

Foundations of Adversarial Machine Learning

Daniel Lowd¹ Christopher Meek² Pedro Domingos¹

¹ University of Washington ² Microsoft Research

Seattle, WA 98195-2350 Redmond, WA 98052

{lowd,pedrod}@cs.washington.edu meek@microsoft.com

October 19, 2007

As classifiers are deployed to detect malicious behavior ranging from spam to terrorism, adversaries modify their behaviors to avoid detection (e.g., [4, 3, 6]). This makes the very behavior the classifier is trying to detect a function of the classifier itself. Learners that account for concept drift (e.g., [5]) are not sufficient since they do not allow the change in concept to depend on the classifier. As a result, humans must adapt the classifier with each new attack. Ideally, we would like to see classifiers that are resistant to attack and that respond to successful attacks automatically.

In this abstract, we argue that the development of such classifiers requires new frameworks combining machine learning and game theory, taking into account the utilities and costs of both the classification system and its adversary. We have recently developed such a framework that allows us to identify weaknesses in classification systems, predict how an adversary could exploit them, and even deploy preemptive defenses against these exploits. Although theoretically motivated, these methods achieve excellent empirical results in realistic email spam filtering domains.

In general, we assume that the goal of the adversary is to evade detection while minimizing cost. Consider the task of an email spammer. The goal is to get an email message past a spam filter, and the cost comes from modifying the message by adding or removing words. These changes may make the spam more likely to pass through the filter, but they may also make for a less effective sales pitch. In credit card fraud, less desired fraudulent purchases may be less likely to be flagged as suspicious. Terrorists may attempt to disguise their activities to avoid detection, but it makes their operations more expensive. Even search engine optimization can be seen as an attempt to gain a higher ranking with minimal web page modifications and infrastructure investment. The advantage of a general framework is that it can be applied to a wide variety of important real-world problems.

In Dalvi et al. [2], we investigate automatically adjusting a classifier by predicting the adversary's behavior in advance. We model utility and cost functions for both the classifier and the adversary and compute optimal strategies for a sequential game. First, the classifier learns a cost-sensitive classification function on presumably untainted data. The adversary, who is assumed to have full knowledge of this function, modifies malicious examples to make them appear innocent while minimizing its own cost. Finally, the classifier, assumed to have full knowledge of the adversary's utility, adjusts its classification strategy by testing to see if innocent-looking instances could actually be optimally modified versions of malicious instances. This sequence can be repeated any number of times as the adversary and classifier iteratively respond to each other.

We evaluated our methods by taking publicly available email databases and running our adversary and classification algorithms with different utility settings and cost models. Against every attack, the adversary-aware classifier vastly outperformed an adversary-ignorant baseline. It remains to be seen if these methods can be effective in the real world, where information is much more limited and the space of possible moves by the adversary is not known in advance.

In Lowd and Meek [7], we look at a similar scenario but from the perspective of an adversary with limited information. As in Dalvi et al. [2], we assume that the adversary wishes to have instances (e.g., emails) misclassified with minimal cost (e.g., the number of added or removed words). However, instead of assuming complete knowledge, the adversary is allowed a polynomial number of membership queries to test what labels the classifier would assign to manufactured instances. For the case of spam, this can be done by seeing if test messages reach an email account protected by the spam filter. From these queries, the adversary must find an innocently-labeled instance whose cost

is within a factor k of the optimal cost. We refer to this task as an adversarial classifier reverse engineering (ACRE) learning problem. ACRE learnability is thus defined for pairs of concept classes and adversarial utility functions. The ACRE learning problem differs significantly from both the probably approximately correct (PAC) model of learning [9] and active learning [1] in that (1) the goal is not to learn the entire decision surface, (2) there is no assumed distribution governing the instances and (3) success is measured relative to a cost model for the adversary.

In principle, this approach could be used to investigate the hardness of any attackers attacking any system. We analyze linear classifiers with Boolean features, some of the most common classifiers used in practice. When the adversary has a simple, linear cost model, we show that the problem is ACRE learnable within a factor of 2.

We then evaluated the feasibility of attacking two linear spam filters trained on a corpus of 500,000 Hotmail messages. Given 1000 features (i.e., words) to modify, our algorithm used fewer than 700,000 queries in the median case. This may seem like a lot, but could be feasible when automated and parallelized across a distributed network of “zombie” machines. The median instance found was only 11% more costly than the optimal instance.

If we relax the optimality requirement, we can focus on a less formal but more practical adversarial question: which features are the best ones to modify, or for spam, which words are best at making a spam email look legitimate? With a simple algorithm, we found a set of very effective words using only 4000 queries [8]. Using this word set, the median spam could be disguised with fewer than 60 added words. Going one step further, we generated word lists without querying the spam filter at all, using only publically available data and simple heuristics. These word lists were still effective at disguising the median spam email with fewer than 200 words.

It may not be possible to entirely prevent these attacks, but we can discourage them by increasing their cost. We can increase the cost of finding out information about the classifier by making it harder to execute the membership queries. We can also use features that are harder for an adversary to modify and depend less on those features that are easily modified. In this way, understanding these attacks allows us to better defend against them.

A number of interesting questions remain. While these methods can be illuminating, they would be more useful if we could remove more of the assumptions restricting them. In both, it is assumed that we know the adversary’s utility function – how can we estimate the utility function empirically? And can we adjust our systems to make attacking them more costly than rewarding? That last question is the most important one, and will likely involve both obfuscation (selecting classifiers that are hard to reverse engineer) and counter-attacks (recognizing adversarial behavior using a predictive model).

References

- [1] Dana Angluin. Queries and concept learning. *Machine Learning*, 2(4):319–342, 1988.
- [2] Nilesch Dalvi, Pedro Domingos, Mausam, Sumit Sanghai, and Deepak Verma. Adversarial classification. In *KDD-2004*, pages 99–108, Seattle, WA, 2004. ACM Press.
- [3] T. Fawcett and F. Provost. Adaptive fraud detection. *Data Mining and Knowledge Discovery*, 1(3):291–316, 1997.
- [4] Tom Fawcett. “in vivo” spam filtering: A challenge problem for KDD. *SIGKDD Explorations*, 5(2):140–148, 2003.
- [5] G. Hulten, L. Spencer, and P. Domingos. Mining time-changing data streams. In *KDD-2001*, pages 97–106, San Francisco, CA, 2001. ACM Press.
- [6] D. Jensen, M. Rattigan, and H. Blau. Information awareness: A prospective technical assessment. In *KDD-2003*, pages 378–387, Washington, DC, 2003. ACM Press.
- [7] Daniel Lowd and Christopher Meek. Adversarial learning. In *KDD-2005*, Chicago, IL, 2005. ACM Press.
- [8] Daniel Lowd and Christopher Meek. Good word attacks on statistical spam filters. In *CEAS-2005*, Palo Alto, CA, 2005.
- [9] L. G. Valiant. A theory of the learnable. *Communications of the ACM*, 27(11):1134–1142, 1984.