A Survey of Game Theoretic Approaches for Adversarial Machine Learning in Cybersecurity Tasks

Prithviraj Dasgupta, Joseph B. Collins

■ *Machine learning techniques are used extensively for automating various cybersecurity tasks. Most of these techniques use supervised learning algorithms that rely on training the algorithm to classify incoming data into categories, using data encountered in the relevant domain. A critical vulnerability of these algorithms is that they are susceptible to adversarial attacks by which a malicious entity called an adversary deliberately alters the training data to misguide the learning algorithm into making classification errors. Adversarial attacks could render the learning algorithm unsuitable for use and leave critical systems vulnerable to cybersecurity attacks. This article provides a detailed survey of the state-of-the-art techniques that are used to make a machine learning algorithm robust against adversarial attacks by using the computational framework of game theory. We also discuss open problems and challenges and possible directions for further research that would make deep machine learning–based systems more robust and reliable for cybersecurity tasks.*

Adversarial learning (Tygar 2011) is an instance of machine learning by which two entities called the learner and adversary attempt to learn a prediction mechanism for data related to a problem domain at hand, albeit with different objectives. The learner's objective in learning the prediction mechanism is to correctly predict or classify the data. In contrast, the adversary's objective is to imperceptibly coerce the learner into making incorrect predictions for the data in the future. A very popular instance of adversarial learning is e-mail spam filtering (Tygar 2011; Huang et al. 2011). Here, the learner is the spam filter with a prediction mechanism that classifies incoming e-mail into two categories, spam or nonspam. The adversary is the spammer that, in addition to generating spam e-mail, tries to add, remove, or alter certain words or characters in the e-mail text (Dalvi et al. 2004) so that it can disguise nonspam e-mail as spam and vice versa. If the spammer is successful, the spam filter ends up misclassifying the altered nonspam e-mails as spam (false positives) or the altered spam e-mails as nonspam (false negatives). Both misclassifications could be dangerous for the integrity of the e-mail filtering system—not only do they block legitimate e-mail and allow potentially malicious spam e-mail to pass through, but they also reduce confidence in the e-mail classification performed by the spam filter.

Adversarial learning poses a severe cybersecurity threat in several domains that use machine learning–based classifier systems, including automated e-mail spam filters and antivirus software, image classification algorithms in defense and medical applications, and text-based sentiment analysis algorithms used on social media data. To combat these challenges, researchers have proposed several techniques that aim to make the learner's classifier robust against adversarial attacks (Huang et al. 2011). Many of these techniques use game theory, a popular decision-making framework at the intersection of mathematics, economics, and computer science (Fudenberg and Tirole 1991; Myerson 1997; Shoham and Leyton-Brown 2009). Game theory is an attractive tool for adversarial learning as it provides a means by which to mathematically model the learner's and the adversary's behaviors in terms of defense and attack strategies and to determine suitable strategies for reducing the learner's loss from adversarial attacks. In this survey, we focus on such game theory–based techniques that have been used to make machine learning algorithms robust against adversarial attacks.

The remainder of the article is structured as follows. In the next section we provide background information on adversarial learning and game theory. Following that, we summarize the contributions of game theory–based adversarial learning approaches and solution techniques. Then, we discuss open issues and challenges for future research directions in game theory–based modeling of adversarial learning, and finally we conclude. In the rest of the article, in accordance with the machine learning literature, we assume that the output of the learner's prediction mechanism classifies the data into a finite set of classes, and each class is identified with an output label. For legibility, while following most of the existing literature in this area, we assume that the learner uses a classifier for its prediction mechanism. In general, the learner could use any other prediction mechanism, such as clustering, ranking, or regression.

Background: Adversarial Learning and Games

Adversarial learning deals with techniques used by a machine learning–based prediction mechanism such as a classifier to make itself robust against adversarial attacks. Huang et al. (2011) provide a comprehensive overview of adversarial learning techniques. Their work includes a taxonomy for adversarial learning while characterizing it along three dimensions — influence, specificity, and security violation — as shown in figure 1. Influence and security violation–based attacks are divided into distinct categories marked by solid and dashed dividing lines. Specificity attacks range over a continuous spectrum marked by a dotted double arrow.

Influence is the most relevant and widely researched dimension for adversarial learning because it characterizes the adversary based on its behavior, with the objective of developing appropriate learner strategies to counter the adversary's behavior. The influence dimension specifies two types of adversarial attacks, causative and exploratory (also known as probing), illustrated in figure 2. In causative attacks the adversary acquires data used to train the learner's classifier and modifies these data. The modified data, called adversarial data, are then used by the learner during further training of its classifier. This process causes the learner to learn an incorrect classifier that gives classification errors (false positives and false negatives) during testing or when the classifier is used. In exploratory attacks, the adversary observes the output of the learner's classifier for various data and tries to discover its vulnerabilities (for example, what data it misclassifies). It then creates adversarial data that exploits those vulnerabilities to increase the classifier's misclassification rate during test or application time. Two other dimensions of adversarial learning shown in figure 1 are security violation and specificity. See Huang et al. (2011) for approaches that counter adversarial attacks along these three dimensions. The rest of this article focuses specifically on how the influence dimension's causative and exploratory attacks are countered with game theory–based techniques.

The scenario in figure 2 shows the two types of influence-based adversarial attacks — causative (top right) and exploratory (bottom right). The learner uses a support vector machine to classify input. Red input represents adversarial examples that are created by the adversary using a perturbation function $\phi()$ from valid examples from the training set and reinjected into the training set.

Noncooperative Game

Adversarial learning has been extensively modeled as a two-player, noncooperative game. A noncooperative game can be informally defined as an interaction between two or more players over a resource that has to be shared between the players. The game, represented in normal form, is given by (N, A, U). Here N is the set of players, $A = \{A_i\}$, where A_i is the set of actions for player i, and $U = \{U_i\}$, where $U_{\mathrm{i}}(a_{\mathrm{i}}, a_{-i})$ gives a real-valued number called utility received by each player $i \in N$ when it selects action $a_i \in A_i$ while other players jointly select $a_{-i} \in A_{-i}$, $A_{-i} = X\ A_{j \neq i}$. The utility of each player gives its preference over the various outcomes of the game resulting from different joint actions by the players. Player i's strategy set specifies a probability distribution over its actions A_i. The outcome of a game is a strategy selected by each player. One of the most widely used techniques to calculate a player's strategy in a game is given by the Nash equilibrium. The Nash equilibrium assumes that players behave rationally and each player i plays its best response strategy given by s_i that satisfies $U_i(s_i, s_{-i}) \geq U_i(s'_i, \mathrm{s}_{-i})$ for all $i \in N$. The Nash equilibrium of a two-player game can be represented as either a linear complementarity problem and solved using linear programming or as a search problem (Shoham and Leyton-Brown 2009). In the following, we briefly mention a few aspects of games that are relevant to adversarial learning.

Zero-Sum Versus Non-Zero Sum Game

In a two-player, zero-sum game, the utilities of the players, the learner and the adversary, sum to zero. In other words, the gain in utility of the learner comes at the cost of the loss of the adversary's utilities and vice versa. In a two-player, zero-sum game, the Nash equilibrium can be calculated using the minimax theorem, which says that the game's Nash equilibrium outcome is the same as its minimax outcome. The minimax outcome can be represented as a constrained optimization problem and solved as a linear program. Adversarial learning has been extensively modeled as a two-player, zero-sum game. However, Bruckner and Scheffer (2011) recently observed that assuming adversarial learning is a zero-sum game is overly pessimistic — the utility loss of the learner might not equal the utility gain of the adversary. Consequently, they model adversarial learning as a non-zero-sum or general-sum game. Unfortunately, the minimax theorem's result about the correspondence between the Nash equilibrium and minimax outcome does not hold for general-sum games, and, more complicated, general Nash equilibrium solution techniques (Shoham and Leyton-Brown 2009) have to be used to determine the learner's and the adversary's selected strategies.

Simultaneous Move Versus Sequential Game

In a simultaneous move game, players make their strategy selection simultaneously and cannot observe each other's strategy before selecting their own. In contrast, in a sequential move game, players take turns selecting their strategies (or making their moves).

Adversarial learning has been modeled as the latter — the learner is the player making the first move or the leader, because it usually publishes its classifier and is not aware of the presence of the adversary (Huang et al. 2011; Grosshans et al. 2013). The adversary, on the other hand, is the follower because it can observe the learner's classifier and then make its move of selecting a suitable strategy to generate adversarial instances. Sequential move games are easier to solve than simultaneous move games because the follower knows the leader's selected strategy and can use this information to select its utility-maximizing strategy. The leader's strategy selection technique, however, does not have information about the follower's selected strategy. Consequently, when selecting its strategy, the leader has to incorporate this uncertainty about the follower's strategy using a type of game called a Bayesian game, as described next.

Bayesian Game

The normal game form assumes that each player has information about the utilities of other players for each action. This assumption might not be valid in many practical, real-life scenarios because it is unrealistic for a player to have accurate information about competing players' utilities. For example, in adversarial learning, the learner might not have accurate information about the adversary's cost to generate adversarial data (Bruckner and Scheffer 2011; Grosshans et al. 2013) or the adversary might not have accurate information about the learner's classification cost (Lowd and Meek 2005). The problem is addressed through a Bayesian game (Harsanyi 1968), in which each player is assumed to have a set of types. The utility that a player gets from each of its actions now also depends on its types. A player does not know the exact type of the other players, but it does know the probability distribution over the types. Based on this information, a player can calculate expected utilities, conditioned on the other players' types. Some researchers have modeled adversarial learning as a Bayesian sequential move game, in which the learner assumes a set of types for the adversary along with probability distribution over the types. It then selects a strategy based on its expected utilities conditioned on the prior probabilities of the various adversary types, as discussed later in the section on non-zero-sum games.

The topic of security games is closely related to adversarial learning, although the roles and objectives of the learner and the adversary in a security game are slightly different from those in adversarial learning. In security games (Paruchuri et al. 2008), the learner is called the defender, whose objective is to protect a set of targets from an adversary, referred to as an attacker. The problem facing the defender is to allocate protection resources within budget and operational

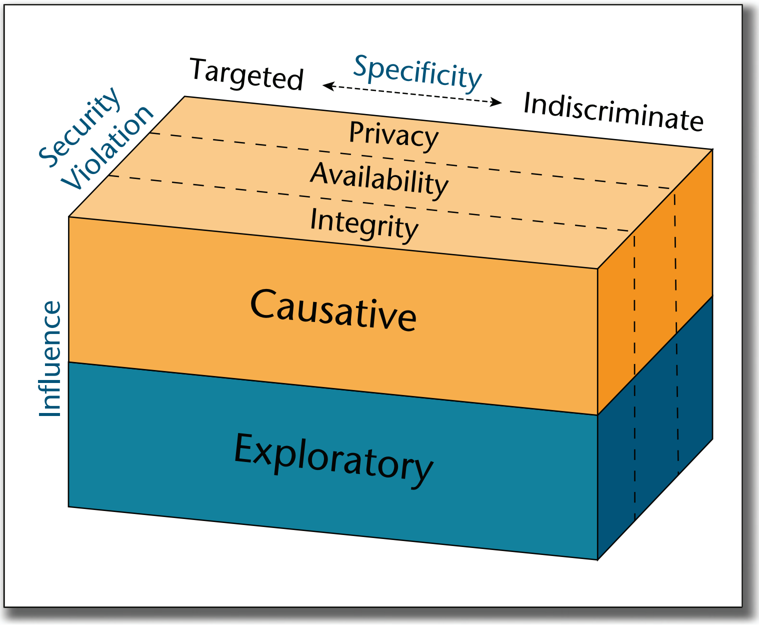

Figure 1. Taxonomy of Types of Adversarial Attacks along Three Dimensions.

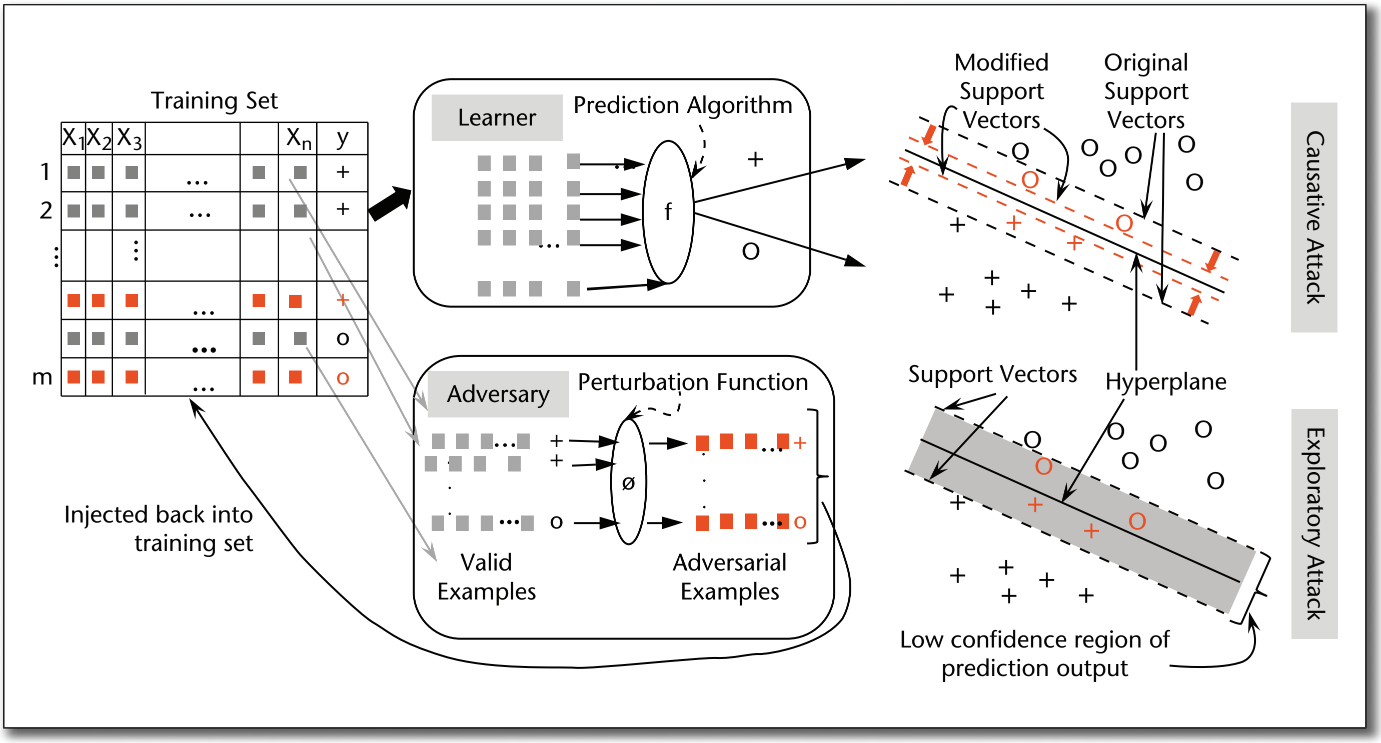

Figure 2. Adversarial Learning Scenario.

constraints, to achieve a desired level of security. Security games have been applied to real-life applications including airport security (Pita et al. 2011), wildlife protection (Fang et al. 2017), and natural resource conservation (Ford et al. 2016). Tambe (2011) provides an excellent discussion on security games.

Game Theory–Based Adversarial Learning Techniques

Adversarial learning is usually modeled as a two-player game between the learner and the adversary. The learner's set of actions corresponds to selecting different hyperparameters for its classifier, while the adversary's set of actions corresponds to different strategies for changing valid data into adversarial data. For example, an adversary's action could be adding different amounts of perturbation or noise to valid data or removing certain features from valid data. The utilities for the learner and the adversary are defined in terms of their joint actions.

In an early, seminal work on adversarial classification, Dalvi et al. (2004) formulated adversarial classification as a two-player competitive game between the learner and the adversary called a classification game. The learner's prediction mechanism is a binary classifier, while the adversary creates adversarial input by perturbing features from legitimate input. The game is asymmetric as the adversary is aware of the learner's classifier parameters, utilities, and costs, but the learner is not aware whether an input is adversarial versus legitimate. Within this setting, the learner's utility is defined as its value from classifying input (misclassification yields negative value) minus a per-feature cost for including input features in the classification algorithm.

Similarly, the adversary's utility is defined as its value from misclassification of an adversarial input by the learner (correct classification by the learner yields negative value to the adversary) minus its cost to generate the adversarial input from legitimate input. Both learner and adversary play a Nash equilibrium strategy.

The problem is formulated as a constrained optimization problem and solved as a mixed integer linear program for the adversary that is then used by the learner to determine a robust classification strategy. The proposed strategy is validated using spam e-mail data sets with different adversarial perturbation strategies and learner misclassification penalties and has been shown to yield positive classifier utilities, implying low misclassification rates of adversarial input by the learner.

Following Dalvi et al. (2004), researchers have proposed various approaches to defining these utilities depending on their learner and adversary behavior models. We categorize the solution techniques proposed in the literature into two categories — those that use zero-sum games to model the interaction between the learner and adversary followed by a minimax-based linear optimization solution to solve it and those that model the interaction as a non-zero-sum game and use a Nash equilibrium–based, bilevel optimization or a related solution. Table 1 lists the major game theory–based approaches for adversarial learning along with different learner and adversary models and solution techniques.

Zero-Sum Games: Constrained Optimization-Based Solution Techniques

Lowd and Meek (2005) extended the model of Dalvi et al. (2004) while relaxing the assumption that the adversary has full information about the learner's classification algorithm, utilities, and costs. Although their main algorithm, called adversarial classifier reverse engineering (ACRE), is not based on a game, we review it here as it has formed the basis for game theory–based techniques for adversarial learning. In the adversarial learning problem considered by Lowd and Meek (2005), the adversary can discover information about the learner's classifier by sending a limited number of queries containing adversarial input. With this limited knowledge of the learner's classifier, the adversary's objective is to determine the input instance that incurs the lowest cost to modify into an adversarial input. The set of such input instances is called attack vectors. Using the ACRE algorithm, the adversary can find attack vectors that can defeat the learner's classification algorithm when the input consists of either continuous or binary features. The algorithm is validated on spam e-mail data sets when the learner uses either a naive Bayes or a maximum entropy–based classifier and shows that an adversary using ACRE can find instances within 17% of the actual lowest cost instance while using a few thousand queries. Nelson et al. (2010) generalized ACRE to an ε-instance minimal adversarial cost problem, where the learner's set of classifiers is expanded from the linear classifier to more general, convex-inducing classifiers.

The solution of Nelson et al. searches over the adversarial cost space to determine the minimum set of adversarial examples, called the evasion set, that needs to be generated by the adversary to effect classification errors by the learner. Avoiding reverse engineering of the classifier allows their approach, unlike that of Lowd and Meek (2005), to handle classifiers that are difficult to reverse engineer. Recently, Li and Vorobeychik (2014; Vorobeychik and Li 2014) also extended the ACRE framework. Their proposed techniques include defining the cost between instances using equivalence class–based cost functions and solving the optimization problem facing the adversary as a mixed integer linear program, as well as using a concept called moving target defense (Jajodia et al. 2012), in which the learner employs randomization over multiple classifiers instead of tuning parameters of a single classifier to make its prediction robust against adversarial attacks.

A parallel, related line of research considers the adversary's behavior from a slightly different approach, the adversary generating adversarial data by selecting, removing, or corrupting features from the input data set. Globerson and Roweis (2006) describe such a setting, where the adversary can remove multiple or single features from input and the learner's objective is to determine an optimal set of feature weights for its classifier that minimize a metric called the hinge loss. The problem is formulated as a minimax zero-sum game and is represented as a constrained optimization problem. The proposed algorithm was verified with adversarial data and shown to give lower error rates than a support vector machine classifier for handwritten digit classification and spam filtering. Their model of selective feature removal by the adversary was extended by Dekel, Shamir, and Xiao (2010) with two variants of the learner. When the learner is able to train a classifier using training data, the problem is solved as a linear program. On the other hand, when the learner does not have access to training data, the learning task becomes an online learning problem. In this case, the learner's classifier is determined using a neural network perceptron algorithm (Rosenblatt 1958) followed by a batching technique to model the online classifier as a statistical learning algorithm while making statistical guarantees about the classifier's performance. Experiments comparing the algorithm of Dekel, Shamir, and Xiao with those of Globerson and Roweis (2006) and Teo et al. (2007), using the same data sets, showed that their proposed technique improves classification accuracy over the compared techniques.

Hardt et al. (2016) considered adversarial learning games from the perspective of information revelation by the learner and the adversary. They considered two variants of the learner: when the learner has information about the adversary's cost function, ground truth, and input distribution, and when the learner knows only the adversary's cost function but not its ground truth or input distribution. Similarly, two types of adversary are considered: one that knows all parameters that the learner knows plus its own adversarial example generation function, and one that knows only its own cost function and adversarial example generation function. Within these settings, the learner's objective is to determine a strategy robust algorithm — one that selects a classifier that maximizes the probability of the classifier's output corresponding to the ground truth for possibly adversarial examples. On the other hand, the adversary tries to create adversarial examples that maximize its utility given by the difference between its benefit from the learner's classifier output for that example and its cost to generate the example. Their results theoretically analyze the running time and sample complexity of the learner for different types of adversary functions, called separable and nonseparable.

As mentioned, most of the existing literature on adversarial learning games assumes a sequential move game with the learner as the leader. However, few researchers have analyzed adversarial learning where the learner knows the adversary's strategy to generate adversarial data, but the adversary does not have information about the learner's classifier; this makes the adversary the leader and the learner the follower (Kantarcioglu, Xi, and Clifton 2011; Liu and Chawla 2009). Liu and Chawla (2009) considered such a setting in which the adversary tries to generate adversarial data that results in moving the learner's

Reference	Initial Information about Learner with Adversary	Adversary Attack Model	Validation Domain
	Zero Sum Games		
Globerson 2006; Dekel 2010; Teo, 2007,	No information about learner's utility, costs, and classifier parameters	Exploratory attacks by removing features from future input	Spam filtering
Hardt 2016	With and without information about probability distribution of input and ground truth	Exploratory attacks by changing values of future input	Spam filtering
	Sequential, Bayesian, Nonzero Sum Games		
Dalvi 2004	Full information about learner's utility, cost, and classifier parameters	Causative attacks	Spam filtering
Bruckner 2011; Groshans 2013; Bruckner 2012; Groshans 2015	No information about learner's utility, costs, and classifier parameters	Exploratory attacks by changing values of future input	Spam filtering
Alteld 2017	No information about learner's utility, costs, and classifier parameters	Exploratory attacks on test set only	Stock prices
Zhou 2012; Zhou 2014; Dritsoula 2017	No information about learner's utility, costs, and classifier parameters	Exploratory attacks by mixing valid and adversarial input, for example, altering all or part of input features	Spam filtering

Table 1. Comparison of Game Theory–Based Adversarial Learning Techniques.

Only first authors' names are given.

classification boundary between spam versus nonspam e-mail. The learner uses a genetic algorithm to search classifier parameters that reduce its classification error in response to these adversarial attacks. This work was extended to calculate the Nash equilibrium of a constant-sum game efficiently with reduced computation (Liu et al. 2012) and, recently, with the learner using a deep convolutional neural network as its classifier (Chivukula and Liu 2017). Nevertheless, several authors (Huang et al. 2011; Bruckner and Scheffer 2011) justify the learner as the leader while observing that most real-life classifier-based systems like e-mail spam filters and network traffic filters publish their classifier algorithms publicly.

In contrast to sequential move games, simultaneous move games have been used less often to model interaction between an adversary and the learner.

Recently, Schuurmans and Zinkevich (2016) addressed the problem of training a deep neural network as solving for a Nash equilibrium in a repeated, zero-sum game between two players, called the protagonist and the antagonist. The antagonist's objective is to determine a set of parameters that reduce the loss function during training, while the protagonist tries to select weight values of the edges in the deep network such that its utility is the negative of the antagonist's utility. Additional players called zannis are placed at the input and hidden layer nodes to select those nodes' parameter values.

Iterations used to adjust weights of the neural network's edges in a conventional supervised training algorithm are modeled as repeated plays of the game aimed at converging to the Nash equilibrium and are implemented using two algorithms, exponentiated weight and regret matching. Although not directly related to an adversarial setting where the adversary generates adversarial data to misguide the learner's classification, the deep learning game provides an interesting direction that can be extended to an adversarial setting.

Non-Zero-Sum Games: Bayes–Nash Equilibrium and Related Solution Techniques

Non-zero-sum games have been proposed as a more realistic model for the interactions between the learner and the adversary in adversarial learning (Bruckner and Scheffer 2011; Grosshans et al. 2013; Grosshans and Scheffer 2015; Mei and Zhu 2015; Alfeld, Zhu, and Barford 2017) because the loss in utility of the learner might not exactly equal the gain in utility of the adversary and vice versa. In a non-zero-sum adversarial learning game, the learner calculates its loss in utility as proportional to the number of data instances on which it made a classification error because the data were modified by the adversary. However, a wrinkle in this approach is that the learner does not know whether the adversary had indeed modified the data to make it commit a classification mistake. For example, consider a learner in an automated e-mail spam filter that has a spam identification rule as follows: "if there are more than three misspelled words in an e-mail text, then the e-mail is spam." Suppose that this learner receives an e-mail that has five misspelled words and classifies it as spam according to its spam identification rule. However, the learner does not know whether the misspelled words were generated by an adversary or whether they were genuine typographical errors made by a human. To address this problem, the learner tries to estimate probabilistically whether an instance it classifies was generated by an adversary and then weighs its classification error by this probability. Correspondingly, the adversary also calculates its loss in utility depending on whether its adversarially generated instance was able to fool the classifier. Using this framework, Bruckner and Scheffer (2011) proposed a non-zero-sum game to model adversarial learning. The pairs of utilities of the learner and the adversary form a probabilistic version of the normal form game called a Bayesian game. The strategies adopted by the players in this Bayesian game — for example, the classifier hyperparameters selected by the learner and the perturbation strategy to modify legitimate data selected by the adversary — are calculated using the Bayes–Nash equilibrium.

Like the ACRE algorithm (Lowd and Meek 2005), the model by Bruckner and Scheffer (2011) has been extended from different aspects in future research. Mei and Zhu (2015) considered attacks on the training set in adversarial learning within a problem called machine teaching. Here, the adversary takes the role of a teacher, while the learner takes the role of a student whose objective is to learn a concept from data provided by the teacher. The objective of the teacher is to make causative attacks so that it can coerce the learner toward learning a concept that it desires. Like that of Bruckner and Scheffer (2011), the problem is modeled as a bilevel optimization problem and solved by relaxing it to linear optimization using Karush–Kuhn–Tucker conditions. Subsequently, Alfeld, Zhu, and Barford (2017) extended this framework to test time attacks on data. Bruckner's model (Bruckner and Scheffer 2011) has also been generalized by Bulò et al. (2016) using randomized prediction games where the learner's prediction algorithm randomizes over classifiers with different weight parameters while the adversary randomizes over its adversarial vectors.

Another direction of adversarial learning investigated by Zhou et al. (2012) is along the lines of Globerson's model of selective feature removal (Globerson and Roweis 2006). Here, the authors consider two adversarial attack models called full range attacks and restrained attacks. In a full-range attack, the adversary can perturb any fraction of the maximum range of a feature. On the other hand, in a restrained-range attack, the adversary can perturb only a fraction of the difference between its intended value and the actual value of a feature. The learner's prediction mechanism to counter these attacks is an SVM-based classifier that minimizes hinge loss. Extending this work toward more robust learner, the authors proposed a mixture of the Bayesian expert's approach (Zhou and Kantarcioglu 2014) as the learner's prediction mechanism.

An adversarial learning game called a classification game in the paper by Dritsoula, Loiseau, and Musacchio (2017) considers a practical adversarial strategy used by adversaries like spammers. Spammers might sometimes behave nonmaliciously and not generate adversarial data to misguide the learner. This could result in false alarms by the learner if it incorrectly identifies the adversary as malicious when it is not. To account for this, the learner maintains a probability of the adversary being malicious versus nonmalicious. The game is nonzero sum as the learner's utility includes the negative of the adversary's utility when it is malicious, plus the learner's expected penalty from false alarms. Their work analyzes the existence and uniqueness of the Nash equilibrium for this classification game and proposes a constrained optimization solution, solved as a linear program, to calculate the Nash equilibrium. Their model is validated by generating numeric values of learner and adversary costs and size of strategy sets when data instances have either single or multiple features. Their results show that the learner utility decreases while attacker utility increases when either the cost of a single attack or the false alarm penalty increases.

Learner Robustness via Adversarial Data Modeling

The game theory–based adversarial learning techniques discussed thus far mainly focus on strategies that the learner could use to develop robustness against attacks from an adversary. Another approach to building the learner's defense mechanism, although not based on game theory, focuses on modeling the malicious data generated by the adversary so that the learner can understand the nature of adversarial attacks. Armed with information about the characteristics of adversarial attacks, the learner can then build appropriate defenses, such as train its classifier with the adversarial data, to improve robustness against the adversarial attacks. In this section, we provide an overview of three popular techniques for adversarial data generation that could be used in conjunction with adversarial learning: adversarial data generation using perturbation techniques on valid examples, transferring adversarial examples across different learner models, and generative adversarial networks (GANs).

Adversarial Data Generation via Perturbation

Before building the learner's defense mechanism against adversarial attacks, a first line of defense toward protecting against attacks is to understand how the adversary crafts those attacks. The topic of adversarial data generation seeks to address this issue by developing and analyzing different techniques that create synthetic, adversarial data, which could be used by potential adversaries. In most of these techniques, an adversarial example is constructed by adding a certain amount of noise or perturbation to a valid example. The main objective when creating an adversarial example is to perturb a valid example so that the perturbation is imperceptible to a human; in other words, the perturbed example appears to have the same label as a valid example to a human. However, when presented to a machine classifier, the same perturbed example would be assigned a different label than the valid example's label. For example, in a spam filtering scenario, when a perturbed example is created from a valid spam e-mail message, the perturbed example would still appear to be a spam message to a human, but a spam filtering classifier would categorize it as nonspam and vice versa. To achieve this property of imperceptibility to humans but deception for machine classifiers, the perturbation added to a valid example should take the perturbed example just across a decision boundary of the machine's classifier. Too little perturbation prevents the perturbed example from crossing the decision boundary — the perturbed example appears valid to the human, but does not fool the classifier either. On the other hand, too much perturbation takes the perturbed example far across the decision boundary, the classifier does not classify it correctly, but the excessively perturbed example appears as nonsense or rubbish (Goodfellow, Shlens, and Szegedy 2014) to a human who can easily discern it as an adversarially perturbed example. The main problem in adversarial data generation is then to determine this suitable amount of perturbation.

In one of the earliest works in this direction, Biggio et al. (2013) proposed a gradient descent technique that used the gradient of the discriminant function of the classifier along with the density function of the data to calculate a suitable amount of perturbation and generate a perturbed example. The proposed technique was validated to generate adversarial data from valid examples of handwritten digits and PDF text files while using various machine learning classifiers, including linear classifiers, support vector machines, and neural network classifiers. Optimization-based algorithms for determining the minimum amount of perturbation have been proposed by Szegedy et al. (2013) and Carlini and Wagner (2016). Goodfellow, Shlens, and Szegedy (2014) made several fundamental contributions toward understanding properties of perturbations that create adversarial examples as well as properties of learner models that make them susceptible to adversarial examples. They proposed a fast, simple, yet effective perturbation technique called the *fast gradient sign method* to add perturbation proportional to the gradient of the cost function (for example, loss function) to classify an example in a deep neural network. Their work also made valuable observations about perturbation techniques such as that the direction of perturbation rather than amount of perturbation is more critical in creating adversarial examples, training a classifier with adversarial examples is akin to regularization of the classifier, and a positive correlation exists between the degree to which a learning model can be optimized and its susceptibility to perturbation. Building on these directions, researchers have proposed more refined perturbation techniques such as perturbing the label that has the lowest probability for the valid example in a single step or multiple steps (Kurakin, Goodfellow, and Bengio 2016a, 2016b), perturbing an example's features that are most likely to change the classifier's output based on forward gradients (Papernot et al. 2016), universal perturbations to determine the minimal perturbation that will generate a certain fraction of adversarial examples guaranteed to result in misclassification when the examples are drawn from a given data distribution (Moosavi-Dezfooli et al. 2016), and neural networks called adversarial transformation networks that are trained to create adversarial examples (Baluja and Fischer 2018). Most of these techniques have been proposed for generating adversarial data of handwritten digits or images. In contrast, Sethi and Kantardzic (2018) described methods to generate adversarial text data. Adversarial examples called attack data are constructed from probing the classifier with randomly perturbed text and retaining the perturbations that are successful in fooling the classifier, when the number of probes that the adversary can make is limited. Without limits on the number of probes, the adversary can reverse engineer (Lowd and Meek 2005) the classifier to create more precise adversarial examples that are able to fool the classifier more often (Sethi and Kantardzic 2018). Carlini and Wagner (2018) have recently proposed methods for generating adversarial audio data for misleading speech-to-text machine classifiers. In general, the topic of adversarial data generation techniques for gaining insight into how the adversary can deceive the learner's classifier with malicious data is still an open research problem that requires meticulous analysis of data perturbation techniques in conjunction with the characteristics of the model used by the learner for classification.

Transferring Adversarial Examples

The adversarial data generation techniques discussed in the previous section require the virtual adversary to use the model used by the learner to classify examples so that the virtual adversary can determine whether the adversarial examples it generates are able to deceive the learner's classifier. Because the virtual adversary cannot gain access to the learner's model, the adversary usually resorts to reconstructing the learner's model via probing — sending valid and adversarial examples to the learner's model of the classifier and observing the output label assigned by the classifier. Each probe incurs a cost for the adversary because the adversary has to expend resources to acquire valid examples and perturb them, plus the learner could limit the number of examples the adversary could send. To reduce costs, it would be beneficial for the adversary if it could generate adversarial examples to fool a classifier while utilizing a certain learner model, then reuse those same adversarial examples to fool multiple different classifiers. This technique of sending adversarial examples generated using one learner model to a different model for classification is called transferring examples. The transfer problem is very relevant in the context of cybersecurity because it gives adversaries a low-cost technique with which to attack diverse machine learning–based classifiers such as e-mail spam filters, network intrusion detection systems, and identity authentication systems, presumably at different locations, while generating only one set of adversarial data.

As in the case of adversarial data generation, research in transferring adversarial examples has focused mainly on what characteristics of learner models of classifiers favor transferring adversarial examples across the models. Goodfellow, Shlens, and Szegedy (2014) identified that when the weight vectors of two neural network–based learning models are aligned with each other, adversarial examples generated using one model could be transferred to the other. The transferability of adversarially generated image data across different learning-based models of image classifiers was investigated by Liu et al. (2016) to discover that the alignment of the decision boundaries of different models favors transferability of adversarial examples across models. Recently, Tramèr et al. (2017) investigated the dimensionality of adversarial subspaces as a means to determine the transferability of adversarial examples. They concluded that the subspace of adversarial examples has a large dimension (about 25) and adversarial examples are transferable across two learner models when there is significant overlap in the subspace of the adversarial examples generated using the two models. Based on these findings the same authors proposed a technique called ensemble adversarial training (Tramèr et al. 2017), in which perturbations generated using one learner model are transferred or informed to another learner model to make the latter model more robust to adversarial examples. Like adversarial data generation, transferability of adversarial examples between different learner models is also an open problem whose investigation could lead to more robust adversarial learning techniques.

GANs

GANs (Goodfellow et al. 2014) have recently been proposed as a game theory–based technique for simultaneously generating perturbed examples from an adversary and then using those adversarial examples to train the learner's model. In a GAN, the learner uses a function called the discriminator (usually a classifier), while the adversary's function is called the generator. The interaction between the discriminator and the generator is modeled as a zero-sum game, and both the discriminator and the generator iteratively solve a minimax optimization function. The optimization is done iteratively over chunks or batches of data and implements a gradient descent over the respective loss functions of the discriminator and generator. Experimental results of this approach show that both discriminator and generator are able to continuously adapt their prediction and data corruption mechanisms, respectively. Rather than improving the robustness of the learner to adversarial examples, the main contribution of GANs has been to demonstrate that the adversary (generator) can create very convincing adversarial or counterfeit examples, for example, images of animals, human faces, or traffic signs, that are not distinguishable from a real image by the human eye but can cause the discriminator to output an incorrect classification. For example, a picture of a cat could be modified by the generator in a way that is imperceptible to the human eye but causes the discriminator to mislabel it as an airplane. More generally, the GAN techniques have shown for the first time that data labeling, which is largely a supervised learning task utilizing labeled training data from humans, can also be implemented as an unsupervised learning task by exploiting the adversarial attacks of the generator. GANs have been used in several applications, including image, audio, and video generation and computer vision tasks such as image and video labeling (Vondrick, Pirsiavash, and Torralba 2016; Reed et al. 2016). Improvements to the basic GAN, including the Wasserstein GAN (Arjovsky, Chintala, and Bottou 2017), SeqGAN (Yu et al. 2017), StackGAN (Zhang et al. 2016), and EnergyGAN (Zhou and Kantarcioglu 2016), have also been proposed. Defense mechanisms of the discriminator in a GAN include adversarial training, defensive distillation, and gradient masking (Papernot et al. 2017), as well as statistical data analysis methods (Grosse et al. 2017), although many of these techniques have been shown to be vulnerable more recently (Carlini and Wagner 2016; He et al. 2017). Similar to perturbation-based adversarial data generation, techniques developed for GANs to generate adversarial examples could also be used toward developing stronger adversarial learning mechanisms.

Open Problems and Further Directions

As we've discussed, game theory offers a convenient means of modeling learner and adversary behaviors in adversarial learning. However, with recent developments in game theory–based behavior modeling using repeated, evolutionary games and machine learning using deep networks, there are certain directions in which these models could be improved to make adversarial learning more robust against real-life adversarial threats. We next identify some of these potential directions for future research.

Richer Models of Learner and Adversary Behavior

A shortcoming of many existing game theory–based models of adversarial learning is that the learner uses only one-time interaction history with the adversary to build and update its model of the adversary's behavior of generating adversarial data instances. We envisage that a more complex model that considers the history of interactions between the learner and the adversary can enable the learner to make more accurate decisions of the adversary's behavior and adapt its classifier accordingly.

Repeated games with Bayesian learning provide a theoretical foundation for building such a history-based model of the adversary by the learner. Investigation in this direction, while developing fast, heuristics-based algorithms that can guarantee accurate prediction of the adversary's behavior within quantifiable bounds, would lead toward more accurate and robust models of adversarial learning. In the following, we identify some specific directions and open problems in adversarial learning.

Bounded Life-Force of Adversary and Learner

Most of the game theory–based adversarial learning models discussed assume that the adversary has unlimited resources (for example, access to valid data, Internet access) and the budget to craft adversarial examples. Taken together, these assets could be considered as a life force of the adversary. However, in real life, Internet adversaries such as spammers usually have limited life force within which they attempt to maximize the impact of the adversarial data they generate. Researchers have started exploring this direction by considering a bounded feature adversary (Park, Weimer, and Lee 2017) that is limited in the extent of change it can effect on features in valid data and on the number of queries it can make to the learner (Globerson and Roweis 2006; Hardt et al. 2016). An interesting and practical future direction that has been less investigated is the effect of diminishing life force on the strategy of the adversary. For example, an adversary that perceives very little remaining life force would adopt aggressive strategies to maximize its harm on the learner. Going to a level deeper, the learner could also build a model of the adversary's life force by analyzing the adversary's attacks and then strategize its defense mechanism to minimize harm. Repeated game frameworks that incorporate life force–based strategizing are a suitable direction in investigating this problem.

Diminishing Value of Shared Resources (Data Set)

Yet another limitation of existing adversarial learning models is that the value of a shared resource, for example, e-mail data, is considered to be immutable while being subject to adversarial attacks. In reality, because of nonzero error rates of the learner's classifier, a small but nonnegligible number of causative and exploratory attacks get past the classifier, giving rise to a compromised data set and reducing reliability of the classifier. It would make sense to investigate techniques that incorporate diminished value of data in training and testing, and the effect on the classifier confidence, in the defense mechanism used by the learner. Once again, the adversary could simultaneously attempt to model the value of the data and the classifier confidence and incorporate these metrics into the adversarial data generation strategy.

Tactical Defender and Attacker Strategies

As discussed earlier, game theory–based adversarial learning models use learner and adversary utility values to parameterize strategies. In real life, adversaries like spammers of website attackers use tactical strategies for their attacks. Example learner strategies could include guns or butter, growing soft, strict justice, or layered defense, while the attacker could strategize with low but slow attacks, surprise attack, David and Goliath–type attack, or suicide attack. Techniques from behavioral game theory provide a suitable framework for modeling such tactical strategies and bringing adversarial learning research closer to practical attacks.

Deeper Behavior Modeling by Adversary and Learner

Most existing techniques for adversarial learning are based on a sequential game in which the adversary has information about the classifier used by the learner, although it does not have information about the parameters of the classifier. An interesting and practical direction worthy of investigation is a game theory–based, adversarial learning model in which the adversary has only partial, possibly inaccurate information about the learner's classifier. This setting would be relevant in most real-life settings, as the classifier used by the learner is usually proprietary information that is confidential to the learner. Similarly, moving beyond the Nash equlibrium, solutions like the price of anarchy and regret minimization (Shoham and Leyton-Brown 2009) could provide faster means of calculating strategies by the learner and adversary. Related to this direction, game theory models could be made more informed by incorporating modeling and reasoning costs such as cost to solve for Nash equilibrium, cost to maintain game play history, and cost to build opponent models from the history. Similarly, expenses incurred by the adversary to get access to resources like legitimate e-mail data sets and to the learner's classifier could be modeled as a reward that is proportional to its success in compromising the learner.

Robust Classification with Sparse Data

Supervised learning algorithms, including support vector machines and regression learning, that are used to build the learner's classifier in adversarial learning rely heavily on large, information-rich training sets to predict correctly. In many instances, these algorithms suffer from low accuracy if the data used in training are sparse or do not contain all possible feature instances. To mitigate this problem, a technique called domain adaptation or transfer learning has been proposed in the literature.

However, to the best of our knowledge, transfer learning has not been investigated in the context of adversarial learning. Using transfer learning would be very relevant in adversarial learning with sparse training data. For example, the mapping determined by the classifier from input instances to class labels for helpful versus not-helpful movie reviews in a movie reviews data set (for example, the IMDB data set) could be reused, after suitable transformations, for classifying Internet clients as malicious versus nonmalicious from sparse server log data. The critical problem here is to find correspondences between data sets of the source and target domains, and then suitably adapt the mapping learned in the source domain to the target domain.

Conclusions

We have provided a systematic classification of adversarial learning techniques using game theoretical frameworks. While adversarial learning has been researched for more than a decade, recent advances in machine learning, especially with deep networks, could be used to enhance existing game theory techniques for deeper learner and adversary behavior modeling, as well as to compute more efficient and robust action selection strategies by the learner. We have identified several open problems and challenges for future research in these directions. With the recent phenomenal growth of machine learning–based intelligent systems, we believe that addressing these challenges will advance real-life, classifier-based learning systems like e-mail spam classifiers, social network sentiment analysis tools, and image and sensor data recognition systems on autonomous vehicles toward becoming more robust and reliable for seamless human use.

Acknowledgments

The authors would like to acknowledge support from the US Office of Naval Research Summer Faculty Research program for supporting the work of Prithviraj Dasgupta at the US Naval Research Laboratory in 2017.

References

Alfeld, S.; Zhu, X.; and Barford, P. 2017. Explicit Defense Actions Against Test-Set Attacks. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, 1274–80. Menlo Park, CA: AAAI Press.

Arjovsky, M.; Chintala, S.; and Bottou, L. 2017. Wasserstein Generative Adversarial Networks. *Proceedings of Machine Learning Research* 70: 214–23.

Baluja, S., and Fischer, I. 2018. Learning to Attack: Adversarial Transformation Networks. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence*, 2687–95. Menlo Park, CA: AAAI Press.

Biggio, B.; Corona, I.; Maiorca, D.; Nelson, B.; Šrndić, N.; Laskov, P.; Giacinto, G.; and Roli, F. 2013. Evasion Attacks Against Machine Learning at Test Time. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, 387–402. Berlin: Springer. doi.org/10.1007/978-3-642-40994-3_25

Bruckner, M., and Scheffer, T. 2011. Stackelberg Games for Adversarial Prediction Problems. In *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 547–55. New York: Association for Computing Machinery.

Bulò, S. R.; Biggio, B.; Pillai, I.; Pelillo, M.; and Roli, F. 2016. Randomized Prediction Games for Adversarial Machine Learning. *IEEE Transactions on Neural Networks and Learning Systems* 28: 2466–78.

Carlini, N., and Wagner, D. 2017. Towards Evaluating the Robustness of Neural Networks. In *Proceedings of the 2017 IEEE Symposium on Security and Privacy*, 39–57. New York: IEEE.

Carlini, N., and Wagner, D. 2018. Audio Adversarial Examples: Targeted Attacks on Speech-to-Text. In *Proceedings of the 2018 IEEE Security and Privacy Workshops*, 1–7. New York: IEEE.

Chivukula, S., and Liu, W. 2017. Adversarial Learning Games with Deep Learning Models. In *Proceedings of the International Joint Conference on* Neural Networks, 2758–2767. New York: IEEE. doi.org/10.1109/IJCNN.2017.7966196

Dalvi, N.; Domingos, P.; Mausam; Sanghai, S.; and Verma, D. 2004. Adversarial Classification. In *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 99–108. New York: Association for Computing Machinery.

Dekel, O.; Shamir, O.; and Xiao, L. 2010. Learning to Classify with Missing and Corrupted Features. *Machine Learning* 81(2): 149–178. doi.org/10.1007/s10994-009-5124-8

Dritsoula, L.; Loiseau, P.; and Musacchio, J. 2017. A Game-Theoretic Analysis of Adversarial Classification. *IEEE Transactions on Information Forensics and Security* 12: 3094–109. doi.org/10.1109/TIFS.2017.2718494

Fang, F.; Nguyen, T. H.; Pickles, R.; Lam, W. Y.; Clements, G. R.; An, B.; Singh, A.; Schwedock, B. C.; Tambe, M.; and A. Lemieux. 2017. PAWS — A Deployed Game-Theoretic Application to Combat Poaching. *AI Magazine* 38(1): 23–36. doi.org/10.1609/aimag.v38i1.2710

Ford, B. J.; Brown, M.; Yadav, A.; Singh, A.; Sinha, A.; Srivastava, B.; Kiekintveld, C.; and Tambe, M. 2016. Protecting the NECTAR of the Ganga River Through Game-Theoretic Factory Inspections. In *Advances in Practical Applications of Scalable Multi-Agent Systems. The PAAMS Collection*, 97–108. New York: Springer. doi.org/10.1007/978-3-319-39324-7_9

Fudenberg, D., and Tirole, J. 1991. *Game Theory*. Cambridge, MA: MIT Press.

Globerson, A., and Roweis, S. T. 2006. Nightmare at Test Time: Robust Learning by Feature Deletion. In *Proceedings of the Twenty-Third International Conference on Machine Learning*, 353–60. New York: Association for Computing Machinery.

Goodfellow, I. J.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, C.; and Bengio, Y. 2014. Generative Adversarial Nets. In *Proceedings of Advances in Neural Information Processing Systems* 27, 2672–80.

Goodfellow, I. J.; Shlens, J.; and Szegedy, C. 2014. Explaining and Harnessing Adversarial Examples. arXiv CoRR abstract: 1412.6572. Ithaca, NY: Cornell University Library.

Grosse, K.; Manoharan, P.; Papernot, N.; Backes, M.; and McDaniel, P. D. 2017. On the (Statistical) Detection of Adversarial Examples. arXiv CoRR abstract: 1702.06280. Ithaca, NY: Cornell University Library.

Grosshans, M.; Sawade, C.; Bruckner, M.; and Scheffer, T. 2013. Bayesian Games for Adversarial Regression Problems. *Proceedings of Machine Learning Research* 28(3): 55–63.

Grosshans, M., and Scheffer, T. 2015. Solving Prediction Games with Parallel Batch Gradient Descent. In *Proceedings of the European Conference on Machine Learning and Knowledge Discovery in Databases*, 152–67. New York: Springer. doi.org/10.1007/978-3-319-23528-8_10

Hardt, M.; Megiddo, N.; Papadimitriou, C.; and Wootters, M. 2016. Strategic Classification. In *Proceedings of the 2016 ACM Conference on Innovations in Theoretical Computer Science*, 111–22. New York: Association for Computing Machinery. doi.org/10.1145/2840728.2840730

Harsanyi, J. 1968. Games with Incomplete Information Played by Bayesian Players, Part III. The Basic Probability Distribution of the Game. *Management Science* 14(7): 486–502. doi.org/10.1287/mnsc.14.7.486

He, W.; Wei, J.; Chen, X.; Carlini, N.; and Song, D. 2017. Adversarial Example Defense: Ensembles of Weak Defenses Are Not Strong. arXiv CoRR abstract: 1706.04701. Ithaca, NY: Cornell University Library.

Huang, L.; Joseph, D.; Nelson, B.; Rubinstein, B. I.; and Tygar, J. D. 2011. Adversarial Machine Learning. In *Proceedings of the 4th ACM Workshop on Security and Artificial Intelligence*, 43–58. New York: Association for Computing Machinery. doi.org/10.1145/2046684.2046692

Jajodia, S.; Ghosh, A. K.; Subrahmanian, V. S.; Swarup, V.; Wang, C.; and Wang, X. S., editors. 2012. *Moving Target Defense II: Application of Game Theory and Adversarial Modeling*. New York: Springer.

Kantarcioğlu, M.; Xi, B.; and Clifton, C. 2011. Classifier Evaluation and Attribute Selection Against Active Adversaries. *Data Mining and Knowledge Discovery* 22(1-2): 291–335. doi.org/10.1007/s10618-010-0197-3

Kurakin, A.; Goodfellow, I. J.; and Bengio, S. 2016a. Adversarial Examples in the Physical World. arXiv CoRR abstract: 1607.02533. Ithaca, NY: Cornell University Library.

Kurakin, A.; Goodfellow, I. J.; and Bengio, S. 2016b. Adversarial Machine Learning at Scale. arXiv CoRR abstract: 1611.01236. Ithaca, NY: Cornell University Library.

Li, B., and Vorobeychik, Y. 2014. Feature Cross-Substitution in Adversarial Classification. In *Proceedings of Advances in Neural Information Processing Systems*, 2087–95. Cambridge, MA: MIT Press.

Liu, W., and Chawla, S. 2009. A Game Theoretical Model for Adversarial Learning. In *Proceedings of the IEEE International Conference on Data Mining Workshops*, 25–30. New York: IEEE.

Liu, W.; Chawla, S.; Bailey, J.; Leckie, C.; and Ramamohanarao, K. 2012. An Efficient Adversarial Learning Strategy for Constructing Robust Classification Boundaries. In *Australasian Joint Conference on Artificial Intelligence*, 649–60. New York: Springer. doi.org/10.1007/978-3-642-35101-3_55

Liu, Y.; Chen, X.; Liu, C.; and Song, D. 2016. Delving into Transferable Adversarial Examples and Black-Box Attacks. arXiv CoRR abstract: 1611.02770. Ithaca, NY: Cornell University Library.

Lowd, D., and Meek, C. 2005. Adversarial Learning. In *Proceedings of the Eleventh ACM SIGKDD International Conference on Knowledge Discovery in Data Mining*, 641–7. New York: Association for Computing Machinery.

Mei, S., and Zhu, X. 2015. Using Machine Teaching to Identify Optimal Training-Set Attacks on Machine Learners. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence*, 2871–77. Menlo Park, CA: AAAI Press.

Moosavi-Dezfooli, S.; Fawzi, A.; Fawzi, O.; and Frossard, P. 2016. Universal Adversarial Perturbations. arXiv CoRR abstract: 1610.08401. Ithaca, NY: Cornell University Library.

Myerson, R. B. 1997. *Game Theory: Analysis of Conflict*. Cambridge, MA: Harvard University Press.

Nelson, B.; Rubinstein, B. I. P.; Huang, L.; Joseph, D.; Lau, S.; Lee, S. J.; Rao, S.; Tran, A.; and Tygar, J. D. 2010. Near-Optimal Evasion of Convex-Inducing Classifiers. *Proceedings of Machine Learning Research* 9: 549–56.

Papernot, N.; McDaniel, P.; Goodfellow, I.; Jha, S.; Celik, Z. B.; and Swami, A. 2017. Practical Black-Box Attacks Against Machine Learning. In *Proceedings of the 2017 ACM on Asia Conference on Computer and Communications Security*, 506–19. New York: Association for Computing Machinery.

Papernot, N.; McDaniel, P.; Jha, S.; Fredrikson, M.; Celik, Z. B.; and Swami, A. 2016. The Limitations of Deep Learning in Adversarial Settings. In *Proceedings of the IEEE European Symposium on Security and Privacy*, 372–87. New York: IEEE. doi.org/10.1109/EuroSP.2016.36

Park, S.; Weimer, J.; and Lee, I. 2017. Resilient Linear Classification: An Approach to Deal with Attacks on Training Data. In *Proceedings of the 8th International Conference on Cyber-Physical Systems*, 155–64. New York: Association for Computing Machinery. doi.org/10.1145/3055004.3055006

Paruchuri, P.; Pearce, J. P.; Marecki, J.; Tambe, M.; Ordóñez, F.; and Kraus, S. 2008. Playing Games for Security: An Efficient Exact Algorithm for Solving Bayesian Stackelberg Games. In *7th International Joint Conference on Autonomous Agents and Multiagent Systems*, 895–902. Richland, SC: International Foundation for Autonomous Agents and Multiagent Systems.

Pita, J.; Tambe, M.; Kiekintveld, C.; Cullen, S.; and Steigerwald, E. 2011. Guards — Innovative Application of Game Theory for National Airport Security. In *Proceedings of the 22nd International Joint Conference on Artificial Intelligence*, 2710–15. Menlo Park, CA: AAAI Press.

Reed, S.; Akata, Z.; Yan, X.; Logeswaran, L.; Schiele, B.; and Lee, H. 2016. Generative Adversarial Text to Image Synthesis. arXiv preprint. arXiv:1605.05396. Ithaca, NY: Cornell University Press.

Rosenblatt, F. 1958. The Perceptron: A Probabilistic Model for Information Storage and Organization in the Brain. *Psychological Review* 65: 386–408.

Schuurmans, D., and Zinkevich, M. A. 2016. Deep Learning Games. In *Proceedings of Advances in Neural Information Processing Systems*, 1678–86. Cambridge, MA: MIT Press.

Sethi, T. S., and Kantardzic, M. 2018. Data Driven Exploratory Attacks on Black Box Classifiers in Adversarial Domains. *Neurocomputing* 289: 129–43. doi.org/10.1016/j.neucom.2018.02.007

Shoham, Y., and Leyton-Brown, K. 2009. *Multiagent Systems — Algorithmic, Game-Theoretic, and Logical Foundations*. New York: Cambridge University Press.

Szegedy, C.; Zaremba, W.; Sutskever, I.; Bruna, J.; Erhan, D.; Goodfellow, I.; and Fergus, R. 2013. Intriguing Properties of Neural Networks. arXiv preprint. arXiv:1312.6199. Ithaca, NY: Cornell University Library.

Tambe, M. 2011. *Security and Game Theory: Algorithms, Deployed Systems, Lessons Learned*. New York: Cambridge University Press. doi.org/10.1017/CBO9780511973031

Teo, C. H.; Globerson, A.; Roweis, S. T.; and Smola, A. J. 2007. Convex Learning with Invariances. In *Proceedings of Advances in Neural Information Processing Systems*, 1489–96. Cambridge, MA: MIT Press.

Tramèr, F.; Kurakin, A.; Papernot, N.; Boneh, D.; and McDaniel, P. D. 2017. Ensemble Adversarial Training: Attacks and Defenses. CoRR abs/1705.07204. Ithaca, NY: Cornell University Library.

Tramèr, F.; Papernot, N.; Goodfellow, I.; Boneh, D.; and McDaniel, P. 2017. The Space of Transferable Adversarial Examples. arXiv preprint. arXiv:1704:03453. Ithaca, NY: Cornell University Library.

Tygar, J. D. 2011. Adversarial Machine Learning. *IEEE Internet Computing* 15(5): 4–6. doi.org/10.1109/MIC.2011.112

Vondrick, C.; Pirsiavash, H.; and Torralba, A. 2016. Generating Videos with Scene Dynamics. In *Proceedings of Advances in Neural Information Processing Systems*, 613–21. Cambridge, MA: MIT Press.

Vorobeychik, Y., and Li B. 2014. Optimal Randomized Classification in Adversarial Settings. In *Proceedings of the International Conference on Autonomous Agents and Multi-Agent Systems*, 485–92. Richland, SC: International Foundation for Autonomous Agents and Multiagent Systems.

Yu, L.; Zhang, W.; Wang, J.; and Yu, Y. 2017. SeqGAN: Sequence Generative Adversarial Nets with Policy Gradient. In *Proceedings of the Thirty-First AAAI Conference on Artificial* Intelligence, 2852–58. Menlo Park, CA: AAAI Press.

Zhang, H.; Xu, T.; Li, H.; Zhang, S.; Huang, X.; Wang, X.; and Metaxas, D. N. 2016. StackGAN: Text to Photo-Realistic Image Synthesis with Stacked Generative Adversarial Networks. arXiv CoRR abstract: 1612.03242. Ithaca, NY: Cornell University Library.

Zhou, Y., and Kantarcioglu M. 2014. Adversarial Learning with Bayesian Hierarchical Mixtures of Experts. In *Proceedings of the 2014 SIAM International Conference on Data Mining*, 929–937. Philadelphia: Society for Industrial and Applied Mathematics. doi.org/10.1137/1.9781611973440.106

Zhou, Y., and Kantarcioglu, M. 2016. Modeling Adversarial Learning as Nested Stackelberg Games. In *Proceedings of the 20th Pacific-Asia Conference on Advances in Knowledge Discovery and Data Mining*, Part II, 350–62. doi.org/10.1007/978-3-319-31750-2_28

Zhou, Y.; Kantarcioglu, M.; Thuraisingham, B. M.; and Xi, B. 2012. Adversarial Support Vector Machine Learning. In *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 1059–67. New York: Association for Computing Machinery. doi.org/10.1145/2339530.2339697

Prithviraj Dasgupta is the Union Pacific Endowed Professor in the Computer Science Department at the University of Nebraska, Omaha, and the director of the CMANTIC Robotics Lab at the university. His research interests are multiagent and multirobot systems, distributed AI, machine learning, and game theory. He has published more than 150 papers in leading journals and conference proveedings and has led several large federal research grants on these topics. He received his PhD and MS in computer engineering from the University of California, Santa Barbara, and his undergraduate degree in computer science and engineering from Jadavpur University, India.

Joseph B. Collins is a Senior Research Physicist and the Section Head of the Distributed Systems Section at the US Naval Research Laboratory, Washington, DC. He has 29 years of broad experience at the Naval Research Laboratory, including applications of pattern recognition techniques to signals and transactional data and use of high-performance computing. He has published a variety of papers and technical reports. He is currently heading up a research project called Adversarial Online Learning, researching the application of game theory principles to machine learning in an adversarial environment.