Cloudy with a Chance of Poaching: Adversary Behavior Modeling and Forecasting with Real-World Poaching Data

Debarun Kar [*1] & Benjamin Ford [*1], Shahrzad Gholami[1], Fei Fang[2], Andrew Plumptre[3], Milind Tambe[1], Margaret Driciru[4], Fred Wanyama[4], Aggrey Rwetsiba[4]
[1]University of Southern California, Los Angeles, CA, 90089, {dkar,benjamif,sgholami,tambe}@usc.edu
[2]Harvard University, Boston, MA, 02138, fangf07@seas.harvard.edu
[3]Wildlife Conservation Society, New York City, NY, 10460, aplumptre@wcs.org
[4]Uganda Wildlife Authority, Uganda, {margaret.driciru,fred.wanyama,aggrey.rwetsiba}@ugandawildlife.org

ABSTRACT

Wildlife conservation organizations task rangers to deter and capture wildlife poachers. Since rangers are responsible for patrolling vast areas, adversary behavior modeling can help more effectively direct future patrols. In this innovative application track paper, we present an adversary behavior modeling system, INTERCEPT (INTERpretable Classification Ensemble to Protect Threatened species), and provide the most extensive evaluation in the AI literature of one of the largest poaching datasets from Queen Elizabeth National Park (QENP) in Uganda, comparing INTERCEPT with its competitors; we also present results from a month-long test of INTERCEPT in the field. We present three major contributions. First, we present a paradigm shift in modeling and forecasting wildlife poacher behavior. Some of the latest work in the AI literature (and in Conservation) has relied on models similar to the Quantal Response model from Behavioral Game Theory for poacher behavior prediction. In contrast, INTERCEPT presents a behavior model based on an ensemble of decision trees (i) that more effectively predicts poacher attacks and (ii) that is more effectively interpretable and verifiable. We augment this model to account for spatial correlations and construct an ensemble of the best models, significantly improving performance. Second, we conduct an extensive evaluation on the QENP dataset, comparing 41 models in prediction performance over two years. Third, we present the results of deploying INTERCEPT for a one-month field test in QENP - *a first for adversary behavior modeling applications in this domain*. This field test has led to finding a poached elephant and more than a dozen snares (including a roll of elephant snares) before they were deployed, potentially saving the lives of multiple animals - including endangered elephants.

CCS Concepts

•**Computing methodologies** → **Multi-agent systems;**

Keywords

Innovative Applications, Human Behavior Modeling, Wildlife Conservation, Deployed Applications

[*]Benjamin Ford & Debarun Kar are both first authors of this paper.

Appears in: *Proceedings of the 16th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2017), S. Das, E. Durfee, K. Larson, M. Winikoff (eds.), May 8–12, 2017, São Paulo, Brazil.*

1. INTRODUCTION

Wildlife crime continues to be a global crisis as more animal species are hunted toward extinction [33, 29]. Species extinction has dire consequences on ecosystems and the local and national economies that depend on them (e.g., eco-tourism, ecosystem services). To combat this trend, wildlife conservation organizations send well-trained rangers to patrol in protected conservation areas to deter and capture poachers and also to confiscate any tools used for illegal activities that they find. At many sites, rangers collect observation data on animals, poachers, and signs of illegal activity. Given the magnitude of the wildlife poaching problem and the difficulty of the patrol planning problem, patrol managers can benefit from tools that analyze data and generate forecasts of poacher attacks - the focus of this paper. In working with real-world wildlife crime data, this innovative application paper illustrates the importance of research driven by data from the field and real-world trials. This work potentially introduces a paradigm shift in showing how adversary modeling ought to be done for deployed security games [30, 8], particularly in domains such as green security games [11, 15, 23, 20], where data is sparse compared to settings such as urban crime [40, 1]. Security games have received significant attention at AAMAS [17, 16, 22, 2, 15], and past work in security games has often focused on behavioral models that are learned from and tested in human subject experiments in the laboratory, which provides a large amount of attacker choice data over a small number of targets [38, 25, 15]. The Quantal Response model is one example that models boundedly rational attackers' choices as a probability distribution via a Logit function [38]. However, the wildlife crime domain introduces a set of real-world challenges (e.g., rangers collect limited, noisy data over a large number of targets with rich target features) that require behavior modeling efforts to not only focus more on real-world data and less on laboratory data, but also not rely on plentiful attack data.

Outperforming previous laboratory-developed models [38, 25], CAPTURE [24] is a two-layered model, developed using real-world wildlife poaching data, that incorporates key insights and addresses the challenges present in wildlife crime data. CAPTURE's top layer attempts to predict the "attackability" of different targets, essentially providing predictions of poacher attacks. The bottom observation layer predicts how likely an attack that has occurred would be observed given the amount of patroller coverage (also known as effort). CAPTURE models the attackability layer as a hidden layer and uses the Expectation Maximization (EM) algorithm to learn parameters for both layers simultaneously. Moreover, CAPTURE also contains a Dynamic Bayesian Network, allowing it to model attacker behavior as being temporally dependent on past attacks. The CAPTURE model, the current state-of-the-art in the

wildlife crime domain, represents a level of complexity not previously seen in behavior modeling in the security game literature.

While the focus of CAPTURE is on the observation layer's performance (i.e., "Where will patrollers observe past poaching attacks given their patrol effort?"), our focus is on forecasting where future attacks will happen and thus we are interested in the attackability layer's predictions and performance (e.g., "Where will poachers attack next?"). However, CAPTURE's attackability predictions would sometimes predict too many targets to be attacked with a high probability and would thus have poor performance, as discussed in more detail later in the paper. Given that CAPTURE embodied the latest in modeling adversary behavior in this domain, our first attempt focused on three different enhancements to CAPTURE: replacement of the observation layer with a simpler layer adapted from [6] (CAPTURE-LB), modeling attacker behavior as being dependent on the defender's historical coverage in the previous time step (CAPTURE-PCov), and finally, exponentially penalizing inaccessible areas (CAPTURE-DKHO). Unfortunately, all of these attempts ended in failure.

While poor performance is already a significant challenge, there are two additional, important shortcomings of CAPTURE and other complex models in this same family. First, CAPTURE's learning process takes hours to complete on a high-performance computing cluster - unacceptable for rangers in Uganda with limited computing power. Second, CAPTURE's learned model is difficult to interpret for domain experts since it makes predictions based on a linear combination of different decision factors; the values of all its parameters' feature weights (i.e., 10 weights and a free parameter for the attack layer) need to be simultaneously accounted for in a single interpretation of poacher preferences. These limitations and the poor performance of CAPTURE, the most recent in a long line of behavioral game theory models, drove us to seek an alternative modeling approach.

This paper presents INTERCEPT (INTERpretable Classification Ensemble to Protect Threatened species), a new adversary behavior modeling application, and its three major contributions. (1) Given the limitations of traditional approaches in adversary behavior modeling, INTERCEPT takes a fundamentally different modeling approach, decision trees, and delivers a surprising result: although decision trees are simpler and do not take temporal correlations into account, they perform significantly better than CAPTURE (a complex model that considers temporal relationships), its variants, and other popular machine learning models (e.g., Logistic Regression, SVMs, and AdaBoost). Furthermore, decision trees satisfy the fundamental requirement of interpretability; without an interpretable model, relevant authorities would not test INTERCEPT in the field, thus completely defeating the spirit of innovative applications research. However, decision trees do not take into account the spatial correlations present in this dataset, and we introduce a spatially aware decision tree algorithm, BoostIT, that significantly improves recall with only modest losses in precision. To further augment INTERCEPT's performance, we construct an ensemble of the best classifiers which boosts predictive performance to a factor of 3.5 over the existing CAPTURE model. (2) These surprising results raise a fundamental question about the future of complex behavioral models (e.g., Quantal Response based security game models [38, 25, 24]) in real-world applications. To underline the importance of this question, we conduct the most extensive empirical evaluation to date of the QENP dataset with an analysis of 41 different models and a total of 193 model variants (e.g., different cost matrices) and demonstrate INTERCEPT's superior performance to traditional modeling approaches. (3) As a first for adversary behavior modeling applications applied to the wildlife crime domain, we present the results of a *month long* real-world deployment of INTERCEPT: compared to historical observation rates of illegal activity, rangers that used INTERCEPT observed 10 times the number of findings than the average. In addition to many signs of trespassing, rangers found a poached elephant, a roll of elephant snares, and a cache of 10 antelope snares before they were deployed (pictures in Figure 1). Each confiscated snare represents an animal's life saved; while the rangers' finding of a poached elephant carcass is a grim reminder that poachers are active, these successful snare confiscations demonstrate the importance of real-world data in developing and evaluating adversary behavior models.

Figure 1: Ashes and snare found by rangers directed by INTERCEPT. Photo credit: Uganda Wildlife Authority ranger.

2. RELATED WORK

There have been recent efforts on planning effective patrol strategies to combat poaching [11, 10], which have led to an application, PAWS, being deployed in the field. However, unlike our work, the focus of PAWS has been on generating risk-based randomized patrols and not on predicting poacher attacks. INTERCEPT's predictive analysis is essential to efficiently allocating limited ranger patrolling resources and can thus be the driving force for further prescriptive analysis (i.e., patrol planning). In addition, the deployment of our work in the field has shown a level of success that has not been previously seen in PAWS.

Models inspired by previous work in behavioral game theory [21, 26, 4, 31] have been extensively used in recent years to predict human behavior in simultaneous-move games [34, 35, 36] and also to predict adversary behavior in multiple security game domains including counter-terrorism [25], wildlife crime [37, 15, 24], fisheries protection [12, 3], and even in urban crime [41, 1, 39] where a Dynamic Bayesian Network similar to CAPTURE was used. Furthermore, researchers in the conservation community have also used two-layered behavioral models similar to CAPTURE to predict future poaching behavior [5]. CAPTURE is only the latest model in a long chain of behavioral models used for human behavior prediction in game theory and also in the conservation literature. However, as detailed in later sections, CAPTURE suffers from several limitations and performs poorly in predicting attacks in the real-world wildlife crime dataset.

Modeling and predicting other agents' behavior has also been studied in application domains such as RoboCup and military operations [19, 32], but such predictions are often based on real-time information, which is not available in this particular problem or dataset. There have been other attempts to predict poacher behavior in Machine Learning research: [28] uses association rule mining to get a single rule that classifies locations with poaching attack, but the expressiveness of this approach is limited due to the single rule; [27] uses standard classification algorithms to predict the attacka-

bility of targets and uses a regression model to predict the probability of attack. However, this work only reports accuracy, which is not an informative metric given the extreme class imbalance present in real-world wildlife crime datasets (i.e., just predicting no attacks everywhere could possibly lead to high accuracy) and the potentially high cost of false negatives (i.e., an endangered animal may be poached). Moreover, our decision tree based model can be seen as a generalization of this work since we can view a set of rules (instead of just one) that describe the model in richer terms than a single rule.

3. WILDLIFE CRIME DATASET

The following discussion is on wildlife crime data collected over 13 years at the Queen Elizabeth National Park (QENP) in Uganda. QENP (Figure 2) is a wildlife conservation area covering 1,978 square kilometers. Among their many duties, wildlife park rangers there conduct foot patrols to monitor wildlife habitat, apprehend any poachers sighted inside the park, and collect data on animal signs and signs of illegal human activity.

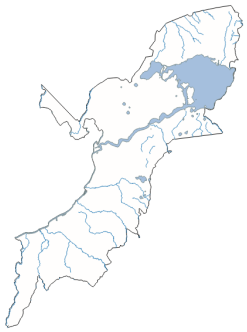

Figure 2: QENP

3.1 Dataset Challenges

Because this is a real-world geospatial crime dataset, it is important to understand the inherent challenges in analyzing its contents, such as nonlinear relationships between features [14]. Additionally, data can only be collected in areas that are patrolled, and even in the areas that are patrolled, poaching signs may remain undetected. This occurs because poaching signs (such as snares) are often well-hidden, and rangers may need to conduct a thorough patrol in order to detect any attack – an infeasible task to undertake for all targets all the time due to limited patrolling resources. This real-world constraint not only leads to uncertainty in the negative class labels (i.e., when poaching signs are not observed we are uncertain whether an attack actually happened at the corresponding target or not) but also results in a small number of positive samples being recorded in the dataset thus creating a huge class imbalance. As such, it is necessary to evaluate the attack prediction model's performance with metrics that account for this uncertainty, such as those for Positive and Unlabeled Learning (PU Learning) [18], and are discussed in more detail in the following sections.

3.2 Dataset Composition

The entire QENP area was discretized into 1 square kilometer grid cells (total 2,522 cells), each as a potential target of poaching. For each target, the ranger patrol effort level (i.e., coverage) and observed illegal human activity signs (e.g., poached animal carcasses, snares) were recorded. In addition, each target is associated with a non-static average ranger patrol effort value and a set of static features (that are constant throughout the entire time period): terrain features such as habitat (the terrain type and relative ease of travel) and terrain slope; distances to nearby roads, water bodies, patrol posts, and villages; and animal density.

For the following analysis, we examine animal poaching data from 2003-2015. We aim to find the targets that are liable to be attacked since predicting the attackability of targets is helpful in guiding future patrols. We assume a target is attackable if an attack is ever observed at that target at any point in time. Therefore, when creating training sets, we combine observations from the entire training period for each target and label it as attackable if any observations were made.

Given the uncertainty in negative labels, there are bound to be training and testing samples that contradict one another. We consider a sample in the training set and a sample in the testing set to be contradictory when they have the same combination of static domain features values (e.g., terrain, distances, animal density) and non-static patrol coverage amount (i.e., low or high coverage) but different class labels (attacked or not attacked). These contradictions introduce additional noise in evaluating the performance of learned models and would thus cause any model to perform poorly on said contradictory data. As such, we remove these contradictions, about 10% of the data, from testing sets.

4. CAPTURE AND PROPOSED VARIANTS

The natural first step towards predicting future poaching attacks based on our real-world wildlife crime dataset was to use the best previous model, CAPTURE [24]. CAPTURE was shown to have superior predictive performance to a number of other standard models in the behavioral game theory literature (e.g., Quantal Response (QR) [38], Subjective Utility Quantal Response (SUQR) [25]).

To make attackability predictions, we discretized the protected area into a set of targets I. Each target $i \in I$ has a set of domain-specific features $x_i \in x$ such as animal density d_i and distance to water. In a given time period t, a target i will be patrolled/covered by rangers with probability $c_{t,i}$.

CAPTURE consists of a two-layered behavior model. CAPTURE's first layer, the attackability layer, computes the probability that a poacher will attack a given target i at time step t. Similar to SUQR, which has been used to describe human players' stochastic choice of actions in security games, CAPTURE predicts attacks based on a linear combination of domain features $x_{t,i}$, ranger coverage probability $c_{t,i}$ at the current time step t, and whether the target was attacked in the previous time step $a_{t-1,i}$. With this last feature, $a_{t-1,i}$, CAPTURE models attacker behavior as being temporally dependent on past attacks.

$$p(a_{t,i} = 1 | a_{t-1,i}, c_{t,i}, x_{t,i}) = \frac{e^{\lambda^\intercal [a_{t-1,i}, c_{t,i}, x_{t,i}, 1]}}{1 + e^{\lambda^\intercal [a_{t-1,i}, c_{t,i}, x_{t,i}, 1]}} \quad (1)$$

λ is a parameter vector representing the importance of the features.

CAPTURE's second layer, the observation layer, computes the probability that rangers will observe an attack if poachers did attack that patrolled area based on a subset of domain features (e.g., habitat and slope) $\hat{x}_{t,i}$ and ranger coverage probability $c_{t,i}$.

$$p(o_{t,i} = 1 | a_{t,i} = 1, c_{t,i}, \hat{x}_{t,i}) = c_{t,i} \times \frac{e^{\omega^\intercal [\hat{x}_{t,i}, 1]}}{1 + e^{\omega^\intercal [\hat{x}_{t,i}, 1]}} \quad (2)$$

ω is a parameter vector that measures how domain features impact observation probability. The model parameters (λ, ω) that can maximize the likelihood of observations are estimated via the Expectation Maximization (EM) algorithm.

However, CAPTURE has a few limitations that lead to poor predictive performance in its *attackability layer*. First, CAPTURE's attackability predictions would sometimes predict too many targets to be attacked with a high probability (e.g., 80% of the targets will be attacked with almost 100% probability), leading to poor performance (see Section 7). One explanation is CAPTURE's parameter learning algorithm focuses on maximizing the performance of the observation layer rather than on the attackability layer. As the observation layer acts as a filter for the attackability layer, CAPTURE's learning process will converge to solutions that obtain de-

cent performance for the observation layer even if the attackability layer's performance is poor.

Therefore, we propose several novel variants of CAPTURE as attempts to improve its predictions. In an attempt to restrict the degrees of freedom in the observation layer, and thus restrict the values the attackability layer can take in the learning process, we propose **CAPTURE-LB** which replaces the observation layer with a simpler observation layer, adapted from [6], described as follows:

$$p(o_{t,i} = 1 | a_{t,i} = 1, c_{t,i}) = 1 - e^{-\beta \times c_{t,i}} \tag{3}$$

where $\beta \in [0, 1]$ is the parameter that estimates the detection efficiency. This not only provides a straightforward way of assessing the performance of patrol effort to observations but also has a smaller chance of overfitting, due to fewer parameters. For a given attack probability $p(a_{t,i} = 1)$, the unconditional probability of observing an attack at target i at time step t is given by:

$$p(o_{t,i}) = p(a_{t,i} = 1) \times p(o_{t,i} = 1 | a_{t,i} = 1, c_{t,i}) \tag{4}$$

Second, CAPTURE's attackability layer assumes that poachers plan attacks based on the patrol coverage in the current time step, which may not be realistic in the real world as the poachers may not get up-to-date information about the current patrol strategy and thus would rely on historical patrol coverage instead [11]. Therefore, we propose another variant of CAPTURE, **CAPTURE-PCov**, that learns based on the previous time step's patrol coverage instead of the current time step's patrol coverage (Equation 5). Similarly, we propose **CAPTURE-PCov-LB**, a model that uses the attackability layer of CAPTURE with previous coverage as a feature but instead uses the LB observation layer defined in Equation 3.

$$p(a_{t,i} = 1 | a_{t-1,i}, c_{t-1,i}, x_{t,i}) = \frac{e^{\lambda^\intercal [a_{t-1,i}, c_{t-1,i}, x_{t,i}, 1]}}{1 + e^{\lambda^\intercal [a_{t-1,i}, c_{t-1,i}, x_{t,i}, 1]}} \tag{5}$$

Finally, CAPTURE's attackability predictions fail to take into account the domain knowledge that inaccessible and unattractive areas of the park will not be attacked with high probability, and we thus propose another variant **CAPTURE-DKHO**, which is the same as CAPTURE-PCov-LB except that it exponentially penalizes the attractiveness of inaccessible areas (Equation 6).

$$p(a_{t,i} = 1 | a_{t-1,i}, c_{t-1,i}, x_{t,i}) = \frac{e^{\lambda^\intercal [a_{t-1,i}, c_{t-1,i}, x'_{t,i}, 1]}}{1 + e^{\lambda^\intercal [a_{t-1,i}, c_{t-1,i}, x'_{t,i}, 1]}} \tag{6}$$

x' corresponds to the linear combination of features x but with the modified habitat feature $\sigma'_i = -\sigma_i e^{\sigma_i}$ which heavily penalizes high habitat values (i.e., hard to access areas).

5. INTERCEPT

The attempts of using the best previous model CAPTURE and the more complex variants of CAPTURE, proposed to address the limitations of CAPTURE, all suffered from poor attackability prediction performance as shown in Section 7. The natural progression then would have been to pursue more complex models in this behavioral game theory family of models with the expectation that they would improve performance on our real-world data. However, as reported in [24], complex models such as CAPTURE and its variants incur heavy computational costs; it takes approximately 6 hours for these models to complete execution. In addition, these models become more difficult to interpret when the dimensionality of the feature space increases (e.g., more numerical values to simultaneously account for in a single interpretation). We wanted to use models that would address all of these shortcomings by, not only significantly reducing computational costs so as to be usable by rangers with limited computing power in Uganda, but also remain interpretable to domain experts as the feature space dimensionality increases. All of these factors pointed against using more complex behavioral models. Therefore, we break from the current trend in behavior modeling in security games and model adversary behavior in terms of decision tree-based behavior models, even though we were initially skeptical about its predictive capabilities. Surprisingly, this simpler approach led to significant improvements in performance over the prior state-of-the-art (i.e., CAPTURE).

5.1 BoostIT

A binary decision tree D is trained on a set Θ of independent variables x (the domain features), a dependent variable o (attack observations), and outputs a binary classification D_i for each target i: {not attacked ($D_i = 0$), attacked ($D_i = 1$)}. A decision tree's negative predictions for a test set Ψ are denoted by $P^-_\Psi(D)$ and positive predictions by $P^+_\Psi(D)$ (i.e., vectors of binary predictions).

Crime hot spots are part of a well-known theory in Criminology [9] that views crime as an uneven distribution; crime is likely to be concentrated in particular areas called hot spots. If a particular geographic area has a high concentration of predicted attacks, it is reasonable to interpret these predictions as a hot spot prediction (i.e., predicting a high concentration of crime). While CAPTURE explicitly models attacks as a probability distribution decided by a linear combination of feature values and thus can implicitly represent the hot spots with soft boundaries in the geographic space, decision trees' rules with hard boundaries in the feature space would lead to fine-grained segmentations in the geographic space and is thus less capable of representing hot spots. As such, we designed the **Boost**ed decision tree with an **It**erative learning algorithm (henceforth referred to as BoostIT) (Algorithm 1), where proximity to a predicted hot spot is encoded as an additional input feature.

Algorithm 1 BoostIT

$D^0 \leftarrow$ LEARNDECISIONTREE(Θ^0)
repeat
  $h^\Theta \leftarrow$ CALCHOTSPOTPROXIMITY($P_{\Theta^{m-1}}(D^{m-1}), \alpha$)
  $h^\Psi \leftarrow$ CALCHOTSPOTPROXIMITY($P_{\Psi^{m-1}}(D^{m-1}), \alpha$)
  $\Theta^m \leftarrow$ ADDFEATURE(Θ^0, h_Θ)
  $\Psi^m \leftarrow$ ADDFEATURE(Ψ^0, h_Ψ)
  $D^m \leftarrow$ LEARNDECISIONTREE(Θ^m)
  $m = m + 1$
until iterationStoppingLevelReached
return P

D^0 is the initial decision tree learned without the hot spot proximity feature h, and Θ^0 and Ψ^0 correspond to the initial training and test sets, respectively. For each level of iteration m, a feature h^Θ (and h^Ψ) is computed for each target $i \in I$ that corresponds to whether that target is close to a predicted hot spot in the training (and test sets); for example, if a target $i \in P_{\Theta^{m-1}}(D^{m-1})$ is adjacent to α or more targets in $P^+_{\Theta^{m-1}}(D^{m-1})$ (i.e., targets that are predicted to be positive), then $h^\Theta_i = 1$. We then re-learn the decision tree at each iteration m with a feature augmented dataset Θ^m. As an example, BoostIT may add a feature to a target i that i is near a hot spot if there are two adjacent targets that are predicted to be attackable. In the next iteration, this new feature ("near a hot spot") will get used in learning about predicting attacks on i. This continues until an iteration criterion is reached. Note that the test set Ψ is not used while learning new decision trees (only training data Θ is used) and is only used to update the test set prediction P_Ψ. In the rest of the paper, we will refer to BoostIT with an α

as BoostIT-αNearestNeighbors (or BoostIT-αNN). With this algorithm, the final decision tree D^m would generally predict more positive predictions with concentrated areas (i.e., hot spots) compared to D^0, but the set of predictions of D^m is not necessarily a superset of the set of predictions of D^0.

Although we are primarily interested in predicting attackability, we can also predict where patrollers would observe attacks by cascading attackability predictions with the LB observation layer (Equation 3). We convert the unconditional observation probability, derived from the cascaded model (Equation 4), to binary predictions by classifying samples as observed/not observed based on whether they are above or below the mean respectively.

5.2 INTERCEPT: Ensemble of Experts

We investigated the predictions of the traditional decision tree and BoostIT and observed that they are diverse in terms of their predictions. Here, by diversity, we mean that they predict attacks at a variety of targets. Therefore, while one model may fail to correctly classify a particular target as attacked, another model may succeed. This indicates the ability of different models to correctly learn and predict on different regions of the feature space. For example, let us consider the following three models: (i) DecisionTree, (ii) BoostIT-3NN and (iii) BoostIT-2NN. While computing pairwise disagreement between the models' attackability predictions, we observed that: (i) DecisionTree and BoostIT-3NN disagree on 105 out of 2211 target samples; (ii) DecisionTree and BoostIT-2NN disagree on 97 out of 2211 samples; and (iii) BoostIT-3NN and BoostIT-2NN disagree on 118 out of 2211 samples. This observation led us to consider combining the best decision tree and BoostIT based models, thus forming INTERCEPT–an ensemble of experts.

Because of uncertainty in negative labels, INTERCEPT considers not only decision tree models with the standard false positive (FP) cost of one, but also decision trees with various FP costs. For a decision tree with FP cost of 0.6, during the learning process, the decision tree will not receive the full penalty of 1 but will instead receive a penalty of 0.6 for each false positive prediction it makes.

In INTERCEPT, each expert model voted for the final attack prediction on a particular target. We considered three types of voting rules to determine whether a target should be predicted to be attacked by the ensemble: (a) majority of the experts predict an attack; (b) all experts predict an attack; and (c) any one expert predicts an attack. INTERCEPT uses the best voting rule: majority.

We considered ensembles with three and five experts. Having at most 5 experts makes the ensemble easily interpretable. In other words, the final prediction at a target is due to only 5 decision rules at a maximum, and it is easy to walk the domain experts through the 5 rules in a way that the logic is easily verified.

6. EVALUATION METRICS

To evaluate INTERCEPT and other models, we first prepared two separate train/test splits on the dataset. For one dataset, we trained on data from 2003 to 2013 and evaluated our models on data in 2014, and for the other dataset, we trained on data from 2003 to 2014 and evaluated on data from 2015. Prior to discussing the evaluation results, we briefly discuss the metrics we use for computing our performance on predicting attackability and observed attacks.

Any metric to evaluate the *attackability* of targets in domains such as our wildlife poaching domain must account for the uncertainty in negative class labels. Therefore, in addition to standard metrics (Precision, Recall, and F1-score) that are used to evaluate models on datasets where there is no uncertainty in the underlying ground truth, we also evaluate our models with a metric that accounts for the uncertainty present in our dataset. The metric introduced in [18], henceforth referred to as L&L, is an appropriate metric since it is specifically designed for models learned on Positive and Unlabeled (PU) datasets (i.e., datasets with uncertain negative labels). L&L is defined in equation 7, where r denotes the recall and $Pr[f(Te) = 1]$ denotes the probability of a classifier f making a positive class label prediction. We compute $Pr[f(Te) = 1]$ as the percentage of positive predictions made by our model on a given test set.

$$L\&L(D, Te) = \frac{r^2}{Pr[f(Te) = 1]} \tag{7}$$

As we are certain about the positive samples in our dataset, L&L rewards a classifier more for correctly predicting where attacks have occurred (i.e., positive labels). However, it also prevents models from predicting attacks everywhere, via its denominator, and ensures that the model is selective in its positive predictions.

We also evaluate the models in terms of *observation* predictions. Here, we report standard metrics (Precision, Recall, and F1-score). In addition, we also compute the area under the Precision-Recall curve (PR-AUC). PR-AUC is a more appropriate metric for evaluating models on datasets with severe class imbalance [7] compared to area under the ROC curve. When there are many more negative points than positive points, the model can make many false positive predictions and the false positive rate would still be low, and thus the ROC curve becomes less informative. In contrast, precision better captures how well the model is making correct positive predictions given a small number of positive examples. L&L is no longer used to evaluate the observation probability model as there is no uncertainty in terms of the observations, i.e., we either observed or did not observe an attack, and we are measuring the model's ability to predict whether we will observe attacks at already attacked targets.

7. EVALUATION ON HISTORICAL REAL-WORLD PATROL DATA

To compare INTERCEPT with its competitors, we conducted a thorough investigation of the performance of 41 different models and 193 variants (a detailed list is available in the online appendix [1]). This is one of the largest evaluation efforts on a real-world dataset in the wildlife crime domain, and we compared INTERCEPT against the previous best model CAPTURE, its variants, and other machine learning approaches such as Support Vector Machines (SVM), AdaBoosted Decision Trees, and Logistic Regression[2]. All the numbers highlighted in **bold** in the tables indicate the results of the best performing models in that table. The best performing INTERCEPT system is an ensemble of five decision trees with majority voting. The five decision trees are: a standard decision tree, two BoostIT decision trees ($m = 1$) with $\alpha = 2$ and $\alpha = 3$ respectively, and two decision trees with modified false positive costs 0.6 and 0.9 respectively. Note that, due to data collection methodology changes in 2015, the distribution of attack data in 2015 is significantly different than all other previous years; 2015 is a difficult dataset to test on when the training dataset of 2003-2014 represents a different distribution of attack data, and we will demonstrate this impact in the following evaluation.

7.1 Attackability Prediction Results

In Tables 1 and 2, we show a comparison of the performance between our best INTERCEPT system (the five decision tree ensem-

[1] http://teamcore.usc.edu/papers/AAMAS2017_Ensemble_Appendix.pdf

[2] Note that due to data confidentiality agreements, we are unable to show an example decision tree in this paper.

Classifier Type	F1	L&L	Precision	Recall
PositiveBaseline	0.06	1	0.03	1
UniformRandom	0.05	0.51	0.03	0.50
CAPTURE	0.31	3.52	0.25	0.39
CAPTURE-PCov	0.13	1.29	0.08	0.48
CAPTURE-PCov-LB	0.08	0.87	0.04	0.58
CAPTURE-DKHO	0.10	1.05	0.06	0.67
INTERCEPT	**0.41**	**5.83**	0.37	0.45

Table 1: Attackability Prediction Results on 2014 Test Data

Classifier Type	F1	L&L	Precision	Recall
PositiveBaseline	0.14	1	0.07	1
UniformRandom	0.19	0.50	0.11	0.50
CAPTURE	0.21	1.08	0.13	0.63
CAPTURE-PCov	0.19	0.87	0.11	0.57
CAPTURE-PCov-LB	0.18	0.69	0.11	0.46
CAPTURE-DKHO	0.20	0.71	0.12	0.5
INTERCEPT	**0.49**	**3.46**	0.63	0.41

Table 2: Attackability Prediction Results on 2015 Test Data

Classifier Type	F1	L&L	Precision	Recall
Weighted DecisionTree	0.11	1.01	0.06	0.48
SVM-BestFPCost-0.3	0.13	1.18	0.46	0.45
Logistic Regression	-	-	-	0
AdaBoostDecisionTree-BestFPCost-0.2	0.13	1.22	0.07	0.48
INTERCEPT	**0.41**	**5.83**	0.37	0.45

Table 3: Additional Attackability Prediction Results on 2014 Test Data

Classifier Type	F1	L&L	Precision	Recall
Weighted DecisionTree	0.25	1.42	0.15	0.69
SVM-BestFPCost-0.25	0.19	0.72	0.12	0.43
Logistic Regression	-	-	-	0
AdaBoost-DT-BestFPCost-0.15	0.21	0.86	0.13	0.49
INTERCEPT	**0.49**	**3.46**	0.63	0.41

Table 4: Additional Attackability Prediction Results on 2015 Test Data

ble with majority voting), the current state-of-the-art CAPTURE, its variants, and other baseline models towards accurately predicting the attackability of targets in QENP for years 2014 and 2015, respectively. The PositiveBaseline corresponds to a model that predicts every target to be attacked ($p(a_{t,i}) = 1; \forall i, t$), and the UniformRandom corresponds to the baseline where each target is predicted to be attacked or not attacked with equal probability. Note that, in this subsection, when evaluating two-layered models such as CAPTURE and its variants, we are examining the performance of just the attackability layer output, and we defer the evaluation of the observation predictions to Section 7.2. Since we evaluate the attackability predictions of our models on metrics for binary classification, the real-valued output of the attackability layer of CAPTURE and its variants were converted to a binary classification where probabilities greater than or equal to the mean attack probability were classified as positive.

We make the following observations from these tables: First, INTERCEPT completely outperforms the previous best model CAPTURE and its variants, as well as other baseline models in terms of L&L and F1 scores. For 2014, INTERCEPT outperforms CAPTURE in terms of precision, recall, F1, and L&L score. For 2015 test data, INTERCEPT represents an even larger performance increase by approximately 3.50 times (L&L score of 3.46 vs 1.08) over CAPTURE and even more so for CAPTURE-PCov (L&L score of 3.46 vs 0.87). CAPTURE-PCov doesn't even outperform the positive baseline. Second, CAPTURE performs better on the 2014 dataset (when the training and testing data were similarly distributed) than on the 2015 dataset. In contrast, INTERCEPT remained flexible enough to perform well on the difficult 2015 testing set. However, CAPTURE-PCov, the more realistic variant of CAPTURE that can actually be used for forecasting, fails to make meaningful predictions about the attackability of targets. Its similar performance to PositiveBaseline demonstrates the need for models to learn the attackability of targets independently of observation probability to avoid learning models that make incorrect inferences about the attackability of the park (e.g., the entire park can be attacked). This is particularly important in the wildlife poaching domain because, due to the limited number of security resources, rangers cannot patrol every target all the time. Therefore, the attack probability model's predictions need to be extremely precise (high precision) while also being useful indicators of poaching activities throughout the park (high recall). Third, CAPTURE-PCov-LB performs even worse than CAPTURE-PCov in terms of L&L score for these attackability predictions, although the only difference between the two models is the observation layer. This occurs because the attackability prediction layer and the observation layer are not independent of one another; with the EM algorithm, the parameters are being learned for both layers simultaneously. In addition, by incorporating domain knowledge and penalizing the unattractive areas, CAPTURE-DKHO unfortunately does not lead to a significant improvement in performance. Fourth, INTERCEPT's precision values are significantly better compared to CAPTURE-PCov in 2014 and both CAPTURE and CAPTURE-PCov in 2015 with only modest losses of recall, indicating a significant reduction in the number of false positive predictions made throughout the park.

In Tables 3 and 4, we also compare INTERCEPT with other models including: (i) a decision tree where each sample was weighted based on the patrol intensity for the corresponding target (Weighted Decision Tree); (ii) the best performing SVM; (iii) Logistic Regression (which predicted no attacks and thus metrics could not be computed); and (iv) the best performing AdaBoosted Decision Tree. INTERCEPT provides significantly better performance than these other models as well.

7.2 Observation Prediction Results

Tables 5 and 6 correspond to how accurately each model predicted the observations in our test datasets. For a fair comparison, we also cascade the attackability predictions of the PositiveBaseline and UniformRandom baselines with an LB observation layer, and convert those unconditional observation probabilities to binary predictions with a mean threshold, as was done for CAPTURE's attackability predictions. We observe the following. First, incorporating the observation model in Equation 4 improved the PR-AUC score of CAPTURE in both test datasets (for 2014, 0.36 vs 0.33; for 2015, 0.32 vs 0.29). Second, INTERCEPT outperforms the other models by a large margin, both in terms of F1 and PR-AUC, for both test datasets. Combined with the attackability results, these results demonstrate the benefit of learning more precise attackability models in order to better predict observation probability.

7.3 Impact of Ensemble and Voting Rules

INTERCEPT consists of five experts with a majority voting rule. We now investigate the impact of combining different decision trees into an ensemble, and the impact of different voting rules.

Classifier Type	F1	Precision	Recall	PR-AUC
PositiveBaseline	0.13	0.07	0.79	0.12
UniformRandom	0.09	0.05	0.46	0.07
CAPTURE	0.14	0.08	0.73	0.33
CAPTURE-PCov	0.12	0.07	0.61	0.31
CAPTURE-PCov-LB	0.13	0.08	0.48	0.36
CAPTURE-DKHO	0.16	0.09	0.72	0.33
INTERCEPT	**0.36**	0.32	0.89	**0.45**

Table 5: Observation Prediction Results on 2014 Test Data

Classifier Type	F1	Precision	Recall	PR-AUC
PositiveBaseline	0.26	0.16	0.66	0.20
UniformRandom	0.19	0.12	0.45	0.14
CAPTURE	0.29	0.18	0.70	0.29
CAPTURE-PCov	0.29	0.18	0.70	0.29
CAPTURE-PCov-LB	0.34	0.21	0.85	0.32
CAPTURE-DKHO	0.36	0.24	0.79	0.32
INTERCEPT	**0.50**	0.65	0.41	**0.49**

Table 6: Observation Prediction Results on 2015 Test Data

Classifier Type	F1	L&L	Precision	Recall
PositiveBaseline	0.06	1	0.03	1
DecisionTree	0.2	1.8	0.14	0.36
BoostIT-1NN	0.19	2.23	0.12	0.55
BoostIT-2NN	0.21	2.13	0.13	0.45
BoostIT-3NN	0.2	2.01	0.13	0.45
INTERCEPT	**0.41**	**5.83**	0.37	0.45

Table 7: Attackability Prediction Results For Decision Tree Models on 2014 Test Data

Classifier Type	F1	L&L	Precision	Recall
PositiveBaseline	0.14	1	0.07	1
DecisionTree	0.39	2.01	0.39	0.38
BoostIT-1NN	0.39	2.16	0.32	0.50
BoostIT-2NN	0.37	2.00	0.30	0.50
BoostIT-3NN	0.42	2.45	0.35	0.52
INTERCEPT	**0.49**	**3.46**	0.63	0.41

Table 8: Attackability Prediction Results For Decision Tree Models on 2015 Test Data

Tables 7 and 8 show that constructing an ensemble, INTERCEPT, significantly improves the performance of the system as a whole, compared to the performance of its individual decision tree and BoostIT members. The standard decision tree is more conservative as it predicts less false positives, leading to higher precision, but suffers from low recall.

Table 9 shows the impact that a voting rule has on performance on 2015 test data (due to space, we omit the 2014 test data as it exhibits the same trends). We evaluate the performances of the best ensemble compositions, with three and five experts for each voting rule. We observe that: (i) Ensembles which predict an attack if any one expert predicts an attack (*Any*) are significantly better in terms of recall (0.68), but do poorly in terms of precision (0.23). This is because such ensembles are more generous in terms of predicting an attack, and this leads to a significantly higher number of false positives; (ii) Ensembles with a voting rule where all experts have to agree (*All*) perform worse in terms of recall (0.16), but do best in terms of precision (0.89) as it makes less positive predictions (both true positives as well as false positives). This would mean that it would miss a lot of attacks in our domain, however; (iii) The majority voting based ensembles (*Maj*), used by INTERCEPT, provide an important balance between precision (0.63) and recall (0.41) as they are neither extremely conservative nor generous in terms of their predictions and therefore outperform other voting rules significantly (L&L of 3.46).

This analysis provides important guidance for selecting ensembles depending on the requirements of the domain. For example, if it is extremely crucial to predict as many true positives as possible and a high number of false positives is acceptable, then using an *Any* voting method would be beneficial. However, in our wildlife poaching prediction problem, we have limited security resources and therefore cannot send patrols to every target all the time. Therefore, we not only wish to limit the number of false positives but also increase the number of correct poaching predictions. The majority voting rule provides this important balance in our domain.

8. EVALUATION ON REAL-WORLD DEPLOYMENT

INTERCEPT represents a paradigm shift from complex logit-based models such as CAPTURE [24], and many others, to decision tree-based models. During development, we worked with a domain expert from the Wildlife Conservation Society to improve and validate our decision tree models and their corresponding predictions. Indeed, one advantage of shifting to a decision tree-based approach (as opposed to methods like CAPTURE) is that the underlying rules can be easily expressed to experts in non-AI fields.

After this development and evaluation on historical data was completed, we deployed INTERCEPT to the field. Based on INTERCEPT's predictions, we chose two patrol areas for QENP rangers to patrol for one month. We selected these areas (approximately 9 square km each) such that they were (1) predicted to have multiple attacks and (2) previously infrequently patrolled as rangers did not previously consider these as important as other areas (and thus are good areas to test our predictions). After providing the rangers with GPS coordinates of particular points in these areas, they patrolled these areas on foot and utilized their expert knowledge to determine where exactly in these areas they were most likely to find snares and other signs of illegal human activity (e.g., salt licks, watering holes). On each patrol, in addition to their other duties, rangers recorded their observations of animal sightings (i.e., 21 animals were sighted in one month) and illegal human activity.

We now present our key findings in Tables 10 and 11 and provide a selection of photos in Figures 1 and 3. The most noteworthy findings of these patrols are those related to elephant poaching; rangers, unfortunately, found one poached elephant with its tusks removed. However, this result demonstrates that poachers find this area, predicted by our model, attractive for poaching. On a more positive note, our model's predictions led rangers to find many snares before they caught any animals: one large roll of elephant snares, one active wire snare, and one cache of ten antelope snares. INTERCEPT's predictions assisted rangers' efforts in potentially saving the lives of *multiple animals including elephants*.

In addition to wildlife signs, which represent areas of interest to poachers, the findings of trespassing (e.g., litter, ashes) are significant as these represent areas of the park where humans were able to enter illegally and leave without being detected; if we can continue to patrol areas where poachers are visiting, rangers will eventually encounter the poachers themselves.

So as to provide additional context for these results, we present a set of base rates in Table 11. These base rates, computed in and around our proposed patrol areas, correspond to the average number of observed crimes per month from 2003-2015. Animal commercial (AnimalCom) crimes correspond to elephant, buffalo, and

Classifier Type	F1	L&L	Precision	Recall
BoostIT-3Experts-Any	0.36	2.11	0.26	0.59
BoostIT-5Experts-Any	0.34	2.13	0.23	0.68
BoostIT-3Experts-All	0.36	2.68	0.88	0.22
BoostIT-5Experts-All	0.28	1.97	0.89	0.16
BoostIT-3Experts-Maj	**0.49**	3.34	0.58	0.43
INTERCEPT	**0.49**	**3.46**	0.63	0.41

Table 9: Attackability Prediction Results For Different Ensembles on 2015 Test Data

Week#	Illegal Activity	Count
2	Trespassing	19
3	Active Snares	1
	Plant Harvesting	1
4	Poached Elephants	1
	Elephant Snare Roll	1
	Antelope Snares	10
	Fish Roasting Racks	2

Table 10: Real World Patrol Results: Illegal Activity

hippopotamus poaching; animal noncommercial (AnimalNoncom) corresponds to all other poaching and poaching via snares; and plant noncommercial (PlantNoncom) corresponds to illegal harvesting of non-timber forest products (e.g., honey). The percentile rank corresponds to the number of months where our deployed patrols recorded more observations than in the historical data. For animal noncommercial crime, there was an average of 0.73 attacks observed monthly; for our deployed patrols, there were 3 separate observations (such as a roll of elephant snares), and in 91% of the months from 2003-2015, 2 or fewer observations were recorded.

9. LESSONS LEARNED

After our extensive modifications to the CAPTURE model and our subsequent evaluation, it is important to identify the reasons why we obtained such a surprising result: decision trees outperformed a complex, domain-specific temporal model. (1) The amount of data and its quality need to be taken into consideration when developing a model. The QENP dataset had significant noise (e.g., imperfect observations) and extreme class imbalance. As such, attempting to develop a complex model for such a dataset can backfire when there does not exist sufficient data to support it. Our decision tree approach, generally regarded as simpler, benefits from being able to express non-linear relationships and can thus

Figure 3: Elephant snare roll found by rangers directed by INTERCEPT. Photo credit: Uganda Wildlife Authority ranger

Crime Type	**INTERCEPT**	Average	Percentile
AnimalCom	**1**	0.16	89%
AnimalNoncom	**3**	0.73	91%
Fishing	**1**	0.73	79%
PlantNoncom	**1**	0.46	76%
Trespassing	**19**	0.20	100%
Total	**25**	2.28	

Table 11: Base Rate Comparison: Hits per Month

work with fewer data points. SVMs, also able to express non-linear relationships, appear to fail due to their complexity and attempt to define very fine-grained divisions of the dataset. (2) Model interpretability is a necessity when working in the real-world. Our decision tree model was deployed because, not only did it have superior performance to CAPTURE, but it was also easy to directly look at the rules the decision tree had learned and evaluate whether or not those rules were reasonable (according to a domain expert). Thus, (3) the tradeoff between interpretability and performance, studied in domains where interpretability is key (e.g., biopharmaceutical classification) [13], may not always exist. Indeed, the most interpretable model, out of all that we evaluated, was also the best performing (by a large margin!); future research should (i) not always forego interpretability in favor of performance under the assumption that there is always a tradeoff but (ii) instead be sure to investigate simpler, interpretable models in case there isn't a tradeoff.

10. CONCLUSION

In this paper, we present INTERCEPT, a paradigm shift from complex logit-based models to simpler decision tree-based models. While the previous state-of-the-art, CAPTURE, represented the latest in a long line of behavioral game theory research, it suffers from poor performance and other critical limitations that preclude its actual deployment in the field. Indeed, in the process of conducting the most extensive empirical evaluation in the AI literature of one of the largest poaching datasets, we show a surprising result: INTERCEPT, based on a simpler model, significantly outperformed the more complex CAPTURE model. Furthermore, decision trees were specifically chosen due to the fundamental requirement of interpretability - a key limitation of previous logit-based models such as CAPTURE. Additionally, as a first for behavior modeling applications applied to this domain, we presented results from a month-long test of our model by rangers in QENP where rangers found and confiscated an active snare and almost a dozen additional snares, including multiple elephant snares, before they were deployed. Given that the rangers also found a poached elephant, their finding and confiscating of new elephant snares before they were deployed is significant; this research has potentially saved the lives of multiple elephants and other animals in QENP. Rangers in QENP are continuing patrols based on INTERCEPT, and we plan on incorporating INTERCEPT into PAWS' automated patrol generation framework [11, 10] to further augment our efforts in assisting conservation agencies to save endangered animals.

Acknowledgments: This research was supported by MURI grant W911NF-11-1-0332 and NSF grant with Cornell University 72954-10598. We are grateful to the Wildlife Conservation Society and the Uganda Wildlife Authority for supporting data collection in Queen Elizabeth National Park. We thank all the rangers and wardens in Queen Elizabeth National Park for their contributions in collecting and providing patrolling data in SMART.

REFERENCES

[1] Y. D. Abbasi, M. Short, A. Sinha, N. Sintov, C. Zhang, and M. Tambe. Human adversaries in opportunistic crime security games: Evaluating competing bounded rationality models. In *Third Annual Conference on Advances in Cognitive Systems ACS*, page 2, 2015.

[2] N. Basilico and N. Gatti. Strategic guard placement for optimal response toalarms in security games. In *International Conference on Autonomous Agents and Multiagent systems*, 2014.

[3] M. Brown, W. B. Haskell, and M. Tambe. Addressing scalability and robustness in security games with multiple boundedly rational adversaries. In *Conference on Decision and Game Theory for Security (GameSec)*, 2014.

[4] M. Costa-Gomes, V. P. Crawford, and B. Broseta. Cognition and behavior in normal-form games: An experimental study. *Econometrica*, 69(5), 2001.

[5] R. Critchlow, A. Plumptre, B. Andira, M. Nsubuga, M. Driciru, A. Rwetsiba, F. Wanyama, and C. Beale. Improving law enforcement effectiveness and efficiency in protected areas using ranger-collected monitoring data. *Conservation Letters*, 2016.

[6] R. Critchlow, A. Plumptre, M. Driciru, A. Rwetsiba, E. Stokes, C. Tumwesigye, F. Wanyama, and C. Beale. Spatiotemporal trends of illegal activities from ranger-collected data in a ugandan national park. *Conservation Biology*, 29(5):1458–1470, 2015.

[7] J. Davis and M. Goadrich. The relationship between precision-recall and roc curves. In *Proceedings of the 23rd International Conference on Machine Learning*, ICML, 2006.

[8] F. M. Delle Fave, A. X. Jiang, Z. Yin, C. Zhang, M. Tambe, S. Kraus, and J. P. Sullivan. Game-theoretic patrolling with dynamic execution uncertainty and a case study on a real transit system. *Journal of Artificial Intelligence Research*, 50:321–367, 2014.

[9] J. Eck, S. Chainey, J. Cameron, and R. Wilson. Mapping crime: Understanding hotspots. 2005.

[10] F. Fang, T. H. Nguyen, R. Pickles, W. Y. Lam, G. R. Clements, B. An, A. Singh, M. Tambe, and A. Lemieux. Deploying paws: Field optimization of the protection assistant for wildlife security. In *Innovative Applications of Artificial Intelligence Conference*, 2016.

[11] F. Fang, P. Stone, and M. Tambe. When security games go green: Designing defender strategies to prevent poaching and illegal fishing. In *International Joint Conference on Artificial Intelligence*, 2015.

[12] W. Haskell, D. Kar, F. Fang, M. Tambe, S. Cheung, and E. Denicola. Robust protection of fisheries with compass. In *Innovative Applications of Artificial Intelligence (IAAI)*, 2014.

[13] U. Johansson, C. Sönströd, U. Norinder, and H. Boström. Trade-off between accuracy and interpretability for predictive in silico modeling. *Future medicinal chemistry*, 3(6):647–663, 2011.

[14] M. Kanevski, A. Pozdnoukhov, and V. Timonin. Machine learning algorithms for geospatial data. applications and software tools. In *4th Biennial Meeting of the International Environmental Modelling and Software Society*, pages 7–10, 2008.

[15] D. Kar, F. Fang, F. D. Fave, N. Sintov, and M. Tambe. "a game of thrones": When human behavior models compete in repeated stackelberg security games. In *International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, 2015.

[16] C. Kiekintveld, T. Islam, and V. Kreinovich. Security games with interval uncertainty. In *International Conference on Autonomous Agents and Multiagent systems*, 2013.

[17] D. Korzhyk, V. Conitzer, and R. Parr. Solving stackelberg games with uncertain observability. In *International Conference on Autonomous Agents and Multiagent Systems*, 2011.

[18] W. S. Lee and B. Liu. Learning with positive and unlabeled examples using weighted logistic regression. In *ICML*, volume 3, 2003.

[19] D. L. Leottau, J. Ruiz-del Solar, P. MacAlpine, and P. Stone. A study of layered learning strategies applied to individual behaviors in robot soccer. In *Robot Soccer World Cup*, pages 290–302. Springer International Publishing, 2015.

[20] S. Mc Carthy, M. Tambe, C. Kiekintveld, M. L. Gore, and A. Killion. Preventing illegal logging: Simultaneous optimization of resource teams and tactics for security. In *AAAI*, 2016.

[21] D. McFadden. Conditional logit analysis of qualitative choice behavior. 1973.

[22] E. Munoz de Cote, R. Stranders, N. Basilico, N. Gatti, and N. Jennings. Introducing alarms in adversarial patrolling games. In *International Conference on Autonomous agents and Multiagent systems*, 2013.

[23] T. H. Nguyen, F. M. Delle Fave, D. Kar, A. S. Lakshminarayanan, A. Yadav, M. Tambe, N. Agmon, A. J. Plumptre, M. Driciru, F. Wanyama, et al. Making the most of our regrets: Regret-based solutions to handle payoff uncertainty and elicitation in green security games. In *International Conference on Decision and Game Theory for Security*, pages 170–191. Springer, 2015.

[24] T. H. Nguyen, A. Sinha, S. Gholami, A. Plumptre, L. Joppa, M. Tambe, M. Driciru, F. Wanyama, A. Rwetsiba, R. Critchlow, et al. Capture: A new predictive anti-poaching tool for wildlife protection. In *International Conference on Autonomous Agents & Multiagent Systems*, 2016.

[25] T. H. Nguyen, R. Yang, A. Azaria, S. Kraus, and M. Tambe. Analyzing the effectiveness of adversary modeling in security games. In *AAAI*, 2013.

[26] T. R. Palfrey and R. McKelvey. Quantal response equilibria in normal form games. *Games and Economic Behavior (special issue on Experimental Game Theory)*, 10:6, 1995.

[27] N. Park, E. Serra, T. Snitch, and V. Subrahmanian. Ape: A data-driven, behavioral model-based anti-poaching engine. *IEEE Transactions on Computational Social Systems*, 2(2):15–37, 2015.

[28] N. Park, E. Serra, and V. Subrahmanian. Saving rhinos with predictive analytics. *IEEE Intelligent Systems*, 30(4), 2015.

[29] Phys.org. More tigers poached so far this year than in 2015: census. Web, April 2016. `http://phys.org/news/2016-04-tigers-poached-year-census.html`.

[30] E. Shieh, B. An, R. Yang, M. Tambe, C. Baldwin, J. DiRenzo, B. Maule, and G. Meyer. Protect: A deployed game theoretic system to protect the ports of the united states. In *International Conference on Autonomous Agents and Multiagent Systems*, 2012.

[31] D. Stahl and P. Wilson. Experimental evidence on players' models of other players. *Journal of Economic Behavior & Organization*, 25(3), 1994.

[32] G. Sukthankar, R. Goldman, C. Geib, D. Pynadath, and H. Bui, editors. *Plan, Activity, and Intent Recognition*. Elsevier, feb 2014.

[33] Traffic.org. South africa reports small decrease in rhino poaching, but africa-wide 2015 the worst on record. Web, January 2016. `http://www.traffic.org/home/2016/1/21/south-africa-reports-small-decrease-in-rhino-poaching-but-af.html`.

[34] J. Wright and K. Leyton-Brown. Beyond equilibrium: Predicting human behavior in normal-form games, 2010.

[35] J. R. Wright and K. Leyton-Brown. Behavioral game theoretic models: A bayesian framework for parameter analysis. In *International Conference on Autonomous Agents and Multiagent Systems*, 2012.

[36] J. R. Wright and K. Leyton-Brown. Level-0 meta-models for predicting human behavior in games. In *Proceedings of the Fifteenth ACM Conference on Economics and Computation*, EC, 2014.

[37] R. Yang, B. Ford, M. Tambe, and A. Lemieux. Adaptive resource allocation for wildlife protection against illegal poachers. In *Iinternational conference on Autonomous Agents and Multiagent Systems*, 2014.

[38] R. Yang, C. Kiekintveld, F. Ordonez, M. Tambe, and R. John. Improving resource allocation strategy against human adversaries in security games. In *International Joint Conference on Artificial Intelligence*, 2011.

[39] C. Zhang, V. Bucarey, A. Mukhopadhyay, A. Sinha, Y. Qian, Y. Vorobeychik, and M. Tambe. Using abstractions to solve opportunistic crime security games at scale. In *International Conference on Autonomous Agents and Multiagent Systems*, 2016.

[40] C. Zhang, A. X. Jiang, M. B. Short, P. J. Brantingham, and M. Tambe. Defending against opportunistic criminals: New game-theoretic frameworks and algorithms. In *International Conference on Decision and Game Theory for Security*, 2014.

[41] C. Zhang, A. Sinha, and M. Tambe. Keeping pace with criminals: Designing patrol allocation against adaptive opportunistic criminals. In *International Conference on Autonomous Agents and Multiagent systems*, 2015.