

---

# Active Learning with Perceptron for Structured Output

---

Dan Roth  
Kevin Small

DANR@UIUC.EDU  
KSMALL@UIUC.EDU

Department of Computer Science, University of Illinois at Urbana-Champaign, Urbana, IL 61801

## Abstract

Typically, structured output scenarios are characterized by a high cost associated with obtaining supervised training data, motivating the study of active learning protocols for these situations. Starting with active learning approaches for multiclass classification, we first design querying functions for selecting entire structured instances, exploring the tradeoff between selecting instances based on a global margin or a combination of the margin of local classifiers. We then look at the setting where subcomponents of the structured instance can be queried independently and examine the benefit of incorporating structural information for active learning in such scenarios. Empirical results using these querying functions on both synthetic data and the semantic role labeling task demonstrate a significant reduction in the need for supervised training data.

## 1. Introduction

The successful application of machine learning algorithms to many domains is limited by the inability to obtain an adequate amount of labeled training data due to practical constraints associated with the specific task. The *active learning* paradigm offers one promising solution to learning with partially labeled data sets by allowing the learning algorithm to incrementally select a subset of the unlabeled data to present for labeling by the domain expert with the goal of maximizing performance while minimizing the labeling effort. This model is especially appealing when learning in structured output spaces as the associated application domains are generally very complex and the cost for supervised data particularly expensive.

---

Appearing in *Proceedings of the ICML Workshop on Learning in Structured Output Spaces*, Pittsburgh, PA, 2006. Copyright 2006 by the author(s)/owner(s).

This paper describes a margin-based method for active learning in structured output spaces where the interdependencies between output variables are described by a general set of constraints able to represent any structural form. Specifically, we study two querying protocols and propose novel querying functions for active learning in structured output spaces: querying complete labels and querying partial labels. We then describe a particular algorithmic implementation of the developed theory based on the Perceptron algorithm and propose a mistake-driven explanation for the relative performance of the querying functions. Finally, we provide empirical evidence on both synthetic data and the semantic role labeling (SRL) task to demonstrate the effectiveness of the proposed methods.

## 2. Preliminaries

This work builds upon existing work for learning in structured output spaces and margin-based active learning. We first describe a general framework for learning in structured output, following the approach of incorporating output variable interdependencies directly into a discriminative learning model (Collins, 2002; Punyakanok et al., 2005). We then describe previous margin-based active learning approaches based on the output of linear classifiers (Tong & Koller, 2001; Yan et al., 2003).

### 2.1. Structured Output Spaces

For our setting, let  $\mathbf{x} \in \mathcal{X}^{n_x}$  represent an instance in the space of input variables  $\mathbf{X} = (X_1, \dots, X_{n_x})$ ;  $X_t \in \mathbb{R}^{d_t}$  and  $\mathbf{y} \in \mathcal{C}(\mathcal{Y}^{n_y})$  represent a structured assignment in the space of output variables  $\mathbf{Y} = (Y_1, \dots, Y_{n_y})$ ;  $Y_t \in \{\omega_1, \dots, \omega_{k_t}\}$ .  $\mathcal{C} : 2^{\mathcal{Y}^*} \rightarrow 2^{\mathcal{Y}^*}$  represents a set of constraints that enforces structural consistency on  $\mathbf{Y}$  such that  $\mathcal{C}(\mathcal{Y}^{n_y}) \subseteq \mathcal{Y}^{n_y}$ . A learning algorithm for structured output spaces takes  $m$  structured training instances,  $S = \{(\mathbf{x}_1, \mathbf{y}_1), \dots, (\mathbf{x}_m, \mathbf{y}_m)\}$  drawn i.i.d over  $\mathcal{X}^{n_x} \times \mathcal{C}(\mathcal{Y}^{n_y})$  and returns a classifier  $h : \mathcal{X}^{n_x} \rightarrow \mathcal{Y}^{n_y}$ . This assignment generated by  $h$  is based on a global scoring function  $f : \mathcal{X}^{n_x} \times \mathcal{Y}^{n_y} \rightarrow \mathbb{R}$ ,

which assigns a score to each structured instance/label pair  $(\mathbf{x}_i, \mathbf{y}_i)$ . Given an instance  $\mathbf{x}$ , the resulting classification is given by

$$\hat{\mathbf{y}}_C = h(\mathbf{x}) = \operatorname{argmax}_{\mathbf{y}' \in \mathcal{C}(\mathcal{Y}^{n_y})} f(\mathbf{x}, \mathbf{y}'). \quad (1)$$

The output variable assignments are determined by a global scoring function  $f(\mathbf{x}, \mathbf{y})$  that can be decomposed into local scoring functions  $f_{y_t}(\mathbf{x}, t)$  such that  $f(\mathbf{x}, \mathbf{y}) = \sum_{t=1}^{n_y} f_{y_t}(\mathbf{x}, t)$ . When structural consistency is not enforced, the global scoring function will output the value  $f(\mathbf{x}, \hat{\mathbf{y}})$  resulting in assignments given by  $\hat{\mathbf{y}} = \operatorname{argmax}_{\mathbf{y}' \in \mathcal{Y}^{n_y}} f(\mathbf{x}, \mathbf{y}')$ . An inference mechanism takes the scoring function  $f(\mathbf{x}, \mathbf{y})$ , an instance  $(\mathbf{x}, \mathbf{y})$ , and a set of constraints  $\mathcal{C}$ , returning an optimal assignment  $\hat{\mathbf{y}}_C$  based on the global score  $f(\mathbf{x}, \hat{\mathbf{y}}_C)$  consistent with the defined output structure. Specifically, we will use general constraints with the ability to represent any structure and thereby require a general search mechanism for inference to enforce structural consistency (Daumé III & Marcu, 2005). As active learning querying functions are very particular about selecting instances with specific properties, we will define the notions of *locally learnable instances* and *globally learnable instances* for exposition purposes.

**Definition 1 (Locally Learnable Instance)** *Given a classifier,  $f \in \mathcal{H}$ , an instance  $(\mathbf{x}, \mathbf{y})$  is locally learnable if  $f_{y_t}(\mathbf{x}, t) > f_{y'}(\mathbf{x}, t)$  for all  $y' \in \mathcal{Y} \setminus y_t$ . In this situation,  $\hat{\mathbf{y}}_C = \hat{\mathbf{y}} = \mathbf{y}$ .*

**Definition 2 (Globally Learnable Instance)** *Given a classifier,  $f \in \mathcal{H}$ , an instance  $(\mathbf{x}, \mathbf{y})$  is globally learnable if  $f(\mathbf{x}, \mathbf{y}) > f(\mathbf{x}, \mathbf{y}')$  for all  $\mathbf{y}' \in \mathcal{Y} \setminus \mathbf{y}$ . We will refer to instances that are globally learnable, but not locally learnable as **exclusively globally learnable** in which case  $\hat{\mathbf{y}} \neq \hat{\mathbf{y}}_C = \mathbf{y}$ .*

## 2.2. Margin-based Active Learning

The key component that distinguishes active learning from standard supervised learning is a querying function  $\mathcal{Q}$  which when given unlabeled data  $\mathcal{S}_u$  and the current learned classifier returns a set of unlabeled examples  $\mathcal{S}_{select} \subseteq \mathcal{S}_u$ . These selected examples are labeled by a domain expert and provided to the learning algorithm to incrementally update its hypothesis. The most widely used active learning schemes utilize querying functions based on heuristics to reduce the labeling effort, often based on assigning a measure of certainty to predictions on the unlabeled data and selecting examples with low certainty.

We denote the margin of an example relative to the hypothesis function as  $\rho(\mathbf{x}, \mathbf{y}, f)$ , noting that this value

is positive if and only if  $\hat{\mathbf{y}}_C = \mathbf{y}$  and the magnitude is associated with the confidence in the prediction. The specific definition of margin for a given setting is generally dependent on the description of the output space. A *margin-based learning algorithm* is a learning algorithm which selects a hypothesis by minimizing a loss function  $\mathcal{L} : \mathbb{R} \rightarrow [0, \infty)$  using the margin of instances contained in  $\mathcal{S}_l$ . We correspondingly define an active learning algorithm with a querying function dependent on  $\rho(\mathbf{x}, \mathbf{y}, f)$  as a *margin-based active learning algorithm*.

The standard active learning algorithm for binary classification,  $Y \in \{-1, 1\}$ , with linear functions utilizes the querying function  $\mathcal{Q}_{binary}$  (Tong & Koller, 2001), which makes direct use of the margin  $\rho_{binary}(\mathbf{x}, \mathbf{y}, f) = y \cdot f(\mathbf{x})$  by assuming the current classifier makes correct predictions on the labeled examples, selecting those unlabeled examples with the smallest margin and thereby the least certainty,

$$\mathcal{Q}_{binary} : x_* = \operatorname{argmin}_{x \in \mathcal{S}_u} |f(\mathbf{x})|.$$

For multiclass classification with a winner-take-all network, a widely accepted definition for multiclass margin is  $\rho_{multiclass}(\mathbf{x}, \mathbf{y}, f) = f_y(\mathbf{x}) - f_{\hat{y}}(\mathbf{x})$ , where  $y$  represents the true label and  $\hat{y}$  represents the highest activation value such that  $\hat{y} \neq y$  (Har-Peled et al., 2003). Previous work on multiclass active learning (Yan et al., 2003) advocate a querying function closely related to using this definition of multiclass margin,

$$\mathcal{Q}_{multiclass} : x_* = \operatorname{argmin}_{x \in \mathcal{S}_u} [f_{\hat{y}}(\mathbf{x}) - f_{\tilde{y}}(\mathbf{x})],$$

where  $\hat{y}$  represents the predicted label and  $\tilde{y}$  represents the label with the second highest activation value.

## 3. Active Learning for Structures

We look to augment the aforementioned work to design querying functions for learning in structured output spaces by exploiting structural knowledge not available for individual classifications. To formalize the task of querying labels in a structured learning scenario, we separate the labels of an instance/label pair  $(\mathbf{x}, \mathbf{y})$  into  $\mathbf{y}_l$  and  $\mathbf{y}_u$ , representing which labels are available and unavailable respectively. Without loss of generality, we assume that  $y_t$  represents a multiclass classification.

### 3.1. Querying Complete Labels

The task of designing a querying function for complete labels entails selecting instances  $\mathbf{x}$  such that all output labels associated with the specified instance will be provided by the domain expert. More formally, complete label querying functions select examples from the

set  $(\mathbf{x}, \mathbf{y}_l = \emptyset, \mathbf{y}_u = \mathbf{y}) \in \mathcal{S}_u$  and receives a labeled instance  $(\mathbf{x}, \mathbf{y}_l = \mathbf{y}, \mathbf{y}_u = \emptyset) \in \mathcal{S}_{select}$  for training.

Following the margin-based justification for designing querying functions, a common definition of margin for learning in structured output spaces is given by  $\rho_{global}(\mathbf{x}, \mathbf{y}, f) = f(\mathbf{x}, \mathbf{y}) - f(\mathbf{x}, \hat{\mathbf{y}}_C)$  where  $\hat{\mathbf{y}}_C$  is the highest activation value consistent with the output structure such that  $\hat{\mathbf{y}}_C \neq \mathbf{y}$ . The corresponding querying function for a structured learner that incorporates the constraints into the learning model is defined by

$$\mathcal{Q}_{global} : x_\star = \operatorname{argmin}_{x \in \mathcal{S}_u} [f(\mathbf{x}, \hat{\mathbf{y}}_C) - f(\mathbf{x}, \tilde{\mathbf{y}}_C)],$$

where  $\hat{\mathbf{y}}_C$  is the predicted labeling and  $\tilde{\mathbf{y}}_C$  is the labeling associated with the second highest activation, both consistent with structural constraints.

It should be noted that  $\mathcal{Q}_{global}$  does not necessarily require the scoring function  $f(\mathbf{x}, \mathbf{y})$  to be decomposable, thereby allowing its use with an arbitrary loss function. The only requirement is that the inference algorithm is able to return both  $f(\mathbf{x}, \hat{\mathbf{y}}_C)$  and  $f(\mathbf{x}, \tilde{\mathbf{y}}_C)$  for a given structured instance. However, for many structured learning settings, the scoring function and consequently the loss function is able to be decomposed into local classification problems. Furthermore, it has been observed that when the local classification problems are easy to learn without regard for structural constraints, directly optimizing these local functions often leads to a lower sample complexity (Punyakonok et al., 2005). Since this lower sample complexity is predicated on making concurrent local updates for each structured instance, it may be desirable to select examples that make as many local updates as possible for such situations. This observation motivates designing a querying function that selects instances based on optimization of local predictions. If we view the individual output variable predictions independently, a reasonable margin-based strategy is to select examples with a small average local multiclass margin,

$$\mathcal{Q}_{local(C)} : x_\star = \operatorname{argmin}_{x \in \mathcal{S}_u} \frac{\sum_{t=1}^{n_y} [f_{\hat{y}_{C,t}}(\mathbf{x}, t) - f_{\tilde{y}_{C,t}}(\mathbf{x}, t)]}{n_y}$$

where  $\hat{y}_{C,t}$  represents the local predicted label consistent with the global constraints and  $\tilde{y}_{C,t}$  represents the second highest valued local prediction consistent with global constraints.

### 3.2. Querying Partial Labels

As  $\mathcal{Q}_{global}$  is the least restrictive querying function, making no assumptions regarding decomposability of of the scoring function, and  $\mathcal{Q}_{local(C)}$  requires only that

the scoring function be decomposable in accordance with the local output variables, we also explore active learning for structured output in settings where local output variables can be queried independently, defined as querying partial labels. More formally, partial label querying functions select examples from the set  $(\mathbf{x}, \mathbf{y}_l, \mathbf{y}_u) \in \mathcal{S}_u$  and receives from the domain expert a label for the local output variable  $y_q$ , thereby moving  $y_q$  from  $\mathbf{y}_u$  to  $\mathbf{y}_l$  for that specific instance.

The intuitive benefit of querying partial labels is that we no longer select entire instances and are thereby not hindered by cases where the label of one local output variable is very informative, but other output variables associated with the same instance are minimally useful, but still add cost to the labeling effort. Conversely, this configuration is not immediately usable for applications not easily decomposable into local output variables that can be independently queried. Secondly, there is a fixed cost of a domain expert labeling structured instances associated with processing the entire instance which would normally be amortized over the individual output assignments. However, as we shall see, this approach is very beneficial in scenarios where querying partial labels is possible.

Noting that querying partial labels is essentially requesting a single multiclass classification, the naive approach to active learning in this case is to simply ignore the structural information and use  $\mathcal{Q}_{multiclass}$ , resulting in the querying function

$$\mathcal{Q}_{local} : (\mathbf{x}, t)_\star = \operatorname{argmin}_{\substack{(\mathbf{x}, y_t) \in \mathcal{S}_u \\ t=1, \dots, n_y}} [f_{\hat{y}_t}(\mathbf{x}, t) - f_{\tilde{y}_t}(\mathbf{x}, t)].$$

One of the stronger arguments for margin-based active learning is the notion of selecting instances which attempt to halve the version space with each selection (Tong & Koller, 2001). A local classifier which either ignores or is ignorant of the structural constraints maintains a version space described by

$$\mathcal{V}_{local} = \{f \in \mathcal{H} | f_{y_t}(\mathbf{x}, t) > f_{\tilde{y}_t}(\mathbf{x}, t); \forall (\mathbf{x}, y) \in \mathcal{S}_l\}.$$

If the learning algorithm has access to an inference mechanism that maintains structural consistency, the version space is only dependent on the subset of possible output variable assignments that are consistent with the global structure,

$$\mathcal{V}_{local(C)} = \{f \in \mathcal{H} | f_{y_t}(\mathbf{x}, t) > f_{\tilde{y}_{C,t}}(\mathbf{x}, t); \forall (\mathbf{x}, y) \in \mathcal{S}_l\}$$

where  $\tilde{y}_{C,t}$  represents the highest local activation value of a predicted label consistent with the global constraints such that  $\tilde{y}_{C,t} \neq y$ . Therefore, if the learning algorithm incorporates structural consistency directly

into the learning model, we advocate also utilizing this information to augment  $\mathcal{Q}_{local}$ , resulting in the querying function

$$\mathcal{Q}_{local(C)} : (\mathbf{x}, t)_* = \underset{\substack{(\mathbf{x}, y_t) \in \mathcal{S}_u \\ t=1, \dots, n_y}}{\operatorname{argmin}} [f_{\hat{y}_{C,t}}(\mathbf{x}, t) - f_{\tilde{y}_{C,t}}(\mathbf{x}, t)].$$

In addition to the version space justification, there are other reasons to exploit structural knowledge in the design of an active learning querying function for partial labels. First of all, if the data is locally separable,  $\mathcal{Q}_{local(C)}$  becomes  $\mathcal{Q}_{local}$  with the only cost being computation associated with inference. Secondly, as other labels within a structured instance become visible, a notion similar to correction propagation (Culotta & McCallum, 2005) becomes possible. In addition to the global constraints, each partial label queried further constrains the other output variables of a structured instance, reducing the consistent output space size at each step. In many cases, this process dramatically reduces the output space for the remaining local variables, reducing the need for further partial queries.

## 4. Active Learning with Perceptron

This work specifically utilizes classifiers of a linear representation with parameters learned using the Perceptron algorithm. In this case,  $f(\mathbf{x}, \mathbf{y}) = \boldsymbol{\alpha} \cdot \Phi(\mathbf{x}, \mathbf{y})$  represents the global scoring function such that  $\boldsymbol{\alpha} = (\boldsymbol{\alpha}^1, \dots, \boldsymbol{\alpha}^{|\mathcal{Y}|})$  is a concatenation of the local  $\boldsymbol{\alpha}^y$  vectors and  $\Phi(\mathbf{x}, \mathbf{y}) = (\Phi^1(\mathbf{x}, \mathbf{y}), \dots, \Phi^{|\mathcal{Y}|}(\mathbf{x}, \mathbf{y}))$  is a concatenation of the local feature vectors,  $\Phi^y(\mathbf{x}, \mathbf{y})$ . Utilizing this notation,  $f_y(\mathbf{x}, t) = \boldsymbol{\alpha}^y \cdot \Phi^y(\mathbf{x}, t)$  where  $\boldsymbol{\alpha}^y \in \mathbb{R}^{d_y}$  is the learned weight vector and  $\Phi^y(\mathbf{x}, t) \in \mathbb{R}^{d_y}$  is the feature vector for local classifications.

### 4.1. Inference Based Training

Margin-based active learning generally relies upon the use of support vector machines (SVM) (Tong & Koller, 2001; Yan et al., 2003). While there is existing work on SVM for structured output (Tsochantaridis et al., 2004), the incremental nature of active learning over large data sets associated with structured output makes these algorithms impractical for such uses. This work builds upon the *inference based training* (IBT) learning strategy (Punyakank et al., 2005; Collins, 2002) shown in Table 1, which incorporates the structural knowledge into the learning procedure. We first modify the IBT algorithm for partial labels by updating only local components with visible labels. Secondly, we add a notion of large margin IBT heuristically by requiring thick separation between class activations. While this can likely be tuned to improve

Table 1. Learning with Inference Based Feedback (IBT)

INPUT: $\mathcal{S} \in \{\mathcal{X}^* \times \mathcal{Y}^*\}^m, \gamma, T$
Initialize $\boldsymbol{\alpha} \leftarrow 0$
Repeat for $T$ iterations
foreach $(\mathbf{x}, \mathbf{y}) \in \mathcal{S}$
$\hat{\mathbf{y}}_C \leftarrow \operatorname{argmax}_{\mathbf{y} \in \mathcal{C}(\mathcal{Y}^{n_y})} \boldsymbol{\alpha} \cdot \Phi(\mathbf{x}, \mathbf{y})$
foreach $t = 1, \dots, n_y$ such that $(\mathbf{x}, y_t) \in \mathcal{S}_l$
if $f_{y_t}(\mathbf{x}, t) - \gamma < f_{\hat{y}_t}(\mathbf{x}, t)$
$\boldsymbol{\alpha}^{y_t} \leftarrow \boldsymbol{\alpha}^{y_t} + \Phi^{y_t}(\mathbf{x}, t)$
$\boldsymbol{\alpha}^{\hat{y}_t} \leftarrow \boldsymbol{\alpha}^{\hat{y}_t} - \Phi^{\hat{y}_t}(\mathbf{x}, t)$
OUTPUT: $\{f_y\}_{y \in \mathcal{Y}} \in \mathcal{H}$

performance depending on the data, we simply set  $\gamma = 1.0$  and require that  $\|\Phi^{y_t}(\mathbf{x}, t)\| = 1$  through normalization for our experiments. During learning, we set  $T = 7$  for synthetic data and  $T = 5$  for experiments with the SRL task. To infer  $\hat{\mathbf{y}}_C$ , we use an index ordered beam search with beam size of 50 for synthetic data and 100 for SRL. Beam search was used since it performs well, is computationally fast, accommodates general constraints, and returns a global score ranking which is required for  $\mathcal{Q}_{global}$ .

### 4.2. Mistake-driven Active Learning

A greedy criteria for active learning querying functions makes the most immediate progress towards learning the target function with each requested label. For the mistake-driven Perceptron algorithm, a reasonable heuristic for measuring progress is to track the number of additive updates for each query. This intuition proposes two metrics to explain the performance results of a given querying function, *average Hamming error per query*,  $\mathcal{M}_{Hamming}$ , and *average global error per query*,  $\mathcal{M}_{global}$ . For a specific round of active learning, the current hypothesis is used to select a set of instances  $\mathcal{S}_{select}$  for labeling. Once the labels are received, we calculate the Hamming loss  $\mathcal{H}(h, \mathbf{x}) = \sum_{t=1; (\mathbf{x}, y_t) \in \mathcal{S}_l}^{n_y} I[\hat{y}_{C,t} \neq y]$  and the global loss  $\mathcal{G}(h, \mathbf{x}) = I[\hat{\mathbf{y}}_C \neq \mathbf{y}]$  at the time when the instance is first labeled.  $I[p]$  is an indicator function such that  $I[p] = 1$  if  $p$  is true and 0 otherwise. We measure the quality of a querying function relative to the average of these values for all queries up to the specific round of active learning.

Noting that only  $\mathcal{H}(h, \mathbf{x})$  is useful for partial labels, we hypothesize that for partial label queries or cases of complete label queries where the data sample  $\mathcal{S}$  is largely locally separable, the relative magnitude of  $\mathcal{M}_{Hamming}$  will determine the relative performance

of the querying functions. Alternatively, in the case of complete queries where the data has a significant portion that is exclusively globally separable,  $\mathcal{M}_{global}$  will be more strongly correlated with querying function performance.

## 5. Experiments

To demonstrate particular properties of the proposed querying functions, we first run active learning simulations on synthetic data. Then, to verify that these methods are practical for actual applications, we perform experiments on the SRL task.

### 5.1. Synthetic Data

Our synthetic structured output problem is comprised of five multiclass classifiers,  $h_1, \dots, h_5$ , each having the output space  $Y_i = \omega_1, \dots, \omega_4$ . In addition, we define the output structure using the following practical constraints:

1.  $\mathcal{C}_1 : [h_2(\mathbf{x}) \neq \omega_3] \wedge [h_5(\mathbf{x}) \neq \omega_1]$
2.  $\mathcal{C}_2 : \text{At most one } h_i(\mathbf{x}) \text{ can output } \omega_2.$
3.  $\mathcal{C}_3 : \text{For one or more } h_i(\mathbf{x}) \text{ to output } \omega_3, \text{ at least one } h_i(\mathbf{x}) \text{ must output } \omega_1.$
4.  $\mathcal{C}_4 : h_i(\mathbf{x}) \text{ can output } \omega_4 \text{ if and only if } h_{i-1}(\mathbf{x}) = \omega_1 \text{ and } h_{i-2}(\mathbf{x}) = \omega_2.$

To generate the synthetic data, we first create four linear functions of the form  $\mathbf{w}_i \cdot \mathbf{x} + b_i$  such that  $\mathbf{w}_i = [-1, 1]^{100}$  and  $b_i = [-1, 1]$  for each  $h_i$ . We then generate 5 examples in the space  $\{0, 1\}^{100}$  such that  $n$  features determined by the normal distribution  $\mathcal{N}(20, 5)$  are assigned the value 1, distributed uniformly over the feature vector. Each vector is labeled according to the function  $\text{argmax}_{t=1, \dots, k} [\mathbf{w}_t \cdot \mathbf{x} + b_t]$  resulting in the label vector  $\mathbf{y}_{local} = (h_1(\mathbf{x}), \dots, h_5(\mathbf{x}))$ . We then run the inference procedure to obtain the final labeling of the data  $\mathbf{y}$ . If  $\mathbf{y} \neq \mathbf{y}_{local}$ , then the data is exclusively globally separable. We control the total amount of such data with the parameter  $\kappa$  which represents the fraction of exclusively globally separable data in  $\mathcal{S}$ . We further filter the difficulty of the data such that all exclusively globally separable instances have a Hamming error drawn from a stated normal distribution. We generate 10000 structured examples, or equivalently 50000 local instances, in this fashion for each set of data parameters we use.

Figure 1 shows the experimental results for the described complete querying functions in addition to  $\mathcal{Q}_{local(C)}$  where an entire structured instance is based

upon the score of a single local classifier to demonstrate that it is prudent to design querying functions specifically for complete labels. The querying schedule starts as  $|\mathcal{S}_l| = 2, 4, \dots, 200$  and slowly increases step size until  $|\mathcal{S}_l| = 6000, 6100, \dots, 8000$  and 5-fold cross validation is performed. The primary observation for the synthetic data set where  $\kappa = 0.0$  is that  $\mathcal{Q}_{\overline{local(C)}}$  performs better than  $\mathcal{Q}_{global}$  when the data is locally separable. For the data set where  $\kappa = 0.3; \mathcal{N}(3, 1)$ , we see that as the data becomes less locally separable,  $\mathcal{Q}_{global}$  performs better than  $\mathcal{Q}_{\overline{local(C)}}$ . We also plot  $\mathcal{M}_{Hamming}$  and  $\mathcal{M}_{global}$  for each respective querying functions. As expected, when the data is locally separable, the querying function performance is closely related to  $\mathcal{M}_{Hamming}$  and when the data is not locally separable, the relative querying function performance is more closely related to  $\mathcal{M}_{global}$ . The vertical lines denote when the specified querying function achieves an accuracy equivalent to the largest accuracy achieved by using  $\mathcal{Q}_{random}$ . Remembering that there are 8000 training examples, we measure between 25% – 75% reduction in required training data.

Figure 2 shows our experimental results for partial querying functions using synthetic data. We ran the two partial querying functions  $\mathcal{Q}_{local}$  and  $\mathcal{Q}_{local(C)}$  in addition to  $\mathcal{Q}_{random}$ , which selects arbitrary unlabeled instances at each step, on three sets of data. The querying schedule starts by querying 10 partial labels at a time from  $|\mathcal{S}_l| = 10, 20, \dots, 2000$  and increases slowly until the step size is  $|\mathcal{S}_l| = 20000, 21000, \dots, 40000$  and once again 5-fold cross validation is performed. The first synthetic data set is where  $\kappa = 0.0$ , where the data is completely locally separable. In this case, active learning for both  $\mathcal{Q}_{local}$  and  $\mathcal{Q}_{local(C)}$  perform better than  $\mathcal{Q}_{random}$ . Somewhat more surprising is the result that  $\mathcal{Q}_{local(C)}$  performs noticeably better than  $\mathcal{Q}_{local}$  even though they should query similar points for  $\kappa = 0.0$ . The results for the synthetic data set  $\kappa = 0.3; \mathcal{N}(3, 1)$  also demonstrate a similar ordering where  $\mathcal{Q}_{local(C)}$  outperforms  $\mathcal{Q}_{local}$  which in turn outperforms  $\mathcal{Q}_{random}$ . Finally, we used a synthetic data set where  $\kappa = 1.0; \mathcal{N}(5, 1)$ , meaning that the data is completely exclusively globally separable and the difference between  $\mathcal{Q}_{local(C)}$  and  $\mathcal{Q}_{local}$  is most noticeable. For this data set, we also plotted  $\mathcal{M}_{Hamming}$  noting that this value is always greater for  $\mathcal{Q}_{local(C)}$  than  $\mathcal{Q}_{local}$ , which is consistent with our expectations for  $\mathcal{M}_{Hamming}$  relative to querying function performance. As there are 40000 training examples for each fold, we show a decrease in necessary data of between 65% – 79% depending on the specific experiment.

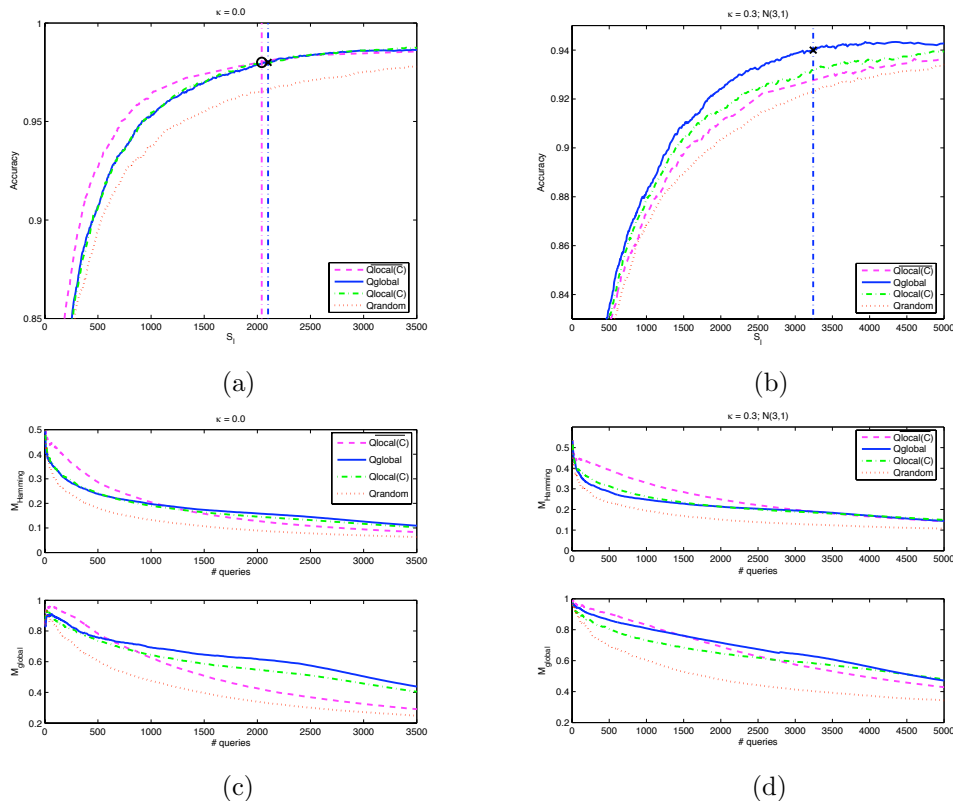


Figure 1. Experimental results for the complete label querying problem, noting that the labeling effort is reduced between 25% – 75% depending of the particular situation. (a) Active learning curve for  $\kappa = 0.0$  (b) Active learning curve for  $\kappa = 0.3; \mathcal{N}(3, 1)$  (c) Plot of  $\mathcal{M}_{hamming}$  and  $\mathcal{M}_{global}$  for  $\kappa = 0.0$  (d) Plot of  $\mathcal{M}_{hamming}$  and  $\mathcal{M}_{global}$  for  $\kappa = 0.3; \mathcal{N}(3, 1)$

## 5.2. Semantic Role Labeling

We also perform experiments on the SRL task as described in the CoNLL-2004 shared task (Carreras & Màrquez, 2004). We essentially follow the model described in (Punyakanok et al., 2005) where linear classifiers  $f_{A0}, f_{A1}, \dots$  are used to map constituent candidates to one of 45 different classes. For a given argument / predicate pair, the multiclass classifier returns a set of scores which are used to produce the output  $\hat{y}_C$  consistent with the structural constraints associated with other arguments relative to the same predicate. We simplify the task by assuming that the constituent boundaries are given, making this an argument classification task. We use the CoNLL-2004 shared task data, but restrict our experiments to sentences that have greater than five arguments to increase the number of instances with interdependent variables and take a random subset of this to get 1500 structured examples comprised of 9327 local predictions. For our testing data, we also restrict ourself to sentences with greater than five arguments, resulting in 301 structured instances comprised of 1862 local predictions.

We use the same features and the applicable subset of families of constraints which do not concern segmentation as described by (Punyakanok et al., 2004). Figure 3 shows the empirical results for the SRL experiments. For querying complete labels, we start with a querying schedule of  $|\mathcal{S}_t| = 100, 200, \dots, 500$  and slowly increase the step size until ending with  $|\mathcal{S}_t| = 1000, 1100, \dots, 1500$ . For the complete labeling case,  $Q_{local(C)}$  performs better than  $Q_{global}$ , implying that the data largely locally separable which is consistent with the findings of (Punyakanok et al., 2005). Furthermore, both functions perform better than  $Q_{random}$  with approximately a 35% reduction in labeling effort. For partial labels, we used a querying schedule that start at  $|\mathcal{S}_t| = 100, 200, \dots, 500$  and increases step size until ending at  $|\mathcal{S}_t| = 6000, 7000, \dots, 9327$ . In this case,  $Q_{local(C)}$  performs better than  $Q_{local}$  and  $Q_{random}$ , requiring only about half of the data to be labeled.

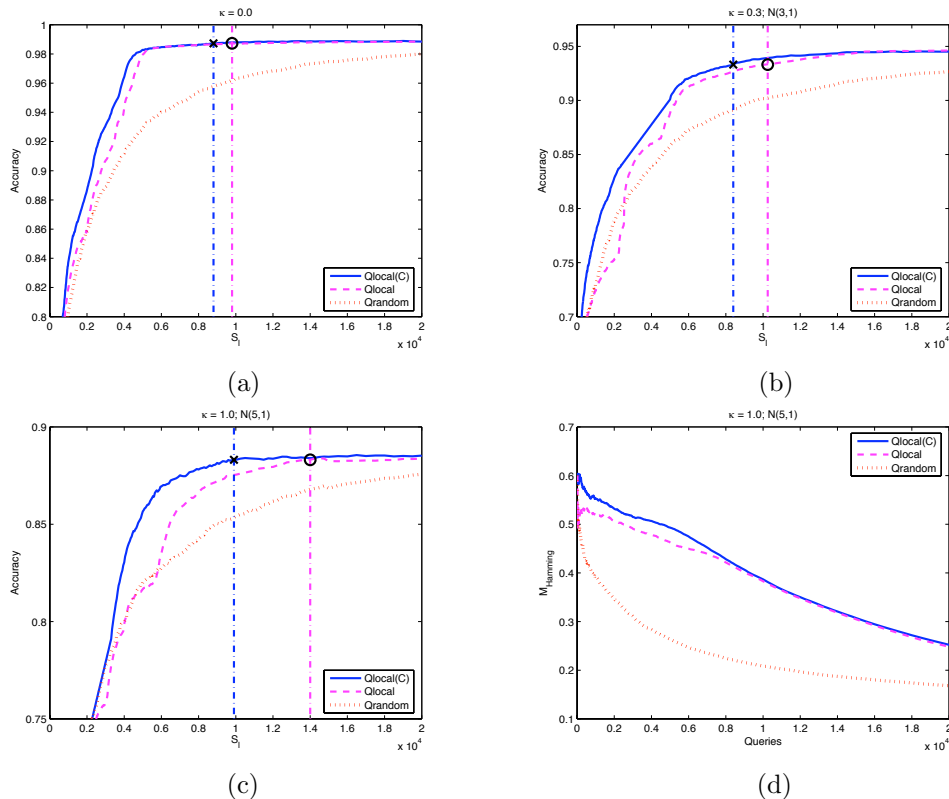


Figure 2. Experimental results for the partial label querying problem, noting that the labeling effort is reduced between 65% – 79% depending on the specific experiment. (a) Active learning curve for  $\kappa = 0.0$  (b) Active learning curve for  $\kappa = 0.3; \mathcal{N}(3, 1)$  (c) Active learning curve for  $\kappa = 1.0; \mathcal{N}(5, 1)$  (d) Plot of  $\mathcal{M}_{hamming}$  for  $\kappa = 1.0; \mathcal{N}(5, 1)$ .

## 6. Related Work

Some of the earliest works on active learning in a structured setting is the work in language parsing including (Thompson et al., 1999; Hwa, 2000), which utilize specific properties of the parsing algorithms to assign uncertainty values. There has also been work on active learning for hidden markov models (HMM), summarized by (Anderson & Moore, 2005), which is a learning algorithm for structured output with a specific set of sequential constraints. More directly related is the active learning work using conditional random fields (CRFs) (Culotta & McCallum, 2005), which can theoretically incorporate general constraints, basing selection on a probabilistic uncertainty metric. In this case, the complete labels are selected and the emphasis is on reducing the actual cost of labeling through a more sophisticated interaction with the expert.

## 7. Conclusions and Future Work

This work describes a margin-based active learning approach for structured output spaces. We first look

at the setting of querying complete labels, defining  $Q_{global}$  to be used in situations where the scoring function  $f(\mathbf{x}, \mathbf{y})$  is not decomposable or the data is expected to be exclusively globally learnable and define  $Q_{local(C)}$  to be used when the scoring function is decomposable and the data is expected to be locally learnable. We further demonstrate that in cases where the local classifications can be queried independently, the labeling effort is most drastically reduced using partial label queries. These propositions are also supported empirically on both synthetic data and the semantic role labeling (SRL) task. There appears to be many dimensions for future work including examining scenarios where subsets of the output variables are queried, providing a continuum between single and complete labels. Furthermore, developing a more realistic model of labeling cost along this continuum and looking at the performance of other margin-based learning algorithms within this framework would likely enable this work to be applied to a wider range of structured output applications.

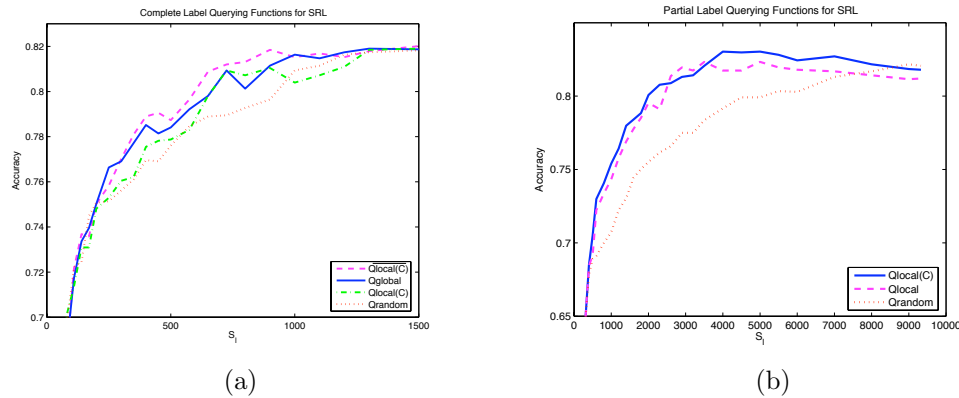


Figure 3. Experimental results for semantic role labeling (SRL) task. (a) Active learning curve for the complete label querying scenario (b) Active learning curve for the partial label querying scenario

## Acknowledgments

The authors would like to thank Ming-Wei Chang, Vasin Punyakanok, Alex Klementiev, Nick Rizzolo, and the reviewers for helpful comments and/or discussions regarding this research. This work has been partially funded by a grant from Motorola Labs and NSF grant ITR-IIS-0428472.

## References

- Anderson, B., & Moore, A. (2005). Active learning for hidden markov models: Objective functions and algorithms. *Proc. of the International Conference on Machine Learning (ICML)*.
- Carreras, X., & Màrquez, L. (2004). Introduction to the CoNLL-2004 shared tasks: Semantic role labeling. *Proc. of the Conference on Computational Natural Language Learning (CoNLL)*.
- Collins, M. (2002). Discriminative training methods for hidden markov models: Theory and experiments with perceptron algorithms. *Proc. of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*.
- Culotta, A., & McCallum, A. (2005). Reducing labeling effort for structured prediction tasks. *Proceedings of the National Conference on Artificial Intelligence (AAAI)*.
- Daumé III, H., & Marcu, D. (2005). Learning as search optimization: Approximate large margin methods for structured prediction. *Proc. of the International Conference on Machine Learning (ICML)*.
- Har-Peled, S., Roth, D., & Zimak, D. (2003). Constraint classification for multiclass classification and ranking. *The Conference on Advances in Neural Information Processing Systems (NIPS)* (pp. 785–792).
- Hwa, R. (2000). Sample selection for statistical grammar induction. *Proc. of the Conference on Empirical Methods in Natural Language Processing (EMNLP)* (pp. 45–52).
- Punyakanok, V., Roth, D., Yih, W., & Zimak, D. (2004). Semantic role labeling via integer linear programming inference. *Proc. the International Conference on Computational Linguistics (COLING)*.
- Punyakanok, V., Roth, D., Yih, W., & Zimak, D. (2005). Learning and inference over constrained output. *Proc. of the International Joint Conference on Artificial Intelligence (IJCAI)* (pp. 1124–1129).
- Thompson, C. A., Califf, M. E., & Mooney, R. J. (1999). Active learning for natural language parsing and information extraction. *Proc. of the International Conference on Machine Learning (ICML)* (pp. 406–414).
- Tong, S., & Koller, D. (2001). Support vector machine active learning with applications to text classification. *Journal of Machine Learning Research*, 2, 45–66.
- Tsochantaridis, I., Hofmann, T., Joachims, T., & Al-tun, Y. (2004). Support vector machine learning for interdependent and structured output spaces. *Proc. of the International Conference on Machine Learning (ICML)* (pp. 823–830).
- Yan, R., Yang, J., & Hauptmann, A. (2003). Automatically labeling video data using multi-class active learning. *Proc. of the International Conference on Computer Vision (ICCV)* (pp. 516–523).