Deep Models and Artificial Intelligence for Military Applications:
AAAI Technical Report FS-17-03

Position Paper: Towards a Repeated Bayesian Stackelberg Game Model for Robustness Against Adversarial Learning

Prithviraj Dasgupta,[1] **Joseph Collins**[2]
[1]Computer Science Department, University of Nebraska, Omaha, NE 68182
[2] Distributed Systems Section (Code 5583), Naval Research Laboratory, Washington D.C. 20375

Abstract

In this position paper, we propose a game theoretic formulation of the adversarial learning problem called a Repeated Bayesian Stackelberg Game (RBSG) that can be used by a prediction mechanism to make itself robust against adversarial examples.

Introduction

Adversarial learning (Huang et al. 2011) is an important problem in several machine learning-based prediction systems such as email spam filters, online recommender systems, text classifier and sentiment analyzing techniques on social media, and, automatic video and image classifers. Recently, a type deep network-based model of adversarial learning called Generative Adversarial Networks (GANs) (Goodfellow and et al. 2014) have gained popularity. In GANs, a learning algorithm, called a discriminator, is tasked with the problem of correctly predicting or labeling data passed to it into a finite set of classes. However, an adversary, called the generator, continuously tries to learn the discriminator's prediction mechanism and creates malicious or adversarial examples of data that are passed on to the discriminator. The objective of the generator is to create enough adversarial examples so that the discriminator's classification mechanism results in increased misclassification. The main contribution of GANs has been to show that the generator can create adversarial examples that are very close to valid examples. A GAN models adversarial learning as a two-player, zero-sum game between the discriminator and generator, and solves it as a minimax optimization problem. While the capability of GANs in generating adversarial examples has been well-researched, the topic of making the discriminator robust to the adversarial examples is less well understood. In this position paper, we posit that a game theory based framework called a Repeated Bayesian Stackelberg Game (RBSG) can be used by the discriminator to model its interaction with the generator within an adversarial GAN setting and make its prediction mechanism more robust against adversarial examples.

Game theory-based analysis of adversarial learning settings have been recently proposed in literature (Liu and Chawla 2009), albeit not in the context of a GAN. Many of these analyses model the interaction between the two players, the adversary and the learner, as a sequential or Stackelberg game where players take turns to make their moves or actions. In the context of adversarial learning GANs, the discriminator's action is to select a prediction mechanism from a set of prediction mechanisms. On the other hand, the generator's action is to select a perturbation function from a set of perturbation functions to create adversarial examples. As the game is sequential, the discriminator moves first, and, consequently, the generator can observe the discriminator's prediction mechanism (action). However, when the discriminator selected its prediction mechanism, it was unaware of the generator's action - how and if the generator created an adversarial example and gave to it as input. To handle this uncertainty, a probability distribution is used by the discriminator to model the generator's possible choices or types over its set of perturbation functions. This uncertainty model used by the generator yields a Bayesian Stackelberg game. In (Grosshans et al. 2013), it was shown that the selected actions of the discriminator and generator calculated using an adversarial learning Bayesian Stackelberg game correspond to a unique Nash equilibrium.

However, a shortcoming of the existing Bayesian Stackelberg game-based adversarial learning is that the probability distribution over the generator types used by the discriminator, is *guessed* by the discriminator. Not having a realistic estimate of this probability distribution can lead to incorrect calculations and incorrect action selection by the discriminator. Consequently, the game's outcome could deviate from the Nash equilibrium. To address this issue, we posit to integrate a repeated game framework with a Bayesian Stackelberg Game, as described in the next section.

Adversarial Learning as Repeated Bayesian Stackelberg Game (RBSG)

We model the interaction between the discriminator and generator as a 2-player repeated, Bayesian Stackelberg game where D and G are the two players representing the discriminator and generator respectively. The objective of D is to learn the probabilities with which G uses different adversarial strategies. Following (Grosshans et al. 2013), each input is denoted by the tuple (X_i, y_i, z_i), where X_i is the set

of input values, y_i is the ground truth label for X_i, and z_i is the label (adversarial or non-adversarial) for X_i intended by G. D takes X_i as input and outputs a label $f_W(X_i)$ using its prediction mechanism, where W is a set of attributes that parameterizes D's prediction mechanism. We assume $W \in \mathbf{W}$, where $\mathbf{W}$ is a set of prediction mechanisms that D is aware of through prior training. D's utility for predicting a set of inputs, $\mathbb{X} = \{X_i\}$, can be expressed as:

$$\hat{u}_d(W, \mathbb{X}, c_d) = -\sum_{i=1}^{|\mathbb{X}|} c_d(X_i)(f_W(X_i) - y_i))^2 + \Omega_d(f_W),$$

where $\Omega_d(f_W) = ||f_W||$ is a regularizer term and c_d is the prediction cost to D for input X_i. Here, $|f_W(X_i) - y_i|$ gives the error in prediction by D and the first term on the r.h.s. can be considered as D's penalty for incorrect prediction. The problem facing D in evaluating the above equation is that it cannot distinguish whether an input X_i is valid versus adversarial. We assume that G uses a perturbation function to generate adversarial example $\bar{X}_i$ from valid example X_i. The degree of perturbation however might vary; different types of generators could perturb the input to different degrees. In general, we say that a generator, G^{θ_j}, of type $\theta_j \in \Theta$, will perturb the input using perturbation function ϕ_j where, $\phi_j(X_i, \theta_j) = \bar{X}_i$. D's utility function can then be rewritten as:

$$\hat{u}_d(W, \mathbb{X}, c_d, \Theta) = -\sum_{\theta_j \in \Theta} p(\theta_j)$$

$$\left(\sum_{i=1}^{|\mathbb{X}|} c_d(\phi_j(X_i, \theta_j))(f_W(\phi_j(X_i, \theta_j)) - y_i))^2\right) + \Omega_d(f_W)$$

where p is a probability distribution over G's set of types Θ.

We model the generator G's utility in a similar manner. Only, we note that G's objective is to get D's output to correspond to its intended label z_i for input example X_i. The utility received by G for set of adversarial inputs is then given by:

$$\hat{u}_g(W, \bar{\mathbb{X}}, c_g) = -\sum_{i=1}^{|\bar{\mathbb{X}}|} c_g(\bar{X}_i)(f_W(\bar{X}_i) - z_i)^2 + \Omega_g(X_i, \bar{X}_i),$$

where $c_g()$ is the benefit that G gets from this misprediction by D and $\Omega_g(X_i, \bar{X}_i)$ is G's effort to perturb X_i (Alfeld, Zhu, and Barford 2017).

To solve this game, D finds the best weight parameters for its classifer $W^* \in \mathbf{W}$, that satisfies $W^*[\phi] = arg\max_{\mathbf{W}} \hat{u}_d(\mathbf{W}, \mathbb{X}, c_d, \Theta)$. Correspondingly, G observes the W selected by D and then select a suitable type (or perturbation) ϕ^* that satisfies $\phi^*[W] = arg\max_{\bar{\mathbb{X}}} \hat{u}_g(W, \bar{\mathbb{X}}, c_g)$. The Bayes-Nash equilibrium of the game is then given by the pair (W^*, ϕ^*).

Till now, we assumed that while solving for $W^*[\phi]$, D has information about p, the probability distribution of G's types (perturbation functions). However, in real-life settings, it is unrealistic for D to have exact knowledge of its adversary, G's type distribution $p()$. To address this issue, we propose that D simulates possible games with a virtual generator G'. Each game is represented as a game tree with alternating moves by D and G upto a specified horizon h. As there are could be presumably many possible perturbation functions (types) used by G' at each of its moves, yielding an infeasibly large game tree, D resorts to probabilistic sampling of the game tree using Monte Carlo Tree Search (MCTS) (Browne and et al. 2012). A schematic for our approach is shown in Fig. 1. Within the repeated game, D can use fictitious play or Bayesian learning with G' to observe the different perturbation function selection frequencies of $G'$and estimate the probability distribution over Gs types more precisely. This would enable D to calculate p and the Nash equilibrium of the RBSG more accurately. As our ongoing work, we are implementing the proposed RBSG approach for adversarial learning for text classification. To the best of our knowledge, our work is one of the first attempts to address robustness issues of the discriminator in adversarial GAN settings using a repeated game framework.

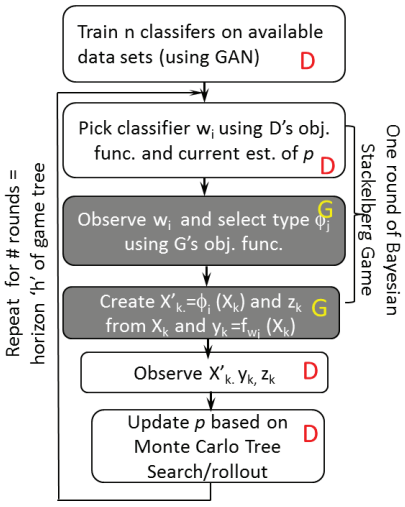

Figure 1: Schematic of our proposed GAN-based adversarial learning model using repeated Bayesian Stackelberg game.

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, February 4-9, 2017, San Francisco, California, USA.*, 1274–1280.

Browne, C., and et al. 2012. A survey of monte carlo tree search methods. *IEEE Trans. Comput. Intel. and AI in Games* 4(1):1–43.

Goodfellow, I. J., and et al. 2014. Generative adversarial nets. In *NIPS*, 2672–2680.

Grosshans, M.; Sawade, C.; Bruckner, M.; and Scheffer, T. 2013. Bayesian games for adversarial regression problems. In *ICML*, III–55–III–63.

Huang, L.; Joseph, A. D.; Nelson, B.; Rubinstein, B. I. P.; and Tygar, J. D. 2011. Adversarial machine learning. In *Proc. 4th ACM Workshop on Security and Artificial Intelligence*, 43–58.

Liu, W., and Chawla, S. 2009. A game theoretical model for adversarial learning. In *2009 IEEE ICDM Workshops*, 25–30.