawat, Martin Wistuba, Valentina Zantedeschi, Nathalie Baracaldo, Bryant Chen, Heiko Ludwig, Ian Molloy, and Ben Edwards. Adversarial robustness toolbox v0.3.0. *CoRR*, 1807.01069, 2018. URL `https://arxiv.org/pdf/1807.01069`.

Nicolas Papernot, Patrick McDaniel, Xi Wu, Somesh Jha, and Ananthram Swami. Distillation as a defense to adversarial perturbations against deep neural networks. *arXiv preprint arXiv:1511.04508*, 2015.

Nicolas Papernot, Fartash Faghri, Nicholas Carlini, Ian Goodfellow, Reuben Feinman, Alexey Kurakin, Cihang Xie, Yash Sharma, Tom Brown, Aurko Roy, Alexander Matyasko, Vahid Behzadan, Karen Hambardzumyan, Zhishuai Zhang, Yi-Lin Juang, Zhi Li, Ryan Sheatsley, Abhibhav Garg, Jonathan Uesato, Willi Gierke, Yinpeng Dong, David Berthelot, Paul Hendricks, Jonas Rauber, and Rujun Long. Technical report on the cleverhans v2.1.0 adversarial examples library. *arXiv preprint arXiv:1610.00768*, 2018.

Mahmood Sharif, Sruti Bhagavatula, Lujo Bauer, and Michael K. Reiter. Accessorize to a crime: Real and stealthy attacks on state-of-the-art face recognition. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, CCS '16, pages 1528–1540, New York, NY, USA, 2016. ACM. ISBN 978-1-4503-4139-4. doi: 10.1145/2976749.2978392. URL `http://doi.acm.org/10.1145/2976749.2978392`.

Adam Shostack. *Threat Modeling: Designing for Security*. Wiley Publishing, 1st edition, 2014. ISBN 1118809998, 9781118809990.

Florian Tramèr, Fan Zhang, Ari Juels, Michael K. Reiter, and Thomas Ristenpart. Stealing machine learning models via prediction apis. *CoRR*, abs/1609.02943, 2016. URL `http://arxiv.org/abs/1609.02943`.

Qixue Xiao, Kang Li, Deyue Zhang, and Weilin Xu. Security risks in deep learning implementations. *arXiv preprint arXiv:1711.11008*, 2017.