SpeedMachines: Anytime Structured Prediction

Alexander Grubb AGRUBB@CMU.EDU
Daniel Munoz DMUNOZ@CS.CMU.EDU
J. Andrew Bagnell DBAGNELL@CS.CMU.EDU
Martial Hebert HEBERT@CS.CMU.EDU
School of Computer Science, Carnegie Mellon University, Pittsburgh, PA 15213 USA

Structured prediction plays a central role in machine learning applications from computational biology to computer vision. These models require significantly more computation than unstructured models, and, in many applications, algorithms may need to make predictions within a computational budget or in an anytime fashion. In this extended abstract we examine an anytime technique for learning structured predictors that, at training time, incorporates both structural elements and feature computation trade-offs that affect test-time inference. We apply our technique to the challenging problem of scene understanding in computer vision and demonstrate efficient and anytime predictions that gradually improve towards state-of-the-art classification performance as the allotted time increases.

1. Structured Prediction Setting

In the structured prediction setting, we are given inputs $x \in \mathcal{X}$ and associated structured outputs $y \in \mathcal{Y}$. These outputs are variable length vectors $(y_1, \dots, y_J)$. Furthermore, we assume that each element of the structured output, y_j, lies in some vector space $y_j \in \mathcal{Z}$.

The goal is to find a function $f : \mathcal{X} \to \mathcal{Y}$ which minimizes the risk $\mathcal{R}[f]$, typically a pointwise loss:

$$\mathcal{R}[f] = \mathbb{E}_{\mathcal{X}}[\sum_j l(f(x)_j)]. \tag{1}$$

We additionally assume we are given structural information about the outputs consisting of a set of groupings, or nodes, $N \in \mathcal{N}$ each of which is linked to some subset of the output, $j \in N$. For example, in the scene labeling domain we are given a set of input images $\mathcal{X}$, and output for each pixel j in a given image a vector $y_j \in \mathbb{R}^K$ containing the scores with respect to each of the K possible class labels. The structural information comes from segments, or superpixels, of the image which group together similar pixels.

2. Anytime Structured Prediction

To obtain predictions for varying test-time budgets, we use the SpeedBoost framework (Grubb & Bagnell, 2012), which is itself based on functional gradient boosting (Mason et al., 1999; Friedman, 2001), for learning an additive predictor $f = \sum_t \alpha_t h_t$. To obtain the functions $h : \mathcal{X} \to \mathcal{Y}$ we consider weak predictors with two parts: a selection function $h_{\mathrm{S}} : \mathcal{X} \times \mathcal{Y} \to 2^{\mathcal{N}}$ which takes in an input x and current prediction $\hat{y}$ and outputs a subset of structural nodes $\mathcal{N}$ to update, and a predictor $h_{\mathrm{P}} : \mathcal{N} \to \mathcal{Z}$ which updates the prediction for each element of the output corresponding to the selected region N.

These two components can be put together to create a weak structured predictor, h, which produces a structured output where element j of the output has prediction:

$$h(x)_j = \sum_{N \in h_{\mathrm{S}}(x,\hat{y})} \mathbf{1}(j \in N) h_{\mathrm{P}}(N). \tag{2}$$

We use the SpeedBoost approach to select the combined predictor with the best gain per unit cost

$$h_t, \alpha_t = \underset{h_{\mathrm{S}} \in \mathcal{H}_{\mathrm{S}}, h_{\mathrm{P}} \in \mathcal{H}_{\mathrm{P}}, \alpha \in \mathbb{R}}{\arg\max} \frac{\mathcal{R}\left[f_{t-1}\right] - \mathcal{R}\left[f_{t-1} + \alpha h\right]}{c(h)} \tag{3}$$

where h is defined from h_{S} and h_{P} as in Eq. 2.

Similar to previous work (Grubb & Bagnell, 2012) we make the above weak learner selection feasible by only evaluating the performance gain for a small set of weak learners near the gradient $\nabla \mathcal{R}[f_{t-1}]$. Given a fixed selection function h_{S}, and weak learning algorithm $\mathcal{A}$, and current predictor f, we can select a candidate predictor h_{P} using functional gradient projection as follows. The standard gradient projection (Friedman, 2001) for a predictor h of the form in (Eq. 2) can be

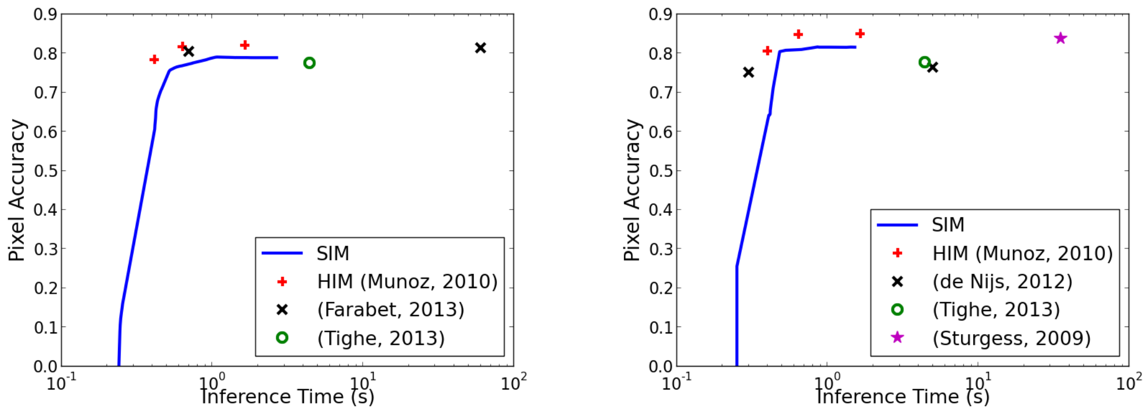

Figure 1. Average pixel classification accuracy for Stanford Background (left) and CamVid (right) datasets as a function of inference time.

reduced to finding h_P that minimizes

$$\operatorname*{arg\,min}_{h_\mathrm{P}\in\mathcal{H}_\mathrm{P}} \mathbb{E}_\mathcal{X}\left[\sum_{N\in h_\mathrm{S}(x,\hat{y})} |N| \left\|\mathbb{E}_{j\in N}\left[\nabla(x)_j\right] - h_\mathrm{P}(N)\right\|^2\right]. \tag{4}$$

2.1. Predictor Cost

We adopt a modified computational cost model of the feature extraction cost model of Xu *et al.* (2012). Like previous work, each feature incurs a fixed cost the first time it is used, but we additionally consider the case where groups of features share a common base feature that need only be computed once for everything in the group. Let $\phi \in \Phi$ be the set of features and $\gamma \in \Gamma$ be the set of feature groups, and c_ϕ and c_γ be the cost for computing derived feature ϕ and the base feature for group γ, respectively. Let $\Phi(f)$ be the set of features used by predictor f and $\Gamma(f)$ the set of its used groups. Given a current predictor f_{t-1}, we can express the cost of a predictor h_P as the cost of the newly computed features and groups:

$$c_\Gamma(h_\mathrm{P}) = \sum_{\gamma\in\Gamma(h_\mathrm{P})\setminus\Gamma(f_{t-1})} c_\gamma,$$

$$c_\Phi(h_\mathrm{P}) = \sum_{\phi\in\Phi(h_\mathrm{P})\setminus\Phi(f_{t-1})} c_\phi.$$

The total cost for a predictor h is then

$$c(h) = \epsilon_\mathrm{S} + \epsilon_\mathrm{P} + c_\Gamma(h_\mathrm{P}) + c_\Phi(h_\mathrm{P}), \tag{5}$$

where ϵ_S and ϵ_P are small fixed costs for evaluating a selection and prediction function, respectively.

In order to generate h_P with a variety of costs, we use a modified regression tree impurity function which uses a cost regularizer, as in (Xu et al., 2012):

$$\sum_i w_i \|y_i - h_\mathrm{P}(x_i)\|^2 + \lambda\left(c_\Gamma(h_\mathrm{P}) + c_\Phi(h_\mathrm{P})\right),$$

where λ regularizes the cost. Training regression trees with different values of λ and using the selection criteria in Eq. 3 enables StructuredSpeedBoost to consider predictors with a variety of costs and select the most beneficial.

3. Results

We now present preliminary results of our StructuredSpeedBoost method applied to the scene understanding domain, where the goal is to label every pixel in an image with the appropriate class label (e.g. car, building, etc). We compare directly to the state-of-the-art technique of Hierarchical Inference Machines (HIM) from Munoz *et al.* (2010). Our approach uses the same hierarchical structure, random forest weak learners and features, to obtain Speedy Inference Machines (SIM). We additionally compare to other scene understanding methods from the literature.

Fig. 1 shows the classification performance of SIM on two datasets. We compare to a number of versions of HIM which use different subsets of the available features, at the expense of significant manual tuning. We also compare to the reported performances of other techniques and stress that these timings are reported from different computing configurations. The single anytime predictor generated by our anytime structured prediction approach is competitive with all of the standalone models without requiring any of the manual analysis necessary to create the different fixed models.

References

Friedman, J. H. Greedy function approximation: A gradient boosting machine. *Annals of Statistics*, 29 (5), 2001.

Grubb, Alexander and Bagnell, J. Andrew. Speedboost: Anytime prediction with uniform near-optimality. In *AISTATS*, 2012.

Mason, L., Baxter, J., Bartlett, P. L., and Frean, M. Functional gradient techniques for combining hypotheses. In *Advances in Large Margin Classifiers*. MIT Press, 1999.

Munoz, Daniel, Bagnell, J. Andrew, and Hebert, Martial. Stacked hierarchical labeling. In *ECCV*, 2010.

Sturgess, Paul, Alahari, Karteek, Ladicky, Lubor, and Torr, Philip H. S. Combining appearance and structure from motion features for road scene understanding. In *BMVC*, 2009.

Tighe, Joseph and Lazebnik, Svetlana. Superparsing: Scalable nonparametric image parsing with superpixels. *IJCV*, 2013.

Xu, Zhixiang, Weinberger, Kilian Q., and Chapelle, Olivier. The greedy miser: Learning under test-time budgets. In *ICML*, 2012.