Adversarial Sequence Tagging

Jia Li,* Kaiser Asif,* Hong Wang, Brian D. Ziebart, Tanya Berger-Wolf
Department of Computer Science, University of Illinois at Chicago, Chicago, IL
{jli213, kasif2,hwang207, bziebart, tanyabw}@uic.edu

Abstract

Providing sequence tagging that minimize Hamming loss is a challenging, but important, task. Directly minimizing this loss over a training sample is generally an NP-hard problem. Instead, existing sequence tagging methods minimize a convex upper bound that upper bounds the Hamming loss. Unfortunately, this often either leads to inconsistent predictors (e.g., max-margin methods) or predictions that are mismatched on the Hamming loss (e.g., conditional random fields). We present *adversarial sequence tagging*, a consistent structured prediction framework for minimizing Hamming loss by pessimistically viewing uncertainty. Our approach pessimistically approximates the training data, yielding an adversarial game between the sequence tag predictor and the sequence labeler. We demonstrate the benefits of the approach on activity recognition and information extraction/segmentation tasks.

1 Introduction

Sequence tagging methods that jointly predict interdependent variables are needed in applications ranging from natural language processing [Lafferty *et al.*, 2001; Sha and Pereira, 2003] to activity recognition [Vail *et al.*, 2007; Liao *et al.*, 2007]. Unfortunately, obtaining a parametric predictor that directly minimizes the Hamming loss (the number of incorrectly predicted variables) is an NP-hard empirical risk minimization (ERM) problem [Hoffgen *et al.*, 1995] in general. Conditional random fields [Lafferty *et al.*, 2001] and maximum margin methods (e.g., structural support vector machines [Joachims *et al.*, 2009] and maximum margin Markov networks [Taskar *et al.*, 2004]) instead minimize convex surrogates of the Hamming loss (i.e., the logarithmic loss and the hinge loss). This mismatch between the surrogate loss function and the Hamming loss leads to inconsistency and sub-optimal predictive performance.

We present adversarial sequence tagging (AST), a supervised sequence tagging approach that is both consistent for the Hamming loss and provides good predictive performance in practice by adversarially approximating the training data. At its core, our approach reduces prediction to solving a zero-sum game based on the Hamming loss between a prediction player trying to minimize the loss and an adversarial player trying to maximize it while being constrained to reflect properties of the training data. Parameter estimation is solved as a convex optimization problem under this formulation even though minimizing the Hamming loss is non-convex in ERM formulations. Our contributions in this paper are:

1. We extend adversarial loss minimization methods for classification [Asif *et al.*, 2015] and multivariate performance measures [Wang *et al.*, 2015] to the structured prediction setting of sequence tagging.
2. We establish the Fisher consistency of our adversarial prediction method and contrast it with the inconsistency of maximum margin methods for sequence prediction.
3. We scale our approach to long sequences of variables with many possible values by leveraging an independence property that allows a single oracle inference method (in contrast, double oracle is exclusively required for multivariate losses [Wang *et al.*, 2015]).
4. We evaluate our approach on natural language processing and activity recognition tasks, demonstrating its competitive predictive performance compared with maximum margin methods and CRFs.

2 Background and Related Work

2.1 Notation

In this work, we seek a sequence predictor, $\hat{P}(\mathbf{y}|\mathbf{x})$, for variables $\mathbf{y} = \mathbf{y}_{1:T} = \{y_1, y_2, \ldots, y_T\} \in \boldsymbol{\mathcal{Y}}$, conditioned on provided input variables, $\mathbf{x} = \mathbf{x}_{1:T} = \{x_1, x_2, \ldots, x_T\} \in \boldsymbol{\mathcal{X}}$. We consider the supervised learning setting where m sequence examples, $\{\mathbf{y}^{(j)}, \mathbf{x}^{(j)}\}_{j=1:m}$ drawn from empirical training distribution $\tilde{P}(\mathbf{x}, \mathbf{y})$ (samples from true distribution $P(\mathbf{y}, \mathbf{x})$), are available to estimate the model. We distinguish between the actual label variables, $\mathbf{y}$, and the predicted label variables, $\hat{\mathbf{y}}$, using "hat" notation, and will later introduce a set of adversarially-chosen labels $\check{\mathbf{y}} = \{\check{y}_1, \check{y}_2, \ldots, \check{y}_T\}$. We make extensive use of expectation notation, $\mathbb{E}_{P(x)}[f(X)] = \sum_{x \in \mathcal{X}} P(x) f(x)$, in which random variables are capitalized. We also denote statistics of the sequence of variables as

[*] Both authors contributed equally.

$\Phi(\mathbf{x}, \mathbf{y}) \in \mathbb{R}^k$. These typically decompose additively over the sequence: e.g., $\Phi(\mathbf{x}, \mathbf{y}) = \sum_{t=1}^{T-1} \phi(\mathbf{x}, \mathbf{y}_{t:t+1})$.

2.2 Empirical Sequence Risk Minimization

Conditional random fields (CRFs) and structured support vector machines (SSVMs) are two prominent methods for sequence tagging based on minimizing the empirical risk:

$$\underset{\theta}{\operatorname{argmin}} \mathbb{E}_{\tilde{P}(\mathbf{x},\mathbf{y})\hat{P}_\theta(\hat{\mathbf{y}}|\mathbf{x})} \left[\text{loss}\left(\mathbf{Y}, \hat{P}_\theta(\cdot|\mathbf{x})\right)\right] + \lambda||\theta|| \quad (1)$$

$$\text{or} \underset{\theta}{\operatorname{argmin}} \mathbb{E}_{\tilde{P}(\mathbf{x},\mathbf{y})} \left[\text{loss}\left(\mathbf{Y}, \hat{f}_\theta(\mathbf{X})\right)\right] + \lambda||\theta||. \quad (2)$$

For conditional random fields [Lafferty *et al.*, 2001], the logarithmic loss, $-\log \hat{P}(\mathbf{y}|\mathbf{x})$, and an exponential random field model, e.g., $\hat{P}(\mathbf{y}|\mathbf{x}) \propto \exp(\theta \cdot \Phi(\mathbf{x}, \mathbf{y}))$ are employed in Eq. (1). For structured support vector machines [Tsochantaridis *et al.*, 2004], the structured hinge loss is a convex approximation to the Hamming loss, $\Delta(\mathbf{y}, \hat{\mathbf{y}}) = \sum_{t=1}^{T} I(\hat{y}_t \neq \tilde{y}_t)$,

$$\left[\max_{\mathbf{y}' \neq \mathbf{y}} \Delta(\mathbf{y}, \mathbf{y}') + \theta \cdot (\Phi(\mathbf{x}, \mathbf{y}') - \Phi(\mathbf{x}, \mathbf{y}))\right]_+, \quad (3)$$

where $[f(x)]_+ \triangleq \max(0, f(x))$, and a linear discriminant function, $\hat{f}_\theta(\mathbf{x}) = \operatorname{argmax}_{\hat{\mathbf{y}} \in \boldsymbol{\mathcal{Y}}} \theta \cdot \Phi(\mathbf{x}, \hat{\mathbf{y}})$, are employed in Eq. (2). The loss function of each model is a convex upper bound on the Hamming loss, $\sum_{t=1}^{T} I(\hat{y}_t \neq \tilde{y}_t)$.

2.3 Sequence Tagging Consistency

Predictors that minimize a loss measure when provided with the true data distribution for training are desirable. Definition 1 formalizes this notion in terms of the Fisher consistency.

Definition 1. *Predictor $\hat{f}(\mathbf{x})$ with full representational ability (e.g., parameterized by potential functions $\psi(\mathbf{x}, \mathbf{y})$) is Fisher consistent for loss function $\Delta(\hat{\mathbf{f}}(\mathbf{x}), \mathbf{y})$ if it minimizes the expected loss, $\mathbb{E}_{P(\mathbf{x},\mathbf{y})}[\Delta(\hat{f}(\mathbf{X}), \mathbf{Y})]$, when trained (e.g., surrogate ERM) under the true data distribution $P(\mathbf{x}, \mathbf{y})$.*

Similar to multi-class SVM incosistency [Liu, 2007], Theorem 1 shows SSVM's inconsistency for sequence tagging. This inconsistency motivates our desires for a better method.

Theorem 1. *Given the distribution $P_{11} = 0.4$, $P_{22} = 0.3$, $P_{33} = 0.3$, and $P_{ij} = 0 \; \forall i \neq j$ over sequences of length two, where P_{ij} compactly denotes $P(y_1 = i, y_2 = j|\mathbf{x})$, the hinge loss of SSVM is not Fisher consistent for the Hamming loss.*

Proof. For SSVM and Hamming loss, ψ^* minimizes:

$$\mathbb{E}\left[\sum_{\mathbf{y} \in \boldsymbol{\mathcal{Y}}} P_{\mathbf{y}} \left[\max_{\mathbf{y}' \neq \mathbf{y}} \Delta(\mathbf{y}, \mathbf{y}') + \psi(\mathbf{x}, \mathbf{y}') - \psi(\mathbf{x}, \mathbf{y})\right]_+\right]. \quad (4)$$

The minimizer ψ^* must satisfy $\psi(\mathbf{x}, 11) \geq \psi(\mathbf{x}, \mathbf{y}')$ where $\mathbf{y}' \neq 11$. Otherwise, the result will not be Fisher consistent. Assume (w.l.o.g.) $\psi(\mathbf{x}, 22) \geq \psi(\mathbf{x}, 33)$. (4) becomes $P_{11} [\max_{\mathbf{y}' \neq 11}\{\Delta(11, \mathbf{y}') + \psi(\mathbf{x}, \mathbf{y}') - \psi(\mathbf{x}, 11)\}]_+ + 0 + 0 + 0 + P_{22} [\max_{\mathbf{y}' \neq 22}\{\Delta(22, \mathbf{y}') + \psi(\mathbf{x}, \mathbf{y}') - \psi(\mathbf{x}, 22)\}]_+ + 0 + 0 + 0 + P_{33}[\max_{\mathbf{y}' \neq 33}\{\Delta(33, \mathbf{y}') + \psi(\mathbf{x}, \mathbf{y}') - \psi(\mathbf{x}, 33)\}]_+$. Since $\psi(\mathbf{x}, 11) \geq \psi(\mathbf{x}, \mathbf{y}')$, $\psi(\mathbf{x}, 11) - \psi(\mathbf{x}, 22) > \psi(\mathbf{x}, \mathbf{y}') - \psi(\mathbf{x}, 22)$, for $\mathbf{y}' \neq 11$, 2 is the maximum Hamming loss for length 2 sequences. As a result, $[\max_{\mathbf{y}' \neq 22}\{\Delta(22, \mathbf{y}') + \psi(\mathbf{x}, \mathbf{y}') - \psi(\mathbf{x}, 22)\}]_+ = 2 + \psi(\mathbf{x}, 11) - \psi(\mathbf{x}, 22)$. Similarity, $[\max_{\mathbf{y}' \neq 33}\{\Delta(33, \mathbf{y}') + \psi(\mathbf{x}, \mathbf{y}') - \psi(\mathbf{x}, 33)\}]_+ = 2 + \psi(\mathbf{x}, 11) - \psi(\mathbf{x}, 33)$. The expected hinge loss is $P_{11}[\max_{\mathbf{y}' \neq 11}\{\Delta(11, \mathbf{y}') + \psi(\mathbf{x}, \mathbf{y}') - \psi(\mathbf{x}, 11)\}]_+ + P_{22}(2 + \psi(\mathbf{x}, 11) - \psi(\mathbf{x}, 22)) + P_{33}(2 + \psi(\mathbf{x}, 11) - \psi(\mathbf{x}, 33))$. When $\psi(\mathbf{x}, 11) = \psi(\mathbf{x}, 22) = \psi(\mathbf{x}, 33) \geq \psi(\mathbf{x}, ij), \forall i \neq j$, the loss is 2. For any other case $[\max_{\mathbf{y}' \neq 11}\{\Delta(11, \mathbf{y}') + \psi(\mathbf{x}, \mathbf{y}') - \psi(\mathbf{x}, 11)\}]_+$, we show the loss exceeds 2:

Case 1: when $\max_{\mathbf{y}' \neq 11}\{\Delta(11, \mathbf{y}') + \psi(\mathbf{x}, \mathbf{y}') - \psi(\mathbf{x}, 11)\} \leq 0$, then $2 + \psi(\mathbf{x}, 22) - \psi(\mathbf{x}, 11) < 0$, $\psi(\mathbf{x}, 11) - \psi(\mathbf{x}, 22) \geq 2$. Similarly, $\psi(\mathbf{x}, 11) - \psi(\mathbf{x}, 33) \geq 2$. The loss is $P_{22}(2 + \psi(\mathbf{x}, 11) - \psi(\mathbf{x}, 22)) + P_{33}(2 + \psi(\mathbf{x}, 11) - \psi(\mathbf{x}, 33)) \geq 4(P_{22} + P_{33})$. As long as $P_{22} + P_{33} > 0.5$, this is greater than 2.

Case 2: when $\max_{\mathbf{y}' \neq 11}\{\Delta(11, \mathbf{y}') + \psi(\mathbf{x}, \mathbf{y}' - \psi(\mathbf{x}, 11)\} > 0$, we have $\max_{\mathbf{y}' \neq 11} \{\Delta(11, \mathbf{y}') + \psi(\mathbf{x}, \mathbf{y}') - \psi(\mathbf{x}, 11)\} \geq 2 + \psi(\mathbf{x}, 22) - \psi(\mathbf{x}, 11)$. We also have $\psi(\mathbf{x}, 11) - \psi(\mathbf{x}, 33) \geq \psi(\mathbf{x}, 11) - \psi(\mathbf{x}, 22)$ The loss is at least $P_{11}(2 + \psi(\mathbf{x}, 22) - \psi(\mathbf{x}, 11)) + P_{22}(2 + \psi(\mathbf{x}, 11) - \psi(\mathbf{x}, 22)) + P_{33}(2 + \psi(\mathbf{x}, 11) - \psi(\mathbf{x}, 33)) \geq 2(P_{11} + P_{22} + P_{33}) + P_{11}(\psi(\mathbf{x}, 22) - \psi(\mathbf{x}, 11)) + P_{22}(\psi(\mathbf{x}, 11) - \psi(\mathbf{x}, 22)) + P_{33}(\psi(\mathbf{x}, 11) - \psi(\mathbf{x}, 22))$. When $P_{22} + P_{33} > 0.5$, the loss exceeds 2. We can see that in the example, the minimized value for the loss function is 2 and is achieved when $\psi(\mathbf{x}, 11) = \psi(\mathbf{x}, 22) = \psi(\mathbf{x}, 33)$. Since $argmax$ cannot distinguish between the different labels, SSVM is not Fisher consistent. □

2.4 Adversarial Estimation

We expand upon prior perspectives for prediction as an adversarial task [Dalvi *et al.*, 2004; Lowd and Meek, 2005; Biggio *et al.*, 2010]. However, unlike those works, we do not assume that the data comes from an adversary attempting to corrupt the test data to, e.g., defeat a spam filter. Instead, our approach is more closely related to the duality between worst-case minimization of information-theoretic loss functions and maximum likelihood estimation of exponential family member probability distributions [Topsøe, 1979; Grünwald and Dawid, 2004; Liu and Ziebart, 2014] and methods that parametrically constrain the adversary [Lanckriet *et al.*, 2003].

Our method follows two recent advances in adversarial classification: a general formulation of cost-sensitive classification as a zero-sum prediction game [Asif *et al.*, 2015]; and adversarial prediction games for multivariate performance measures [Wang *et al.*, 2015]. These previous methods for univariate predictions do not incorporate correlative relationships between predicted variables and cannot be effectively employed for sequence tagging tasks. We demonstrate how the adversarial formulation can be extended to the structured prediction setting using constraint generation methods known as the single and double oracle [McMahan *et al.*, 2003] to avoid exponentially-sized zero-sum games from the latter work [Wang *et al.*, 2015] in the sequence tagging setting. The key difference is that feature functions are multivariate in this work, while loss functions are multivariate in that prior work.

3 Adversarial Sequence Tagging Games

Motivated by the mismatch between convex surrogates and loss measures of interest, we develop our adversarial approach for sequence tagging.

3.1 Adversarial Formulation

Instead of choosing a predictor's parametric form and using ERM on training data to select its parameters, we obtain the predictor that performs best for the worst-case choice of conditional label distributions that match statistics measured from available training data. As we shall see, sequence loss functions for which empirical risk minimization is non-convex and NP-hard can often be solved efficiently in this formulation.

Following recently developed methods for adversarial cost-sensitive classification [Asif *et al.*, 2015], we pose structured prediction as an adversarial game in which an estimator player chooses a conditional distribution, $\hat{P}(\hat{\mathbf{y}}|\mathbf{x})$. An adversarial player then chooses a distribution, $\check{P}(\check{\mathbf{y}}|\mathbf{x})$, from the set of distributions matching certain statistics, $\Phi(\mathbf{x},\mathbf{y})$. The estimator player seeks to minimize an expected loss, while the adversary seeks to maximize this loss:

$$\min_{\hat{P}(\hat{\mathbf{y}}|\mathbf{x})} \max_{\check{P}(\check{\mathbf{y}}|\mathbf{x})} \mathbb{E}_{\tilde{P}(\mathbf{x})\check{P}(\check{\mathbf{y}}|\mathbf{x})\hat{P}(\hat{\mathbf{y}}|\mathbf{x})}[\text{loss}(\hat{\mathbf{Y}},\check{\mathbf{Y}})] \tag{5}$$

$$\text{such that: } \mathbb{E}_{\tilde{P}(\mathbf{x})\check{P}(\check{\mathbf{y}}|\mathbf{x})}[\Phi(\mathbf{X},\check{\mathbf{Y}})] = \mathbb{E}_{\tilde{P}(\mathbf{x},\mathbf{y})}[\Phi(\mathbf{X},\mathbf{Y})],$$

where the feature functions, $\Phi(\mathbf{x},\mathbf{y})$, typically additively decompose over pairs of the $Y_1,\ldots,Y_T$ variables: e.g., $\Phi(\mathbf{x},\mathbf{y}) = \sum_{t=1}^{T-1}\phi(\mathbf{x},y_t,y_{t+1})$.

By leveraging Lagrangian and zero-sum game duality, this formulation reduces to a convex optimization problem:

$$\min_{\theta} \mathbb{E}_{\tilde{P}(\tilde{\mathbf{x}},\tilde{\mathbf{y}})}\left[\max_{\check{\mathbf{p}}_{\mathbf{X}}}\min_{\hat{\mathbf{p}}_{\mathbf{X}}} \hat{\mathbf{p}}_{\mathbf{X}}^{\mathrm{T}}\mathbf{C}'_{\mathbf{X},\theta}\check{\mathbf{p}}_{\mathbf{X}}\right], \tag{6}$$

where $\hat{\mathbf{p}}_{\mathbf{x}}$ and $\check{\mathbf{p}}_{\mathbf{x}}$ are vector representations of the conditional label distributions, $P(\hat{\mathbf{y}}|\mathbf{x})$ and $P(\check{\mathbf{y}}|\mathbf{x})$ and $\mathbf{C}'_{x,\theta}$ is a payoff matrix that incorporates both the loss function and a Lagrangian potential term that enforces the optimization's constraints: $(\mathbf{C}'_{\mathbf{x},\mathbf{y},\theta})_{\hat{\mathbf{y}},\check{\mathbf{y}}} = \text{loss}(\hat{\mathbf{y}},\check{\mathbf{y}}) + \theta\cdot(\phi(\mathbf{x},\check{\mathbf{y}}) - \phi(\mathbf{x},\mathbf{y}))$. Table 1 is the payoff matrix of the 3-length binary-valued sequence game. Rows represent the predictor's pure strategies. Columns represent the adversary's pure strategies. Each payoff combines the Hamming loss (e.g., 1 for sequences 001 and 101) and a Lagrangian potential motivating the adversary to behave "similarly to" training data.

Zero-sum games can be solved as linear programs to find each player's mixed Nash equilibrium [von Neumann and Morgenstern, 1947]. For example, the mixed Nash equilibrium strategy for the adversarial player is obtained from:

$$\max_{\check{\mathbf{p}}\geq\mathbf{0},v} v \text{ such that: } v \leq \mathbf{C}'_{\hat{\mathbf{y}},*}\check{\mathbf{p}}\ \forall\hat{\mathbf{y}}\in\boldsymbol{\mathcal{Y}};\text{ and } \mathbf{1}^{\mathrm{T}}\check{\mathbf{p}} = 1. \tag{7}$$

Similarly, the predictor's mixed Nash equilibrium strategy is:

$$\min_{\hat{\mathbf{p}}\geq\mathbf{0},v} v \text{ such that: } v \geq \hat{\mathbf{p}}^{\mathrm{T}}\mathbf{C}'_{*,\check{y}}\ \forall\check{\mathbf{y}}\in\boldsymbol{\mathcal{Y}};\text{ and } \mathbf{1}^{\mathrm{T}}\hat{\mathbf{p}} = 1. \tag{8}$$

These sets of inequality constraints ensure the game value v is constrained by all possible pure strategies of the opponent.

Table 1: The payoff matrix $\mathbf{C}'_{x,\theta}$ for a game over the length three binary-valued chain of variables between player $\check{Y}$ choosing a distribution over columns and $\hat{Y}$ choosing a distribution over rows. Lagrangian potentials are compactly represented as: $\psi_{\check{y}_1\check{y}_2\check{y}_3} \triangleq \theta\cdot(\Phi(\check{\mathbf{y}},\mathbf{x}) - \Phi(\mathbf{y},\mathbf{x}))$.

	000	**001**	**010**	**011**	**100**	**101**	**110**	**111**
000	$0{+}\psi_{000}$	$1{+}\psi_{001}$	$1{+}\psi_{010}$	$2{+}\psi_{011}$	$1{+}\psi_{100}$	$2{+}\psi_{101}$	$2{+}\psi_{110}$	$3{+}\psi_{111}$
001	$1{+}\psi_{000}$	$0{+}\psi_{001}$	$2{+}\psi_{010}$	$1{+}\psi_{011}$	$2{+}\psi_{100}$	$1{+}\psi_{101}$	$3{+}\psi_{110}$	$2{+}\psi_{111}$
010	$1{+}\psi_{000}$	$2{+}\psi_{001}$	$0{+}\psi_{010}$	$1{+}\psi_{011}$	$2{+}\psi_{100}$	$3{+}\psi_{101}$	$1{+}\psi_{110}$	$2{+}\psi_{111}$
011	$2{+}\psi_{000}$	$1{+}\psi_{001}$	$1{+}\psi_{010}$	$0{+}\psi_{011}$	$3{+}\psi_{100}$	$2{+}\psi_{101}$	$2{+}\psi_{110}$	$1{+}\psi_{111}$
100	$1{+}\psi_{000}$	$2{+}\psi_{001}$	$2{+}\psi_{010}$	$3{+}\psi_{011}$	$0{+}\psi_{100}$	$1{+}\psi_{101}$	$1{+}\psi_{110}$	$2{+}\psi_{111}$
101	$2{+}\psi_{000}$	$1{+}\psi_{001}$	$3{+}\psi_{010}$	$2{+}\psi_{011}$	$1{+}\psi_{100}$	$0{+}\psi_{101}$	$2{+}\psi_{110}$	$1{+}\psi_{111}$
110	$2{+}\psi_{000}$	$3{+}\psi_{001}$	$1{+}\psi_{010}$	$2{+}\psi_{011}$	$1{+}\psi_{100}$	$2{+}\psi_{101}$	$0{+}\psi_{110}$	$1{+}\psi_{111}$
111	$3{+}\psi_{000}$	$2{+}\psi_{001}$	$2{+}\psi_{010}$	$1{+}\psi_{011}$	$2{+}\psi_{100}$	$1{+}\psi_{101}$	$1{+}\psi_{110}$	$0{+}\psi_{111}$

Extending adversarial classification [Asif *et al.*, 2015] to sequence tagging settings leads to inner zero-sum matrix games characterized by $\mathbf{C}'_{\mathbf{x},\theta}$ with $|\mathcal{Y}|^T$ value assignment "pure strategies" for each player. As a consequence, explicitly constructing the corresponding game matrix is intractable for all but the smallest of sequence tagging tasks.

3.2 Double Oracle Method for Efficient Prediction

We overcome the computational difficulties of constructing the entire adversarial game using the double oracle algorithm [McMahan *et al.*, 2003] to iteratively construct an appropriate reduced game that still provides the correct equilibrium. This approach was previously applied to obtain game solutions for multivariate performance measures [Wang *et al.*, 2015]. We extend this to structured prediction problems where consecutive variables are related by measured statistics.

The double oracle game solver considers a subset of pure strategies, $\hat{S}$ or $\check{S}$, for each player. It constructs the payoff matrix and obtains the mixed Nash equilibrium for this subset of pure strategies. It then finds the best response pure strategy, $\check{\mathbf{y}}_{BR}$ or $\hat{\mathbf{y}}_{BR}$, for the player in response to the opponent's equilibrium mixed strategy, $\hat{P}(\hat{\mathbf{y}}|\mathbf{x})$ or $\check{P}(\check{\mathbf{y}}|\mathbf{x})$, and adds it to the set of pure strategies. The algorithm terminates when neither player can improve upon their strategy with additional actions. Thus, the strategies it returns are a Nash equilibrium pair [McMahan *et al.*, 2003]. We refer interested readers to Wang et. al [Wang *et al.*, 2015] for more details. The major difference for sequence tagging from that previous work is in finding best responses. We find the best response $\check{\mathbf{y}}_{BR}$ pure strategy to add to the game according to the maximization of:

$$\begin{aligned}
&\max_{\check{\mathbf{y}}_{1:T}} \mathbb{E}_{\hat{P}(\hat{\mathbf{y}}_{1:T}|\mathbf{x})}\left[\sum_{t=1}^{T} I(\hat{Y}_t \neq \check{y}_t)\right] + \sum_{t=1}^{T-1}\theta\cdot\phi(\mathbf{x},\check{\mathbf{y}}_{t:t+1}) \\
&= \max_{\check{y}_1}\Bigg(\mathbb{E}_{\hat{P}(\hat{y}_1|\mathbf{x})}\left[I(\hat{Y}_1\neq\check{y}_1)\right] + \max_{\check{y}_2}\Bigg(\theta\cdot\phi(\mathbf{x},\check{\mathbf{y}}_{1:2}) \\
&\quad + \mathbb{E}_{\hat{P}(\hat{y}_2|\mathbf{x})}\left[I(\hat{Y}_2\neq\check{y}_2)\right] + \max_{\check{y}_3}\Bigg(\theta\cdot\phi(\mathbf{x},\check{\mathbf{y}}_{2:3}) + \ \ldots \\
&\quad + \max_{\check{y}_T}\theta\cdot\phi(\mathbf{x},\check{\mathbf{y}}_{T-1:T}) + \mathbb{E}_{\hat{P}(\hat{y}_T|\mathbf{x})}\left[I(\hat{Y}_T\neq\check{y}_T)\right]\Bigg)\Bigg)\Bigg),
\end{aligned} \tag{9}$$

which is recursively defined from marginal probabilities $\hat{P}(\hat{y}_i)$ as shown, and solved using the Viterbi algorithm

[Viterbi, 1967] by iteratively computing for $t = \{T, \cdots, 1\}$: $\beta(\check{y}_t) = \mathbb{E}_{\hat{P}(\hat{y}_t)}\left[I(\hat{Y}_t \neq \check{y}_t)\right] + \max_{\check{y}_{t+1}} \theta \cdot \phi(\mathbf{x}, \check{\mathbf{y}}_{t:t+1}) + \beta(\check{y}_{t+1})$ and storing the maximizing variable assignment.

For the complementary problem of adding the best $\hat{\mathbf{y}}$ action to the game, we choose $\hat{\mathbf{y}}$ according to: $\operatorname{argmin}_{\hat{\mathbf{y}}} \mathbb{E}_{\check{p}(\check{\mathbf{y}}|\mathbf{x})}[\text{loss}(\hat{\mathbf{y}}, \check{\mathbf{y}})]$. For the Hamming loss, each term of the best response sequence can be independently obtained from $\hat{\mathbf{y}}_{\text{BR}} = \{\operatorname{argmax}_{y_t} \check{P}(y_t|\mathbf{x})\}_{t=1}^{T}$.

3.3 Single Oracle Method for Efficient Prediction

So long as the loss function additively decomposes into payoff matrix terms $\mathbf{C}_t$ for each $t \in \{1, \dots, T\}$, the estimator's predictions are independent (as observed when finding the best response for the double oracle method). This is in contrast with adversarial prediction methods for structured losses [Wang *et al.*, 2015], in which the loss function prevents independence from both adversary and predictor. The independence found in the sequence tagging game allows the combination of all estimator's "pure strategies" in the sequence tagging game to be efficiently considered using the following pair of linear programs:

$$(1) \min_{\hat{\mathbf{p}}_1, \hat{\mathbf{p}}_2, \dots, \hat{\mathbf{p}}_T, v} v \text{ such that: } \hat{\mathbf{p}}_t \geq \mathbf{0} \text{ and } \mathbf{1}^{\mathrm{T}}\hat{\mathbf{p}}_t = 1, \forall t;$$

$$\text{and } v \geq \theta^{\mathrm{T}}\phi(\mathbf{x}, \check{y}) + \sum_{t=1}^{T} \hat{\mathbf{p}}_t^{\mathrm{T}} [\mathbf{C}_t]_{*,\check{y}}\ \check{y} \in \check{S}; \tag{10}$$

$$(2) \max_{\check{\mathbf{p}} \geq \mathbf{0}, v_1, v_2, \dots, v_T} \theta^{\mathrm{T}} \mathbf{\Phi}_{\mathbf{x},\mathbf{y}} \check{\mathbf{p}} + \sum_{t=1}^{T} v_t \text{ such that: } \mathbf{1}^{\mathrm{T}}\check{\mathbf{p}} = 1$$

$$\text{and } v_t \leq [\mathbf{C}_t]_{\hat{y},*}\ \check{\mathbf{p}}\ \forall t, \hat{y} \in \mathcal{Y}; \tag{11}$$

As the entire set of predictor pure strategies is considered by this revised set of linear programs, the double oracle method can be reduced to a single oracle method. Only the adversary's set of pure strategies over the entire sequence needs to be expanded in this approach (Algorithm 1).

Algorithm 1 Single Oracle Game Solver

```
Input: Lagrangian potential, $\boldsymbol{\psi}$; initial action set $\check{S}$
Output: $[\hat{P}(\hat{\mathbf{y}}|\mathbf{x}), \check{P}(\check{\mathbf{y}}|\mathbf{x})]$
  $\check{\mathbf{y}}_{BR} \leftarrow \{\}$
  repeat
    $\boldsymbol{C}_t \leftarrow$ buildPayoffMatrices($\check{S}$, $\psi$)
    $[\hat{P}(\hat{\mathbf{y}}|\mathbf{x}), v_{\text{Nash}_1}] \leftarrow$ solveZeroSumGame$_{\check{Y}}(\boldsymbol{C})$
    $[\check{\mathbf{y}}_{BR}, \check{v}_{\text{BR}}] \leftarrow$ findBestResponseStrategy($\hat{P}(\hat{\mathbf{y}}|\mathbf{x})$, $\psi$)
    $\check{S} \leftarrow \check{S} \cup \check{\mathbf{y}}_{BR}$
  until $(v_{\text{Nash}_1} = \check{v}_{\text{BR}})$
  return $[\hat{P}(\hat{\mathbf{y}}|\mathbf{x}), \check{P}(\check{\mathbf{y}}|\mathbf{x})]$
```

The size of the payoff matrix, C' from Eq. (7), in the double oracle method is $\mathcal{O}(|\hat{S}||\check{S}|)$, while the single oracle method corresponds to a matrix of size $\mathcal{O}(|\check{S}||T||\mathcal{Y}|)$. Computational benefits may thus be realized by this approach when the number of estimator pure strategies in the double oracle method is sufficiently large. In such cases, reducing the overall size of the payoff matrix compensates for the added complexity of the linear program in the single oracle method. In practice, a hybrid approach that switches between single oracle and double oracle methods based on the length of the sequence can be used to yield faster predictions.

Algorithm 2 Parameter Estimation Algorithm

```
Input: Training dataset $\mathcal{D}$ with pairs $(\mathbf{x}, \mathbf{y}) \in \mathcal{D}$, feature
    function $\Phi : \boldsymbol{\mathcal{X}} \cup \boldsymbol{\mathcal{Y}} \rightarrow \mathbb{R}^k$, learning rate $\{\gamma_t\}$
Output: Model parameter estimate $\theta$
  $t \leftarrow 1$
  while $\theta$ not converged do
    Random shuffle samples for stochastic training
    for all $(\mathbf{x}, \tilde{\mathbf{y}}) \in \mathcal{D}$ do
      Compute $\check{P}(\check{\mathbf{y}}|\mathbf{x})$ using single/double oracle
      $\nabla_\theta \leftarrow \mathbb{E}_{\check{\mathbf{P}}(\mathbf{y}|\mathbf{x})}[\mathbf{\Phi}(\mathbf{x}, \check{\mathbf{y}})] - \mathbf{\Phi}(\mathbf{x}, \tilde{\mathbf{y}})$
      $\theta \leftarrow \theta - \gamma_t \nabla_\theta$;    $t \leftarrow t + 1$
    end for
  end while
```

3.4 Learning via Convex Optimization

We employ stochastic gradient descent to obtain the AST model parameters. As described in Algorithm 2, for each iteration in the update, we use single oracle (Algorithm 1) or double oracle to find the adversary's Nash equilibrium solution to the AST game: $\check{P}(\check{\mathbf{y}}|\mathbf{x})$. Feature expectations are calculated according to Eq. (12):

$$\mathbb{E}_{\check{\mathbf{P}}(\check{\mathbf{y}}|\mathbf{x})}\left[\mathbf{\Phi}(\mathbf{x}, \check{\mathbf{Y}})\right] = \mathbb{E}_{\check{\mathbf{P}}(\check{\mathbf{y}}|\mathbf{x})}\left[\sum_{t=1}^{T-1} \phi(\mathbf{x}, \check{y}_t, \check{y}_{t+1})\right] \tag{12}$$

$$= \sum_{t=1}^{T-1} \sum_{y, y'} \check{P}(\check{Y}_t = y', \check{Y}_{t+1} = y|\mathbf{x}, \theta)\phi(\mathbf{x}, \check{y}_t, \check{y}_{t+1}).$$

This feature expectation under the adversary's distribution is then used to calculate the gradient, as shown in Algorithm 2. Due to convexity, this optimization procedure converges to a global optima given appropriate learning rate parameters γ_t.

3.5 Consistency

An important benefit of AST over maximum margin methods is the consistency guarantee it provides.

Theorem 2. *Given that the sequence's probability distribution factors according to the chain independence assumptions:* $P(\mathbf{y}|\mathbf{x}) = \prod_{t=1}^{T} P(y_t|y_{t-1}, \mathbf{x}_{1:T})$*, and an arbitrarily rich feature representation,* $\psi(y_t, y_{t+1}, \mathbf{x}_{1:T})$*, the AST method provides the loss-optimal sequence tagging,* $\operatorname{argmin}_{\hat{\mathbf{y}}} \mathbb{E}_{P(\mathbf{Y}|\mathbf{x})}[\textit{loss}(\hat{\mathbf{y}}, \mathbf{Y})]$.

Proof. The Lagrangian of Eq. (6) gives, equivalently:

$$\min_{\psi(\cdot,\cdot)} \max_{\check{P}(\check{\mathbf{y}}|\mathbf{x})} \min_{\hat{P}(\hat{\mathbf{y}}|\mathbf{x})} \mathbb{E}_{P(\mathbf{x},\mathbf{y})}\Big[\mathbb{E}_{\hat{P}(\hat{\mathbf{y}}|\mathbf{x})\check{P}(\check{\mathbf{y}}|\mathbf{x})}\Big[\text{loss}(\hat{\mathbf{Y}}, \check{\mathbf{Y}}) + \psi(\mathbf{X}, \check{\mathbf{Y}}) - \psi(\mathbf{X}, \mathbf{Y})\Big|\mathbf{X}\Big]\Big] \tag{13}$$

$$\overset{(a)}{=} \max_{\check{P}(\check{\mathbf{y}}|\mathbf{x})} \min_{\psi(\cdot,\cdot)} \Bigg(\mathbb{E}_{P(\mathbf{x},\mathbf{y})\check{P}(\check{\mathbf{y}}|\mathbf{x})} \Big[\psi(\mathbf{X},\hat{\mathbf{Y}}) - \psi(\mathbf{X},\mathbf{Y}) \Big] \tag{14}$$

$$+ \min_{\hat{P}(\hat{\mathbf{y}}|\mathbf{x})} \mathbb{E}_{P(\mathbf{x})\check{P}(\check{\mathbf{y}}|\mathbf{x})\hat{P}(\hat{\mathbf{y}}|\mathbf{x})} \Big[\text{loss}(\hat{\mathbf{Y}},\check{\mathbf{Y}}) \Big] \Bigg)$$

$$\overset{(b)}{=} \min_{\hat{P}(\hat{\mathbf{y}}|\mathbf{x})} \mathbb{E}_{P(\mathbf{x},\mathbf{y})\hat{P}(\hat{\mathbf{y}}|\mathbf{x})} \Big[\text{loss}(\hat{\mathbf{Y}},\mathbf{Y}) \Big] \tag{15}$$

$$\overset{(c)}{=} \mathbb{E}_{P(\mathbf{x})} \Big[\min_{\hat{\mathbf{y}}} \mathbb{E}_{P(\mathbf{y}|\mathbf{x})} \Big[\text{loss}(\hat{\mathbf{y}},\mathbf{Y}) \Big| \mathbf{X} \Big] \Big], \tag{16}$$

where: (a) follows from Lagrangian duality and rearranging the expectation terms; Eq. (14) can only avoid being unboundedly negative by choosing $\check{P}(\mathbf{y}|\mathbf{x}) = P(\mathbf{y}|\mathbf{x})$, leading to cancellations of (b)[1]; and reducing the minimization of a linear function to a non-probabilistic decision via (c). This is, by definition, the set of risk-minimizing predictions. □

Thus, when learning from any true distribution of sequence data, $P(\mathbf{y}, \mathbf{x})$, using a sufficiently expressive feature representation to capture its sequential relationships, the predictor minimizing the Hamming loss will be obtained.

4 Experiments

In this section, we demonstrate the effectiveness of our proposed AST model.

4.1 Dataset Descriptions

We investigate activity recognition datasets (for both human or animal activities), and natural language processing datasets. The properties of the training datasets and testing datasets are summarized in Table 2.

Table 2: Evaluation datasets and characteristics.

Name	Classes	Attributes	Train/Test Sequences	Train/Test Variables
Human Activity	12	561	395 / 174	7767 / 3162
Baboon (day 1)	7	24	12 / 12	718 / 718
Baboon (day 2)	7	24	12 / 12	718 / 718
FAQSeg	4	24	26 / 22	36327 / 21618

Human Activity Recognition Dataset [Reyes-Ortiz *et al.*, 2015] was collected from 30 volunteers wearing smart phones on their waists. There are 12 different activities labeled in this dataset (walking, walking upstairs, walking downstairs, sitting, standing, laying, stand-to-sit, sit-to-stand, sit-to-lie, lie-to-sit, stand-to-lie, and lie-to-stand). We segment the data into consecutive temporal sequences. There are 561 features for each time step based on accelerometer and gyroscope measurements collected from the smart phones.

Baboon Activity Recognition Dataset [Strandburg-Peshkin *et al.*, 2015; Crofoot *et al.*, 2015] consists of GPS and accelerometer data gathered for 12 hours each day for 35 days from 26 adult and sub-adult members of a baboon troop wearing sensor collars. Four experts labeled two days of troop activities (e.g. sleeping, hanging out, coordinated progression, coordinated non-progression). We consider the majority vote of their annotations to be the ground truth label. We segment each day of data into 12 one hour sequences. We use 24 features to create each prediction model. These include the average speed of the group and other group location-based measurements. We report two results: using the labeled first day of data to classify the second day's activities (Baboon (day 1)); and using the second day's labeled data to classify the first day's activities (Baboon (day 2)).

FAQ Segmentation Dataset [McCallum *et al.*, 2000] contains 48 Frequently Asked Questions (FAQs) downloaded from the Internet. 26 are used for training and 22 for testing. Each line in the document is labeled with four possible labels: head, question, answer, and tail. 24 Boolean features are generated for each line.

4.2 Methodology

We compare our proposed adversarial sequence tagging model against the state-of-the-art methods for structured prediction. The methods details are as follows:

A linear chain **Conditional Random Field (CRF)** [Sarawagi and Cohen, 2004] with features based on the transition between labels $\phi(y_t, y_{t+1})$ and input variables/labels $\phi(x_t, y_t)$. We use LBFGS for optimizing the model. We selected the regularization weights using a validation set (approximately 10% of the data)[2].

For **Structural SVM (SSVM)**, we use the SVM^{hmm} implementation of structural SVM inside the SVM^{light} package [Joachims, 1999]. SVM^{hmm} is implemented to learn a model with chain structure. We include the first-order tag sequence as features. We use a validation set of 10% of the data for selecting the parameter c which controls the trade-off between slack and the magnitude of the weights vectors, and default parameters for the remaining settings.

For our **Adversarial Sequence Tagging (AST)** approach, we implemented our previously described learning and prediction algorithms. Our features are those of the CRF package [Sarawagi and Cohen, 2004]. For training and testing, we use the oracle approach on each data sequence. We optimized using stochastic gradient descent to learn the AST model parameters. We note that the initial action set for our methods does not significantly influence the results (we use sequences $\{11\ldots1, 22\ldots2, \ldots\}$ for each player). We use deterministic predictions using the sequence with the maximum probability rather than making stochastic predictions. We use Gurobi [Gurobi Optimization, 2015] as the linear programming solver to compute equilibria.

[1] For additively decomposing potentials, $\psi(\mathbf{x}, \mathbf{y}) = \sum_t \psi'(\mathbf{x}, y_t, y_{t+1})$, only pairwise conditional probabilities must match: $\hat{P}(y_t, y_{t+1}|\mathbf{x}) = P(y_t, y_{t+1}|\mathbf{x})$. However, since $P(\mathbf{y}|\mathbf{x})$ is Markovian by assumption, the entire conditional sequence distributions match as well.

[2] Since the number of baboon data sequences is small, we did not use a validation set for parameter tuning in baboon experiments.

Table 3: Per-variable accuracy for the three approaches on different datasets.

Dataset	CRF	SSVM	AST
Human Activity	97.12%	97.03%	**97.19%**
Baboon (day 1)	75.63%	75.63%	**77.30%**
Baboon (day 2)	68.66%	63.65%	**69.22%**
FAQSeg	87.62%	94.23%	**94.42%**

4.3 Results

We evaluate performance, shown in Table 3, using the per-variable accuracy (the complement of the Hamming loss) as our performance measure. Our proposed approach, AST, consistently outperforms the CRF and SSVM on the four datasets. However, SSVM performs sometimes better and sometimes worse than CRF. The reason is likely due to the convex approximation of the hinge loss and logloss, which can create more errors in some cases. In contrast, our approach, AST, outperforms CRF and SSVM by minimizing the loss for an adversarial approximation of the training data. This upper bounds the generalization loss since real data is not likely to be worst case. Other approaches minimize surrogate losses, which upper bound the Hamming loss, on training data samples. These two approaches can be viewed as approximating the training data (and using the exact loss function of interest) versus approximating the loss function (and using the exact training data). We believe the former more closely aligns with test performance. Our consistency results show this to be true for certain feature representations and data distributions when compared to the hinge loss surrogate of the Hamming loss.

The differences in loss measures that the methods attempt to optimize offers some explanation for the performance differences of CRF and SSVM. For example, the hinge loss approximation of the Hamming loss on test data for FAQSeg is 2,816.04 for SSVM, 3,961.25 for CRF, and 35,291.35 for AST. Thus, SSVM is providing much better performance on the measure it is designed to minimize, but this does not translate into better Hamming loss due to differences introduced by the hinge approximation.

Table 4: Prediction time for the three approaches on different datasets (in seconds) using double oracle AST.

Dataset	CRF	SSVM	AST
Human Activity	1050	0.04	193
Baboon (day 1)	4.8	0	2.6
Baboon (day 2)	4.5	0	2.5
FAQSeg	108	0.1	15.8

Table 4 shows the amount of time required to make predictions for all of the testing sequences. The SSVM package is well optimized so that the running time is very fast. This provides a good baseline for comparison. Although the AST model takes longer than the SSVM approach, the improvement in accuracy can often be worth the additional running time. At the same time, the AST model's computation time is in some cases almost an order of magnitude more efficient than CRF prediction, which is limited by the need to compute the normalization term for a distribution over sequences.

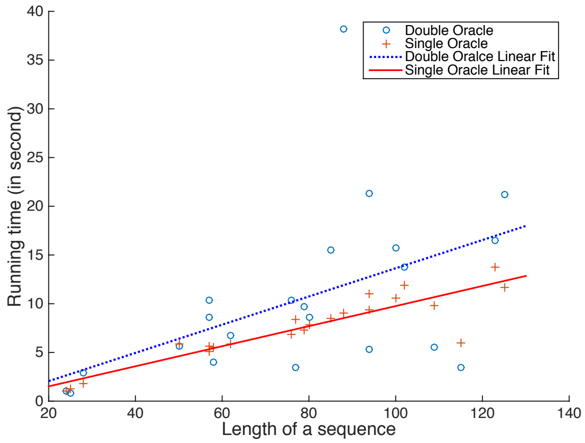

Figure 1: Running time for single oracle and double oracle.

Figure 1 shows the running time comparing double oracle and single oracle approaches on the more time-consuming Human Activity dataset on test instances of minimum length 20. For longer sequences, single oracle requires less time than double oracle. This demonstrates that the single oracle can be useful for long sequences with many labels. Unfortunately, for very short sequences (e.g., those less than length 20), the double oracle method is consistently more efficient on average. When short sequences dominate the distribution of training data, which is the case for many problems, the single oracle method's average running time is slower than double oracle method. This suggests a hybrid approach that uses the double oracle method for short sequences and the single oracle method for longer sequences.

5 Conclusion

We have developed AST, a sequence tagging method for inductively minimizing Hamming loss that is both consistent and performs well in practice. This stands in contrast with existing methods: maximum margin methods (SSVMs and M^3Ns) are not consistent and can be shown to have arbitrarily large loss for certain data distributions; conditional random fields, though consistent, use a surrogate loss that differs substantially from the Hamming loss. For both alternatives, we have shown AST to provide better sequence tags. Further, we have introduced a single oracle inference procedure for AST that improves the computational efficiency of the approach on tasks with long sequences and many possible labels.

Acknowledgments

The research in this paper was supported in part by the NSF grants III-1514126 (Ziebart, Berger-Wolf), CNS-1248080 (Berger-Wolf), and RI-1526379 (Ziebart). We thank the reviewers for their valuable comments.

References

[Asif *et al.*, 2015] Kaiser Asif, Wei Xing, Sima Behpour, and Brian D. Ziebart. Adversarial cost-sensitive classification. In *Proceedings of the Conference on Uncertainty in Artificial Intelligence*, 2015.

[Biggio *et al.*, 2010] Battista Biggio, Giorgio Fumera, and Fabio Roli. Multiple classifier systems for robust classifier design in adversarial environments. *International Journal of Machine Learning and Cybernetics*, 1(1-4):27–41, 2010.

[Crofoot *et al.*, 2015] Margaret C Crofoot, Roland W Kays, and Martin Wikelski. Data from: Shared decision-making drives collective movement in wild baboons, 2015.

[Dalvi *et al.*, 2004] Nilesh Dalvi, Pedro Domingos, Sumit Sanghai, Deepak Verma, et al. Adversarial classification. In *KDD*, pages 99–108. ACM, 2004.

[Grünwald and Dawid, 2004] Peter D. Grünwald and A. Phillip Dawid. Game theory, maximum entropy, minimum discrepancy, and robust Bayesian decision theory. *Annals of Statistics*, 32:1367–1433, 2004.

[Gurobi Optimization, 2015] Inc. Gurobi Optimization. Gurobi optimizer reference manual, 2015.

[Hoffgen *et al.*, 1995] Klaus-Uwe Hoffgen, Hans-Ulrich Simon, and Kevin S Vanhorn. Robust trainability of single neurons. *Journal of Computer and System Sciences*, 50(1):114–125, 1995.

[Joachims *et al.*, 2009] Thorsten Joachims, Thomas Finley, and Chun-Nam John Yu. Cutting-plane training of structural SVMs. *Machine Learning*, 77(1):27–59, 2009.

[Joachims, 1999] Thorsten Joachims. Making large scale SVM learning practical. Technical report, Universität Dortmund, 1999.

[Lafferty *et al.*, 2001] John Lafferty, Andrew McCallum, and Fernando Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proc. of the International Conference on Machine Learning*, pages 282–289, 2001.

[Lanckriet *et al.*, 2003] Gert R.G. Lanckriet, Laurent El Ghaoui, Chiranjib Bhattacharyya, and Michael I. Jordan. A robust minimax approach to classification. *JMLR*, 3:555–582, 2003.

[Liao *et al.*, 2007] Lin Liao, Dieter Fox, and Henry Kautz. Extracting places and activities from gps traces using hierarchical conditional random fields. *The International Journal of Robotics Research*, 26(1):119–134, 2007.

[Liu and Ziebart, 2014] Anqi Liu and Brian D. Ziebart. Robust classification under sample selection bias. In *Advances in Neural Information Processing Systems*, 2014.

[Liu, 2007] Yufeng Liu. Fisher consistency of multicategory support vector machines. In *International Conference on Artificial Intelligence and Statistics*, pages 291–298, 2007.

[Lowd and Meek, 2005] Daniel Lowd and Christopher Meek. Adversarial learning. In *Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining*, pages 641–647. ACM, 2005.

[McCallum *et al.*, 2000] A. McCallum, D. Freitag, and F. Pereira. Maximum entropy Markov models for information extraction and segmentation. In *Proc. International Conference on Machine Learning*, pages 591–598, 2000.

[McMahan *et al.*, 2003] H Brendan McMahan, Geoffrey J Gordon, and Avrim Blum. Planning in the presence of cost functions controlled by an adversary. In *Proceedings of the International Conference on Machine Learning*, pages 536–543, 2003.

[Reyes-Ortiz *et al.*, 2015] Jorge-L Reyes-Ortiz, Luca Oneto, Albert Samà, Xavier Parra, and Davide Anguita. Transition-aware human activity recognition using smartphones. *Neurocomputing*, 2015.

[Sarawagi and Cohen, 2004] Sunita Sarawagi and William W Cohen. Semi-markov conditional random fields for information extraction. In *Advances in Neural Information Processing Systems*, pages 1185–1192, 2004.

[Sha and Pereira, 2003] F. Sha and F. Pereira. Shallow parsing with conditional random fields. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology*, pages 134–141, 2003.

[Strandburg-Peshkin *et al.*, 2015] Ariana Strandburg-Peshkin, Damien R Farine, Iain D Couzin, and Margaret C Crofoot. Shared decision-making drives collective movement in wild baboons. *Science*, 348(6241):1358–1361, 2015.

[Taskar *et al.*, 2004] Ben Taskar, Carlos Guestrin, and Daphne Koller. Max-margin Markov networks. *Advances in neural information processing systems*, 16:25, 2004.

[Topsøe, 1979] Flemming Topsøe. Information theoretical optimization techniques. *Kybernetika*, 15(1):8–27, 1979.

[Tsochantaridis *et al.*, 2004] Ioannis Tsochantaridis, Thomas Hofmann, Thorsten Joachims, and Yasemin Altun. Support vector machine learning for interdependent and structured output spaces. In *Proceedings of the twenty-first international conference on Machine learning*, page 104. ACM, 2004.

[Vail *et al.*, 2007] Douglas L. Vail, Manuela M. Veloso, and John D. Lafferty. Conditional random fields for activity recognition. In *Proc. International Conference on Autonomous Systems and Multiagent Systems*, pages 1–8, 2007.

[Viterbi, 1967] Andrew J Viterbi. Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *Information Theory, IEEE Transactions on*, 13(2):260–269, 1967.

[von Neumann and Morgenstern, 1947] John von Neumann and Oskar Morgenstern. *Theory of Games and Economic Behavior*. Princeton University Press, 1947.

[Wang *et al.*, 2015] Hong Wang, Wei Xing, Kaiser Asif, and Brian D. Ziebart. Adversarial prediction games for multivariate losses. In *Advances in Neural Information Processing Systems*, 2015.