Learning via Inference over Structurally Constrained Output

Vasin Punyakanok Dan Roth Wen-tau Yih Dav Zimak
Department of Computer Science
University of Illinois at Urbana-Champaign
{punyakan,danr,yih,davzimak}@uiuc.edu

Abstract

We experimentally analyze learning structured output in a discriminative framework where values of the output variables are estimated by local classifiers. In this framework, complex dependencies among the output variables are captured by constraints that dictate how global labels can be inferred. We compare two strategies, *learning plus inference* and *inference based training*, by observing their behaviors in different conditions. We conclude that using inference during learning helps when the local classifiers are difficult to learn but requires more examples.

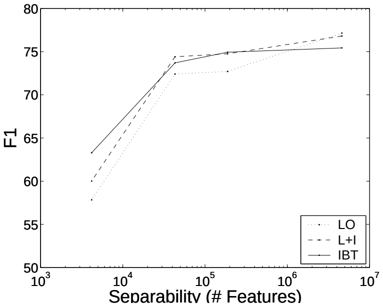
1 Introduction

Making decisions in real world problems involve assigning values to sets of variables where a complex and expressive structure can influence, or even dictate, what assignments are possible. For example, in the task of labeling part-of-speech tags to the words of a sentence, the prediction is governed by the constraints like “no three consecutive words are verbs.” Another example exists in scene interpretation tasks where predictions must respect constraints that could arise from the nature of the data or task specific conditions.

There exist three fundamentally different solutions to problem of learning classifiers over structured output. In the first, structure is ignored; local classifiers are learned and used to predict each output component separately. In the second, learning is decoupled from the task of maintaining structured output. Only after estimators are learned for each local output variable, are they used to produce global output consistent with the structural constraints. Discriminative HMM, conditional models [6, 5] and many dynamic programming based schemes used in the context of sequential predictions fall into the this category. The third class of solutions incorporates the dependencies among the variables into the learning process, and directly induces estimators that optimize the global performance measure. Traditionally these solutions were generative, however recent developments have produced discriminative models of this type, including conditional random fields[4], Collins' Perceptron-based learning scheme [2, 1] and Max-Margin Markov networks which allows incorporating Markovian assumptions among output variables [9].

In this paper, we look at the tradeoffs of using each of the above schemes. In the first, classifiers are learned independently (*learning only* (LO)), in the second, inference is used to maintain structural consistency only after learning (*learning plus inference* (L+I)), and finally inference is used while learning the parameters of the classifier (*inference based training* (IBT)). In semantic role labeling (SRL), it was observed [7] that when the local classification problems are easy to learn, L+I outperforms IBT. However, when using a reduced feature space where the problem was no longer (locally) separable, IBT could overcome the poor local classifications to yield accurate global classifications.

Figure 1: Results on the semantic-role labeling (SRL) problem. As the number of features increases, the difficulty of the local classification problem becomes easier, and the independently learned classifiers (LO) perform well, especially when inference is used after learning (L+I). Using inference during training (IBT) can aid performance when the learning problem is more difficult (few features).



In Section 3, we compare them using the online Perceptron algorithm applied in the three settings (see [2] for details). All three settings use the same linear representation, and L+I and IBT share the same decision function.

Despite the fact that IBT is a more powerful technique, in Section 5, we provide an experiment that shows how L+I can outperform IBT when there exist accurate local classifiers that do not depend on structure, or when there are too few examples to learn complex structural dependencies.

2 Background

Given an assignment $\mathbf{x} \in \mathcal{X}^{n_x}$ to a collection of input variables, $\mathbf{X} = (X_1, \ldots, X_{n_x})$, the structured classification problem involves identifying the "best" assignment $\mathbf{y} \in \mathcal{Y}^{n_y}$ to a collection of output variables $\mathbf{Y} = (Y_1, \ldots, Y_{n_y})$ that are consistent with a defined structure on $\mathbf{Y}$. This structure can be thought of as constraining the output space to a smaller space $\mathcal{C}(\mathcal{Y}^{n_y}) \subseteq \mathcal{Y}^{n_y}$, where $\mathcal{C} : 2^{\mathcal{Y}^*} \rightarrow 2^{\mathcal{Y}^*}$ constrains the output space to be structurally consistent.

In this paper, a structured output classifier is a function $h : \mathcal{X}^{n_x} \rightarrow \mathcal{Y}^{n_y}$, that uses a global scoring function, $f : \mathcal{X}^{n_x} \times \mathcal{Y}^{n_y} \rightarrow \mathbb{R}$ to assign scores to each possible example/label pair. Given input $\mathbf{x}$, it is hoped that, among consistent outputs, the correct output $\mathbf{y}$ achieves the highest score:

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

where n_x and n_y depend on the example at hand. In addition, we view the global scoring function as a composition of a set of local scoring functions $\{f_y(\mathbf{x}, t)\}_{y \in \mathcal{Y}}$, where $f_y : \mathcal{X}^{n_x} \times \{1, \ldots, n_y\} \rightarrow \mathbb{R}$. Each function represents the *score* or confidence that output variable Y_t takes value y:

$$f(\mathbf{x}, (y_1, \ldots, y_{n_y})) = \sum_{t=1}^{n_y} f_{y_t}(\mathbf{x}, t)$$

Inference is the task of determining an optimal assignment $\hat{\mathbf{y}}$ given an assignment $\mathbf{x}$. For sequential structure of constraints, polynomial-time algorithms such as Viterbi or CSCL [6] are typically used for efficient inference. For general structure of constraints, a generic search method (e.g., beam search) may be applied. Recently, integer programming has also been shown to be an effective inference approach in several NLP applications [8, 7].

In this paper, we consider classifiers with *linear representation*. Linear local classifiers can be written as linear functions, $f_y(\mathbf{x}, t) = \boldsymbol{\alpha}^y \cdot \Phi^y(\mathbf{x}, t)$, where $\boldsymbol{\alpha}^y \in \mathbb{R}^{d_y}$ is a weight vector and $\Phi^y(\mathbf{x}, t) \in \mathbb{R}^{d_y}$ is a feature vector. Then, it is easy to show that the global scoring function can be written in the familiar form $f(\mathbf{x}, \mathbf{y}) = \boldsymbol{\alpha} \cdot \Phi(\mathbf{x}, \mathbf{y})$, where $\Phi^y(\mathbf{x}, \mathbf{y}) = \sum_{t=1}^{n_y} \Phi^{y_t}(\mathbf{x}, t) I_{\{y_t = y\}}$ is an accumulation over all output variables of features occurring for class y, $\boldsymbol{\alpha} = (\boldsymbol{\alpha}^1, \ldots, \boldsymbol{\alpha}^{|\mathcal{Y}|})$ is concatenation of the $\boldsymbol{\alpha}^y$'s, and $\Phi(\mathbf{x}, \mathbf{y}) = (\Phi^1(\mathbf{x}, \mathbf{y}), \ldots, \Phi^{|\mathcal{Y}|}(\mathbf{x}, \mathbf{y}))$ is the concatenation of the $\Phi^y(\mathbf{x}, \mathbf{y})$'s. Then, the global classifier is

$$h(\mathbf{x}) = \hat{\mathbf{y}} = \operatorname*{argmax}_{\mathbf{y}' \in \mathcal{C}(\mathcal{Y}^{n_y})} \boldsymbol{\alpha} \cdot \Phi(\mathbf{x}, \mathbf{y}').$$

3 Learning

We present several ways to learn the scoring function parameters differing in whether or not the structure-based inference process is leveraged during training. Learning consists of choosing a function $h : \mathcal{X}^* \rightarrow \mathcal{Y}^*$ from some hypothesis space, $\mathcal{H}$. Typically, the data is supplied as a set $\mathbf{D} = \{(\mathbf{x}_1, \mathbf{y}_1), \ldots, (\mathbf{x}_m, \mathbf{y}_m)\}$ from a distribution $\mathrm{P}^{\mathcal{X},\mathcal{Y}}$ over $\mathcal{X}^* \times \mathcal{Y}^*$. While these concepts are very general, we focus on online learning of linear representations using a variant of the Perceptron algorithm (see [2]).

Learning Local Classifiers: Learning stand-alone local classifiers is perhaps the most straightforward setting. No knowledge of the inference procedure is used. Rather, for each example $(\mathbf{x}, \mathbf{y}) \in \mathbf{D}$, the learning algorithm must ensure that $f_{y_t}(\mathbf{x}, t) > f_{y'}(\mathbf{x}, t)$ for all $t = 1, \ldots, n_y$ and all $y' \neq y_t$. In Figure 2(a), an online Perceptron-style algorithm is presented where no global constraints are used. See [3] for details and Section 5 for experiments.

Learning Global Classifiers: We seek to train classifiers so they will produce the correct global classification. To this end, the key difference from the other approach is that here, feedback from the inference process determines which classifiers to train so that together, the classifiers and the inference procedure yield the desired result. As in [2, 1], we train according to a global criterion. The algorithm presented here is an online procedure, where at each step a subset of the classifiers are updated according to inference feedback. See Figure 2(b) for details of a Perceptron-like algorithm for learning with inference feedback (this algorithm was first proposed in [2]).

Note that in practice it is not uncommon that a problem is modeled in such a way that local classifiers also consider the output of others as part of their inputs. This sort of *interaction* can be incorporated directly to the algorithm for learning a global classifier as long as the appropriate inference process is used to ensure the consistency of the interaction among local classifiers in each example. However, learning local classifiers can be more tricky, and one must dynamically relabel subsets of classes during training. In order to remain focused on the problem of training with and without inference feedback, the experiments presented concern only the local classifiers without interaction.

```
Algorithm OnlineLocalLearning
    Input: $\mathbf{D}^{X,Y} \in \{\mathcal{X}^* \times \mathcal{Y}^*\}^m$
    Output: $\{f_y\}_{y\in\mathcal{Y}} \in \mathcal{H}$
    Initialize $\boldsymbol{\alpha}^y \in \mathbb{R}^{|\Phi^y|}$ for $y \in \mathcal{Y}$
    Repeat until converge
      for each $(\mathbf{x},\mathbf{y}) \in \mathbf{D}^{X,Y}$ do
        for $t = 1,\ldots,n_y$ do
          $\hat{y}_t = \mathrm{argmax}_y\, \boldsymbol{\alpha}^y \cdot \Phi^y(\mathbf{x},t)$
          if $\hat{y}_t \neq y_t$ then
            $\boldsymbol{\alpha}^{y_t} = \boldsymbol{\alpha}^{y_t} + \Phi^{y_t}(\mathbf{x},t)$
            $\boldsymbol{\alpha}^{\hat{y}_t} = \boldsymbol{\alpha}^{\hat{y}_t} - \Phi^{\hat{y}_t}(\mathbf{x},t)$
```

(a)

```
Algorithm OnlineGlobalLearning
    Input: $\mathbf{D}^{X,Y} \in \{\mathcal{X}^* \times \mathcal{Y}^*\}^m$
    Output: $\{f_y\}_{y\in\mathcal{Y}} \in \mathcal{H}$
    Initialize $\boldsymbol{\alpha} \in \mathbb{R}^{|\Phi|}$
    Repeat until converge
      for each $(\mathbf{x},\mathbf{y}) \in \mathbf{D}^{X,Y}$ do
        $\hat{\mathbf{y}} = \mathrm{argmax}_{\mathbf{y}}\, \boldsymbol{\alpha} \cdot \Phi(\mathbf{x},\mathbf{y})$
        if $\hat{\mathbf{y}} \neq \mathbf{y}$ then
          $\boldsymbol{\alpha} = \boldsymbol{\alpha} + \Phi(\mathbf{x},\mathbf{y}) - \Phi(\mathbf{x},\hat{\mathbf{y}})$
```

(b)

Figure 2: Algorithms for learning without (a) and with (b) inference feedback.

4 Conjectures

In this section, we investigate the relative performance of classifier systems learned with and without inference feedback. There are many competing factors. Initially, if the local classification problems are "easy", then it is likely that learning local classifiers only (LO) can yield the most accurate classifiers. However, an accurate model of the structural constraints could additionally increase performance (learning plus inference (L+I)). As the local problems become more difficult to learn, an accurate model of the structure becomes more important, and can perhaps overcome sub-optimal local classifiers. Despite the existence of a global solution, as the local classification problems become increasingly difficult, it is unlikely that structure based inference can fix poor classifiers learned locally. In this case, training with inference feedback (IBT) can be expected to converge given enough examples.

As a first attempt to formalize the difficulty of classification tasks, we define separability and learnability. A classifier, $f \in \mathcal{H}$, *globally separates* a data set $\mathbf{D}$ iff for all examples $(\mathbf{x},\mathbf{y}) \in \mathbf{D}$, $f(\mathbf{x},\mathbf{y}) > f(\mathbf{x},\mathbf{y}')$ for all $\mathbf{y}' \in \mathcal{Y}^{n_y} \setminus \mathbf{y}$ and *locally separates* $\mathbf{D}$ iff for all examples $(\mathbf{x},\mathbf{y}) \in \mathbf{D}$, $f_{y_t}(\mathbf{x},t) > f_y(\mathbf{x},t)$ for all $y \in \mathcal{Y} \setminus y_t$, and all $\mathbf{y}' \in \mathcal{Y}^{n_y} \setminus \mathbf{y}$. A learning $\mathcal{A}$ is a function from data sets to a $\mathcal{H}$. We say that $\mathbf{D}$ is *globally (locally) learnable* by $\mathcal{A}$ if there exists an $f \in \mathcal{H}$ such that f *globally (locally) separates* $\mathbf{D}$.

The following simple relationships exist between local and global learning: local separability implies global separability, but the inverse is not true; local separability implies local and global learnability; global separability implies global learnability, but not local learnability. As a result, it is clear that if there exist learning algorithms to learn global separations, then given enough examples, IBT will outperform L+I. However, learning with unlimited examples is usually not possible either because examples are expensive to label or because some learning algorithms simply do not scale well to many examples. When there are a fixed number of examples, L+I can outperform IBT.

Claim 1 *With a fixed number of examples:*

1. *If the local classification tasks are separable, then L+I outperforms IBT.*
2. *If the task is globally separable, but not locally separable then IBT outperforms L+I only with sufficient examples. This number correlates with the degree of the separability of the local classifiers.*

5 Experiments

We present experiments to show how the relative performance of learning plus inference (L+I) compares to inference based training (IBT) when the quality of the local classifiers and amount of training data varies.

In our experiment, each example $\mathbf{x}$ is a set of c points in d-dimensional real space, where $\mathbf{x} = (\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_c) \in \mathbb{R}^d \times \ldots \times \mathbb{R}^d$ and its label is a sequence of binary variable, $\mathbf{y} = (y_1, \ldots, y_c) \in \{0,1\}^c$, labeled according to:

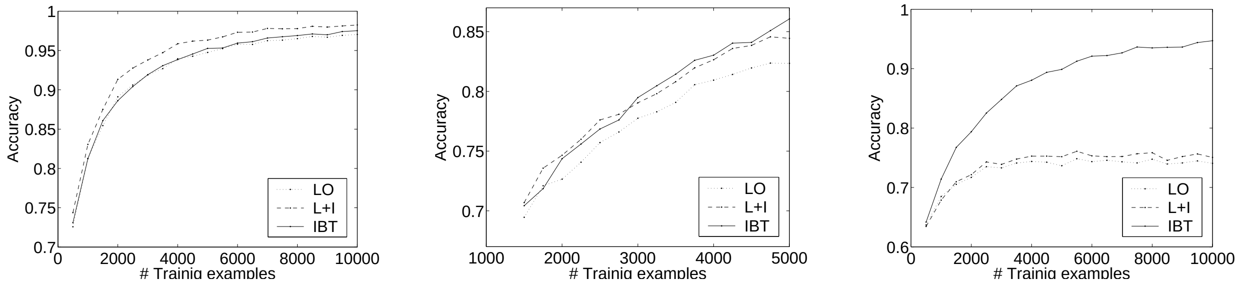
$$\mathbf{y} = (y_1, y_2, \ldots, y_c) = h(\mathbf{x}) = \operatorname*{argmax}_{\mathbf{y}\in\mathcal{C}(\mathcal{Y}^c)} \sum_i y_i f_i(\mathbf{x}_i) - (1-y_i) f_i(\mathbf{x}_i),$$

where $\mathcal{C}(\mathcal{Y}^c)$ is a subset of $\{0,1\}^c$ imposing a random constraint[1] on $\mathbf{y}$, and $f_i(\mathbf{x}_i) = \mathbf{w}_i \mathbf{x}_i + \theta_i$. Each f_i corresponds to a local classifier $y_i = g_i(\mathbf{x}_i) = I_{f_i(\mathbf{x}_i)>0}$. Clearly, the dataset generated from this hypothesis is globally linearly separable. To vary the difficulty of local classification, we generate examples with various degree of linear separability

[1] Among the total 2^c possible output labels, $\mathcal{C}(\cdot)$ fixes a random fraction as legitimate global labels.

of the local classifiers by controlling the fraction κ of the data where $h(\mathbf{x}) \neq g(\mathbf{x}) = (g_1(\mathbf{x}_1), \ldots, g_c(\mathbf{x}_c))$—examples whose labels, if generated by local classifiers independently, violate the constraints (i.e. $g(\mathbf{x}) \notin \mathcal{C}(\mathcal{Y}^c)$).

Figure 3 compares the performance of different learning strategies relative to the number of training examples used. In all experiments, $c = 5$, the true hypothesis is picked at random, and $\mathcal{C}(\mathcal{Y}^c)$ is a random subset with half of the size of $\mathcal{Y}^c$. Training is halted when a cycle complete with no errors, or 100 cycles is reached. The performance is averaged over 10 trials. Figure 3(a) shows the locally linearly separable case where L+I outperforms IBT. Figure 3(c) shows results for the case with the most difficult local classification tasks($\kappa = 1$) where IBT outperforms L+I. Figure 3(b) shows the case where data is not totally locally linearly separable($\kappa = 0.1$). In this case, L+I outperforms IBT when the number of training examples is small. In all cases, inference helps.



(a) $\kappa = 0, d = 100$ (b) $\kappa = 0.15, d = 300$ (c) $\kappa = 1, d = 100$

Figure 3: Comparison of different learning strategies in various degrees of difficulties of the local classifiers. $\kappa = 0$ implies locally linearly separability. Higher κ indicates harder local classification.

6 Conclusion

We studied the tradeoffs between three common learning schemes for structured outputs, i.e. learning without the knowledge about structure (LO), using inference only after learning (L+I), and learning with inference feedback (IBT). The experimental results on synthetic data confirm our main claims – first, when the local classification is linearly separable, L+I outperforms IBT, and second, as the local problems become more difficult and are no longer linearly separable, IBT outperforms L+I, but only with sufficient number of training examples. In the future, we will seek a similar comparison for the more general setting where nontrivial interaction between local classifiers is allowed, and thus, local separability does not imply global separability.

References

[1] X. Carreras and L. Màrquez. Online learning via global feedback for phrase recognition. In *Advances in Neural Information Processing Systems 15*, 2003.

[2] M. Collins. Discriminative training methods for hidden Markov models: Theory and experiments with perceptron algorithms. In *Proceedings of EMNLP 2002*, 2002.

[3] S. Har-Peled, D. Roth, and D. Zimak. Constraint classification: A new approach to multiclass classification and ranking. In *Advances in Neural Information Processing Systems 15*, 2003.

[4] J. Lafferty, A. McCallum, and F. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proc. ICML-01*, pages 282–289, 2001.

[5] A. McCallum, D. Freitag, and F. Pereira. Maximum entropy Markov models for information extraction and segmentation. In *Proceedings of ICML-00*, Stanford, CA, 2000.

[6] V. Punyakanok and D. Roth. The use of classifiers in sequential inference. In *Advances in Neural Information Processing Systems 13*, 2001.

[7] V. Punyakanok, D. Roth, W. Yih, and D. Zimak. Semantic role labeling via integer linear programming inference. In *Proceedings of COLING-04*, 2004.

[8] D. Roth and W. Yih. A linear programming formulation for global inference in natural language tasks. In *Proceedings of CoNLL-2004*, pages 1–8, 2004.

[9] B. Taskar, C. Guestrin, and D. Koller. Max-margin markov networks. In *Advances in Neural Information Processing Systems 16*, 2004.