# Margin-based Active Learning for Structured Output Spaces

Dan Roth and Kevin Small

Department of Computer Science
University of Illinois at Urbana-Champaign
201 N. Goodwin Avenue, Urbana, IL 61801, USA
{danr, ksmall}@uiuc.edu

**Abstract.** In many complex machine learning applications there is a need to learn multiple interdependent output variables, where knowledge of these interdependencies can be exploited to improve the global performance. Typically, these structured output scenarios are also characterized by a high cost associated with obtaining supervised training data, motivating the study of active learning 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 in such scenarios. Empirical results on both synthetic data and the semantic role labeling task demonstrate a significant reduction in the need for supervised training data when using the proposed methods.

## 1 Introduction

The successful application of machine learning algorithms to many domains is limited by the inability to obtain a sufficient amount of labeled training data due to practical constraints. The *active learning* paradigm offers one promising solution to this predicament 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. One particularly appropriate family of machine learning applications for active learning is the scenario where there are multiple learning problems such that there is a specified relationship between the output variables of the individual classifiers, described as *learning in structured output spaces*. In such situations, the target applications are generally more complex than single classification tasks and the cost for supervised training data is correspondingly higher.

There are many applications of learning in structured output spaces across numerous domains, including the semantic role labeling (SRL) task [1]. The goal for SRL is, given a sentence, to identify for each verb in the sentence which

constituents fill a semantic role and determine the type of the specified argument. For the example sentence, "I left my pearls to my daughter-in-law in my will," the desired output is

$[_{A0}$ I $][_V$ left $][_{A1}$ my pearls $][_{A2}$ to my daughter-in-law $][_{AM-LOC}$ in my will $]$,

where A0 represents the *leaver*, A1 represents the *item left*, A2 represents the *benefactor*, and AM-LOC is an adjunct indicating the location of the action. Examples of specifying structural relationships include declarative statements such as *every sentence must contain exactly one verb* or *no arguments can overlap*.

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. In the SRL example, these two protocols correspond to requiring the learner to request the labels for entire sentences during the instance selection process or single arguments, such as *my pearls*, respectively. We proceed to 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 modeling structured output classifiers, following the approach of incorporating output variable interdependencies directly into a discriminative learning model [2,3]. We then proceed by describing previous margin-based active learning approaches based on the output of linear classifiers [4,5].

### 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 $\mathbf{X} = (X_1, \ldots, X_{n_x}); X_t \in \mathbb{R}^{d_t}$ and $\mathbf{y} \in \mathcal{C}(\mathcal{Y}^{n_y})$ represent a structured assignment in the space of output variables $\mathbf{Y} = (Y_1, \ldots, Y_{n_y}); Y_t \in \{\omega_1, \ldots, \omega_{k_t}\}$. $\mathcal{C} : 2^{\mathcal{Y}^*} \to 2^{\mathcal{Y}^*}$ 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, $\mathcal{S} = \{(\mathbf{x}_1, \mathbf{y}_1), \ldots, (\mathbf{x}_m, \mathbf{y}_m)\}$ drawn i.i.d over $\mathcal{X}^{n_x} \times \mathcal{C}(\mathcal{Y}^{n_y})$ and returns a classifier $h : \mathcal{X}^{n_x} \to \mathcal{Y}^{n_y}$. This assignment generated by $h$ is based on a global scoring function $f : \mathcal{X}^{n_x} \times \mathcal{Y}^{n_y} \to \mathbb{R}$, 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

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

The output variable assignments are determined by a global scoring function $f(\mathbf{x}, \mathbf{y})$ which 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 $\hat{\mathbf{y}} = \text{argmax}_{\mathbf{y}' \in \mathcal{Y}^{n_y}} f(\mathbf{x}, \mathbf{y}')$. 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 $\hat{\mathbf{y}}_\mathcal{C}$ based on the global score $f(\mathbf{x}, \hat{\mathbf{y}}_\mathcal{C})$ 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 [6]. As active learning querying functions are designed to select instances with specific properties, we 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 $y' \in \mathcal{Y} \backslash y_t$. In this situation, $\hat{\mathbf{y}} = \hat{\mathbf{y}}_\mathcal{C} = \mathbf{y}$.*

**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 $\mathbf{y}' \in \mathcal{Y} \backslash \mathbf{y}$. We will refer to instances that are globally learnable, but not locally learnable as* **exclusively globally learnable** *in which case $\hat{\mathbf{y}} \neq \hat{\mathbf{y}}_\mathcal{C} = \mathbf{y}$.*

## 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 and provided to the learning algorithm to incrementally update its hypothesis. The most widely used active learning schemes utilize querying functions based on heuristics, often assigning a measure of certainty to predictions on $\mathcal{S}_u$ 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 $\hat{\mathbf{y}}_\mathcal{C} = \mathbf{y}$ 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 $\mathcal{L} : \mathbb{R} \to [0, \infty)$ 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 $\mathcal{Q}_{binary}$ [4], which makes direct use of the margin $\rho_{binary}(\mathbf{x}, y, f) = y \cdot f(\mathbf{x})$ by assuming the current classifier generally makes correct predictions on the training data and selecting those unlabeled examples with the smallest margin and thereby minimal certainty,

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

For multiclass classification, a widely accepted definition for multiclass margin is $\rho_{multiclass}(\mathbf{x}, \mathbf{y}, f) = f_y(\mathbf{x}) - f_{\dot{y}}(\mathbf{x})$ where $y$ represents the true label and $\dot{y} = \operatorname{argmax}_{y' \in \mathcal{Y} \setminus y} f_{y'}(\mathbf{x})$ corresponds to the highest activation value such that $\dot{y} \neq y$ [7]. Previous work on multiclass active learning [5] advocates a querying function closely related to this definition of multiclass margin where $\hat{y} = \operatorname{argmax}_{y' \in \mathcal{Y}} f_{y'}(\mathbf{x})$ represents the predicted label and $\tilde{y} = \operatorname{argmax}_{y' \in \mathcal{Y} \setminus \hat{y}} f_{y'}(\mathbf{x})$ represents the label corresponding to the second highest activation value,

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

## 3   Active Learning for Structured Output

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. Without loss of generality, we assume that $y_t$ represents a multiclass classification.

### 3.1   Querying Complete Labels

The task of 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. Following the margin-based approach for designing querying functions, a reasonable definition of margin for structured output spaces is $\rho_{global}(\mathbf{x}, \mathbf{y}, f) = f(\mathbf{x}, \mathbf{y}) - f(\mathbf{x}, \dot{\mathbf{y}}_{\mathcal{C}})$ where $\dot{\mathbf{y}}_{\mathcal{C}} = \operatorname{argmax}_{\mathbf{y}' \in \mathcal{C}(\mathcal{Y}^{n_y}) \setminus \mathbf{y}} f(\mathbf{x}, \mathbf{y}')$. The corresponding querying function for a structured learner that incorporates the constraints into the learning model is defined by

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

where $\tilde{\mathbf{y}}_{\mathcal{C}} = \operatorname{argmax}_{\mathbf{y}' \in \mathcal{C}(\mathcal{Y}^{n_y}) \setminus \hat{\mathbf{y}}_{\mathcal{C}}} f(\mathbf{x}, \mathbf{y}')$. It should be noted that $\mathcal{Q}_{global}$ does not require $f(\mathbf{x}, \mathbf{y})$ to be decomposable, thereby allowing usage with arbitrary loss functions. The only requirement is that the inference mechanism is capable of calculating $f(\mathbf{x}, \hat{\mathbf{y}}_{\mathcal{C}})$ and $f(\mathbf{x}, \tilde{\mathbf{y}}_{\mathcal{C}})$ for a given structured instance.

However, for many structured learning settings the scoring function and consequently the loss function is decomposable 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 [3]. As these findings are predicated on making concurrent local updates during learning, selecting structured examples that make as many local updates as possible may be desirable for such situations. This observation motivates a querying function that selects instances based on local predictions, resulting in the margin-based strategy of selecting examples with a small average local multiclass margin,

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

where $\hat{y}_{\mathcal{C},t} = \operatorname{argmax}_{y'_t \in \mathcal{C}(\mathcal{Y})} f_{y'_t}(\mathbf{x}, t)$ and $\tilde{y}_{\mathcal{C},t} = \operatorname{argmax}_{y'_t \in \mathcal{C}(\mathcal{Y}) \setminus \hat{y}_t} f_{y'_t}(\mathbf{x}, t)$.

### 3.2 Querying Partial Labels

We noted that $\mathcal{Q}_{global}$ makes no assumptions regarding decomposability of of the scoring function and $\mathcal{Q}_{\overline{local(\mathcal{C})}}$ requires only that the scoring function be decomposable in accordance with the output variables. We now examine active learning in settings where $f(\mathbf{x}, \mathbf{y})$ is decomposable and the local output variables can be queried independently, defined as querying partial labels. The intuitive advantage of querying partial labels is that we are no longer subject to cases where a structured instance has one output variable with a very informative label, but the other output variables of the same instance are minimally useful and yet add cost to the labeling effort. While this configuration is not immediately usable for applications with a scoring function not easily decomposable into local output variables that can be independently queried, we will see this approach is very beneficial in scenarios where such restrictions are possible.

Observing that querying partial labels requires requesting a single multiclass classification, the naive querying function for 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 [4]. A local classifier which either ignores or is ignorant of the structural constraints maintains a version space described by

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

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,

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

where $\dot{y}_{\mathcal{C},t} = \operatorname{argmax}_{y_t' \in \mathcal{C}(\mathcal{Y}) \setminus y_t} f_{y_t'}(\mathbf{x}, t)$. Therefore, if the learning algorithm enforces structural consistency within the learning model, we advocate also utilizing this information to augment $\mathcal{Q}_{local}$, resulting in the querying function

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

## 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 $\boldsymbol{\alpha} = (\boldsymbol{\alpha}^1, \ldots, \boldsymbol{\alpha}^{|\mathcal{Y}|})$ is a concatenation of the local $\boldsymbol{\alpha}^y$ vectors and $\Phi(\mathbf{x}, \mathbf{y}) = (\Phi^1(\mathbf{x}, \mathbf{y}), \ldots, \Phi^{|\mathcal{Y}|}(\mathbf{x}, \mathbf{y}))$ 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.

Margin-based active learning generally relies upon the use of support vector machines (SVM) [4, 5]. While there is existing work on SVM for structured output [8], 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 [3, 2] 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 which have been labeled. 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 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 $\hat{\mathbf{y}}_\mathcal{C}$, 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 $\mathcal{Q}_{global}$.

**Table 1.** Learning wth 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 $(\mathbf{x}, \mathbf{y}) \in \mathcal{S}$
        $\hat{\mathbf{y}}_\mathcal{C} \leftarrow \text{argmax}_{\mathbf{y} \in \mathcal{C}(\mathcal{Y}^{n_y})} \boldsymbol{\alpha} \cdot \Phi(\mathbf{x}, \mathbf{y})$
        foreach $t = 1, \ldots, n_y$ such that $(\mathbf{x}, y_t) \in \mathcal{S}_l$
          if $f_{y_t}(\mathbf{x}, t) - \gamma < f_{\hat{y}_{\mathcal{C},t}}(\mathbf{x}, t)$
            $\boldsymbol{\alpha}^{y_t} \leftarrow \boldsymbol{\alpha}^{y_t} + \Phi^{y_t}(\mathbf{x}, t)$
            $\boldsymbol{\alpha}^{\hat{y}_t} \leftarrow \alpha^{\hat{y}_t} - \Phi^{\hat{y}_t}(\mathbf{x}, t)$

---

OUTPUT: $\{f_y\}_{y \in \mathcal{Y}} \in \mathcal{H}$

### 4.1 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 suitable measurement for 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 $\mathcal{H}(h, \mathbf{x}) = \sum_{t=1;(\mathbf{x},y_t)\in\mathcal{S}_l}^{n_y} I[\![\hat{y}_{\mathcal{C},t} \neq y]\!]$ and the global loss $\mathcal{G}(h, \mathbf{x}) = I[\![\hat{\mathbf{y}}_{\mathcal{C}} \neq \mathbf{y}]\!]$ 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, for complete queries where a significant portion of the data is exclusively globally separable, $\mathcal{M}_{global}$ will be more strongly correlated with querying function performance.

## 5 Experiments

We demonstrate particular properties of the proposed querying functions by first running active learning simulations on synthetic data. We then verify practicality for actual applications by performing experiments on the SRL task.

### 5.1 Synthetic Data

Our synthetic structured output problem is comprised of five multiclass classifiers, $h_1, \ldots, h_5$, each having the output space $Y_t = \omega_1, \ldots, \omega_4$. 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. $\mathcal{C}_2 :$ At most one $h_t(\mathbf{x})$ can output $\omega_2$.
3. $\mathcal{C}_3 :$ For one or more $h_t(\mathbf{x})$ to output $\omega_3$, at least one $h_t(\mathbf{x})$ must output $\omega_1$.
4. $\mathcal{C}_4 : h_t(\mathbf{x})$ can output $\omega_4$ if and only if $h_{t-1}(\mathbf{x}) = \omega_1$ and $h_{t-2}(\mathbf{x}) = \omega_2$.

To generate the synthetic data, we first create four linear functions of the form $\mathbf{w}_i \cdot \mathbf{x} + b_i$ such that $\mathbf{w}_i \in [-1, 1]^{100}$ and $b_i \in [-1, 1]$ for each $h_t$. We then generate five local examples $\mathbf{x}_t \in \{0, 1\}^{100}$ where the normal distribution $\mathcal{N}(20, 5)$ determines the number of features assigned the value 1, distributed uniformly over the feature vector. Each vector is labeled according to the function $\mathrm{argmax}_{i=1,\ldots,k}[\mathbf{w}_i \cdot \mathbf{x} + b_i]$ resulting in the label vector $\mathbf{y}_{local} = (h_1(\mathbf{x}), \ldots, h_5(\mathbf{x}))$. We then run the inference procedure to obtain the final labeling $\mathbf{y}$ of the instance $\mathbf{x}$. 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 $\mathcal{N}(\mu, \sigma)$. 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 $\mathcal{Q}_{random}$, which selects arbitrary unlabeled instances at each step, and $\mathcal{Q}_{local(\mathcal{C})}$ 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 $|\mathcal{S}_l| = 2, 4, \ldots, 200$ and slowly increases the step size until $|\mathcal{S}_l| = 6000, 6100, \ldots, 8000$ and 5-fold cross validation is performed. The primary observation for the synthetic data set where $\kappa = 0.0$ is that $\mathcal{Q}_{\overline{local(\mathcal{C})}}$ 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 $\mathcal{Q}_{\overline{local(\mathcal{C})}}$. 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 less 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 on the synthetic data. We completed experiments with the two partial querying functions $\mathcal{Q}_{local}$ and $\mathcal{Q}_{local(\mathcal{C})}$ in addition to $\mathcal{Q}_{random}$ on three sets of data. The querying schedule starts by querying 10 partial labels at a time from $|\mathcal{S}_l| = 10, 20, \ldots, 2000$ and increases until the step size is $|\mathcal{S}_l| = 20000, 21000, \ldots, 40000$ and once again 5-fold cross validation is performed. The first synthetic data set is where $\kappa = 0.0$ and the data is completely locally separable. In this case, active learning for both $\mathcal{Q}_{local}$ and $\mathcal{Q}_{local(\mathcal{C})}$ perform better than $\mathcal{Q}_{random}$. Somewhat more surprising is the result that $\mathcal{Q}_{local(\mathcal{C})}$ performs noticeably better that $\mathcal{Q}_{local}$ 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 $\mathcal{Q}_{local(\mathcal{C})}$ 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 $\mathcal{Q}_{local(\mathcal{C})}$ and $\mathcal{Q}_{local}$ is most noticable. For this data set, we also plotted $\mathcal{M}_{Hamming}$ noting that this value is always greater for $\mathcal{Q}_{local(\mathcal{C})}$ 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.

## 5.2  Semantic Role Labeling

Finally, we also perform experiments on the SRL task as described in the CoNLL-2004 shared task [1]. We essentially follow the model described in [3] where linear classifiers $f_{A0}, f_{A1}, \ldots$ 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 $\hat{\mathbf{y}}_{\mathcal{C}}$ consistent with
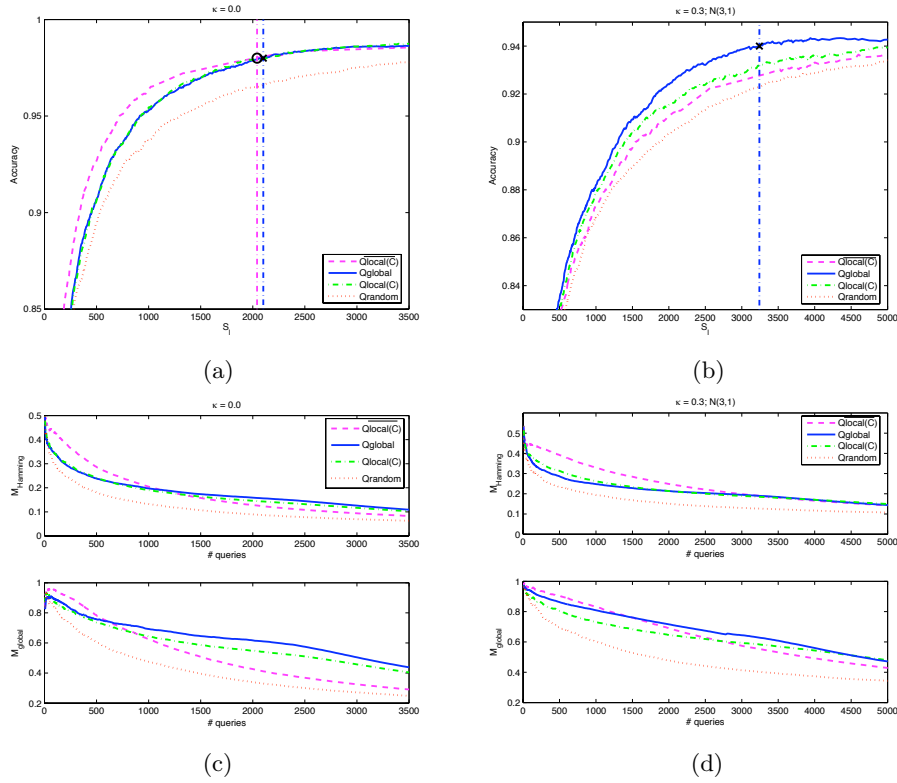
**Fig. 1.** Experimental results for the complete label querying problem, noting that the labeling effort is reduced between $25\% - 75\%$ depending on the particular situation. (a) Active learning curve for $\kappa = 0.0$ (b) Active learning curve for $\kappa = 0.3; \mathcal{N}(3, 1)$ (c) Plot of $\mathcal{M}_{hamming}$ and $\mathcal{M}_{global}$ for $\kappa = 0.0$ (d) Plot of $\mathcal{M}_{hamming}$ and $\mathcal{M}_{global}$ for $\kappa = 0.3; \mathcal{N}(3, 1)$

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 [9]. Figure 3 shows the emperical results for the SRL experiments. For querying complete labels, we start with a querying schedule of $|\mathcal{S}_l| = 50, 80, \ldots, 150$ and slowly increase the step size until ending with $|\mathcal{S}_l| = 1000, 1100, \ldots, 1500$. For
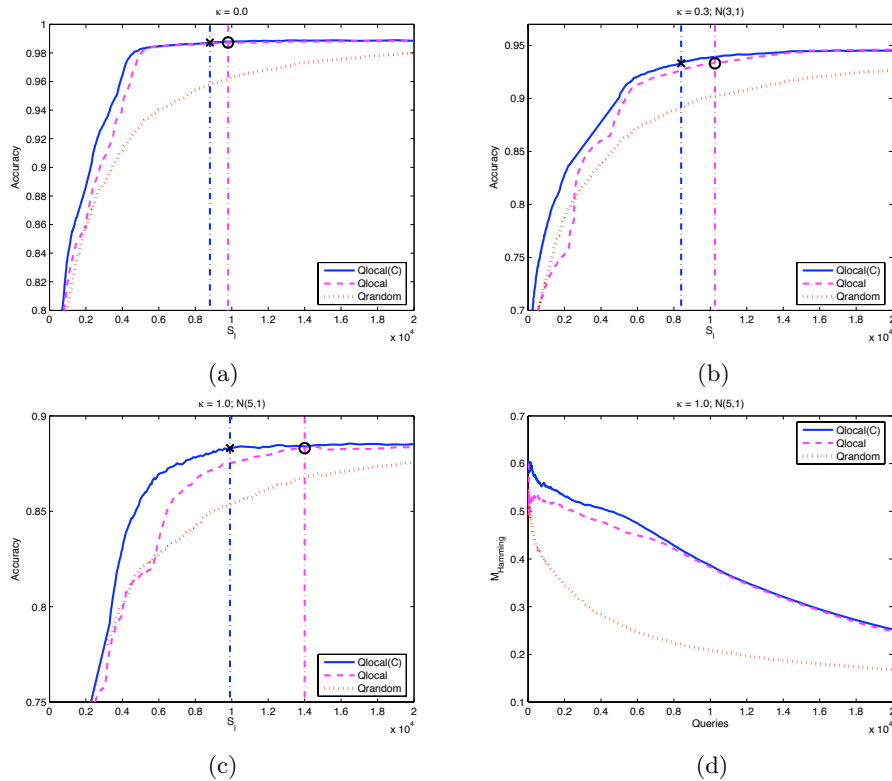
**Fig. 2.** Experimental results for the partial label querying problem, noting that the labeling effort is reduced between $65\% - 79\%$ depending of the particular situation. (a) Active learning curve for $\kappa = 0.0$ (b) Active learning curve for $\kappa = 0.3; \mathcal{N}(3,1)$ (c) Active learning curve for $\kappa = 1.0; \mathcal{N}(5,1)$ (d) Plot of $\mathcal{M}_{hamming}$ for $\kappa = 1.0; \mathcal{N}(5,1)$.

the complete labeling case, $\mathcal{Q}_{\overline{local(\mathcal{C})}}$ performs better than $\mathcal{Q}_{global}$, implying that the data is largely locally separable which is consistent with the findings of [3]. Furthermore, both functions perform better than $\mathcal{Q}_{random}$ with approximately a $35\%$ reduction in labeling effort. For partial labels, we used a querying schedule that starts at $|\mathcal{S}_l| = 100, 200, \ldots, 500$ and increases step size until ending at $|\mathcal{S}_l| = 6000, 7000, \ldots, 9327$. In this case, $\mathcal{Q}_{local(\mathcal{C})}$ performs better than $\mathcal{Q}_{local}$ and $\mathcal{Q}_{random}$, requiring only about half of the data to be labeled.

## 6 Related Work

Some of the earliest works on active learning in a structured setting is the work in language parsing including [10–12], which utilize specific properties of the parsing algorithms to assign uncertainty values to unlabeled instances. There has also been work on active learning for hidden markov models (HMM) [13,
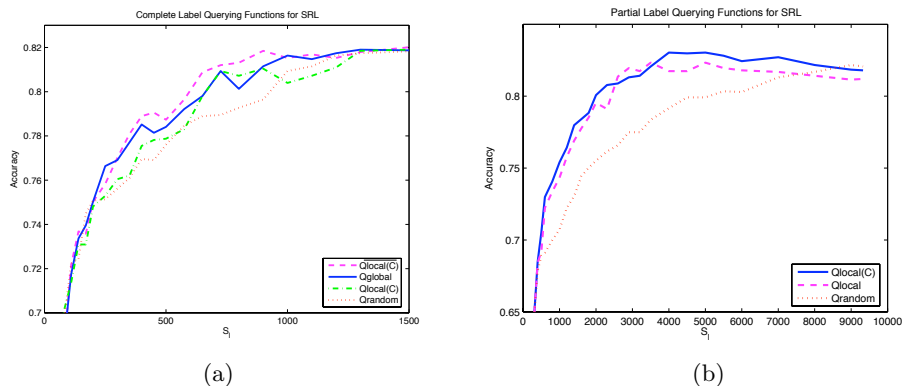
**Fig. 3.** Experimental results for SRL. (a) Active learning curve for the complete label querying scenario (b) Active learning curve for the partial label querying scenario

14], 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) [15], 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 $\mathcal{Q}_{global}$ 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 $\mathcal{Q}_{\overline{local(\mathcal{C})}}$ 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 with the querying function $\mathcal{Q}_{local(\mathcal{C})}$. 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.

## Acknowledgments

## References

1. Carreras, X., Màrquez, L.: Introduction to the CoNLL-2004 shared tasks: Semantic role labeling. In: Proc. of the Conference on Computational Natural Language Learning (CoNLL). (2004)
2. Collins, M.: Discriminative training methods for hidden markov models: Theory and experiments with perceptron algorithms. In: Proc. of the Conference on Empirical Methods in Natural Language Processing (EMNLP). (2002)
3. Punyakanok, V., Roth, D., Yih, W., Zimak, D.: Learning and inference over constrained output. In: Proc. of the International Joint Conference on Artificial Intelligence (IJCAI). (2005) 1124–1129
4. Tong, S., Koller, D.: Support vector machine active learning with applications to text classification. Journal of Machine Learning Research **2** (2001) 45–66
5. Yan, R., Yang, J., Hauptmann, A.: Automatically labeling video data using multiclass active learning. In: Proc. of the International Conference on Computer Vision (ICCV). (2003) 516–523
6. Daumé III, H., Marcu, D.: Learning as search optimization: Approximate large margin methods for structured prediction. In: Proc. of the International Conference on Machine Learning (ICML). (2005)
7. Har-Peled, S., Roth, D., Zimak, D.: Constraint classification for multiclass classification and ranking. In: The Conference on Advances in Neural Information Processing Systems (NIPS). (2003) 785–792
8. Tsochantaridis, I., Hofmann, T., Joachims, T., Altun, Y.: Support vector machine learning for interdependent and structured output spaces. In: Proc. of the International Conference on Machine Learning (ICML). (2004) 823–830
9. Punyakanok, V., Roth, D., Yih, W., Zimak, D.: Semantic role labeling via integer linear programming inference. In: Proc. the International Conference on Computational Linguistics (COLING). (2004)
10. Thompson, C.A., Califf, M.E., Mooney, R.J.: Active learning for natural language parsing and information extraction. In: Proc. of the International Conference on Machine Learning (ICML). (1999) 406–414
11. Hwa, R.: Sample selection for statistical grammar induction. In: Proc. of the Conference on Empirical Methods in Natural Language Processing (EMNLP). (2000)
12. Baldridge, J., Osbourne, M.: Active learning for HPSG parse selection. In: Proc. of the Annual Meeting of the North American Association of Computational Linguistics (NAACL). (2003) 17–24
13. Scheffer, T., Wrobel, S.: Active learning of partially hidden markov models. Lecture Notes in Computer Science **2189** (2001)
14. Anderson, B., Moore, A.: Active learning for hidden markov models: Objective functions and algorithms. In: Proc. of the International Conference on Machine Learning (ICML). (2005)
15. Culotta, A., McCallum, A.: Reducing labeling effort for stuctured prediction tasks. In: Proceedings of the National Conference on Artificial Intelligence (AAAI). (2005)