Entropy-based Latent Structured Output Prediction - Supplementary materials

Diane Bouchacourt
CentraleSupélec and INRIA Saclay
diane.bouchacourt@ecp.fr

Sebastian Nowozin
Microsoft Research
Sebastian.Nowozin@microsoft.com

M. Pawan Kumar
CentraleSupélec and INRIA Saclay
pawan.kumar@ecp.fr

1. Proofs

In this section we derive the proofs of all propositions in the main paper.

Proposition 1. *The AD entropy of the generalized distribution of $\boldsymbol{y}$ can be written as the sum of the negative log-likelihood of $\boldsymbol{y}$ and the AD entropy of the conditional distribution of the hidden variable given the output,*

$$H_{\alpha,\beta}(Q_{\boldsymbol{x}}^{\boldsymbol{y}};\boldsymbol{w}) = -\log P(\boldsymbol{y}|\boldsymbol{x},\boldsymbol{w}) + H_{\alpha,\beta}(P_{\boldsymbol{x}}^{\boldsymbol{y}};\boldsymbol{w}). \tag{1}$$

Proof. The AD entropy of the generalized distribution is

$$\begin{aligned}
H_{\alpha,\beta}(Q_{\boldsymbol{x}}^{\boldsymbol{y}};\boldsymbol{w}) &= \frac{1}{1-\alpha}\log\Big(\frac{\sum_h P(\boldsymbol{y},\boldsymbol{h}|\boldsymbol{x};\boldsymbol{w})^{\alpha+\beta-1}}{\sum_h P(\boldsymbol{y},\boldsymbol{h}|\boldsymbol{x};\boldsymbol{w})^{\beta}}\Big)\\
&= \frac{1}{1-\alpha}\log\Big(\frac{\sum_h P(\boldsymbol{h}|\boldsymbol{y},\boldsymbol{x};\boldsymbol{w})^{\alpha+\beta-1}P(\boldsymbol{y}|\boldsymbol{x};\boldsymbol{w})^{\alpha+\beta-1}}{\sum_h P(\boldsymbol{h}|\boldsymbol{y},\boldsymbol{x};\boldsymbol{w})^{\beta}P(\boldsymbol{y}|\boldsymbol{x};\boldsymbol{w})^{\beta}}\Big)\\
&= -\frac{\alpha-1}{1-\alpha}\log P(\boldsymbol{y}|\boldsymbol{x},\boldsymbol{w}) + \frac{1}{1-\alpha}\log\Big(\frac{\sum_h P(\boldsymbol{h}|\boldsymbol{y},\boldsymbol{x};\boldsymbol{w})^{\alpha+\beta-1}}{\sum_h P(\boldsymbol{h}|\boldsymbol{y},\boldsymbol{x};\boldsymbol{w})^{\beta}}\Big)\\
&= -\log P(\boldsymbol{y}|\boldsymbol{x},\boldsymbol{w}) + H_{\alpha,\beta}(P_{\boldsymbol{x}}^{\boldsymbol{y}};\boldsymbol{w})
\end{aligned}$$

□

The parameters of the model are learned by minimizing the objective function (2). We introduce regularization over the parameters of the model $\boldsymbol{w}$ to avoid overfitting the parameters to the training data.

$$\min_{\boldsymbol{w}}\ \frac{1}{2}||\boldsymbol{w}||^2 + \frac{C}{n}\sum_i\Big[\epsilon_y\log\sum_y\exp\frac{1}{\epsilon_y}\Big(\Delta(\boldsymbol{y}_i,\boldsymbol{y}) - \epsilon_h H_{\alpha,\beta}(Q_{\boldsymbol{x}_i}^{\boldsymbol{y}};\boldsymbol{w})\Big) + \epsilon_h H_{\alpha,\beta}(Q_{\boldsymbol{x}_i}^{\boldsymbol{y}_i};\boldsymbol{w})\Big]. \tag{2}$$

Proposition 2. *Objective* (2) *minimizes an upper bound on the loss $\Delta(\boldsymbol{y}_i,\boldsymbol{y}_i(\boldsymbol{w}))$ where $\boldsymbol{y}_i$ is the ground truth output of training example i and $\boldsymbol{y}_i(\boldsymbol{w})$ is the predicted output. This upper-bound is tightest when $\epsilon_y \to 0^+$.*

Proof. We have $\boldsymbol{y}_i(\boldsymbol{w}) = \operatorname*{argmin}_y H_{\alpha,\beta}(Q_{\boldsymbol{x}_i}^{\boldsymbol{y}_i};\boldsymbol{w})$ thus

$$\begin{aligned}
\Delta(\boldsymbol{y}_i,\boldsymbol{y}_i(\boldsymbol{w})) &\le \Delta(\boldsymbol{y}_i,\boldsymbol{y}_i(\boldsymbol{w})) - \epsilon_h H_{\alpha,\beta}(Q_{x_i}^{\boldsymbol{y}_i(\boldsymbol{w})};\boldsymbol{w}) + \epsilon_h H_{\alpha,\beta}(Q_{\boldsymbol{x}_i}^{\boldsymbol{y}_i};\boldsymbol{w})\\
&\le \epsilon_y\log\sum_y\exp\frac{1}{\epsilon_y}\Big(\Delta(\boldsymbol{y}_i,\boldsymbol{y}) - \epsilon_h H_{\alpha,\beta}(Q_{\boldsymbol{x}_i}^{\boldsymbol{y}_i};\boldsymbol{w})\Big) + \epsilon_h H_{\alpha,\beta}(Q_{\boldsymbol{x}_i}^{\boldsymbol{y}_i};\boldsymbol{w})
\end{aligned} \tag{3}$$

The first inequality holds by definition of $\boldsymbol{y}_i(\boldsymbol{w})$ and the second inequality holds because:

$$\Delta(\boldsymbol{y}_i,\boldsymbol{y}_i(\boldsymbol{w})) - \epsilon_h H_{\alpha,\beta}(Q_{x_i}^{\boldsymbol{y}_i(\boldsymbol{w})};\boldsymbol{w}) \le \max_y(\Delta(\boldsymbol{y}_i,\boldsymbol{y}) - \epsilon_h H_{\alpha,\beta}(Q_{\boldsymbol{x}_i}^{\boldsymbol{y}_i};\boldsymbol{w})) \tag{4}$$

For any set $P = (p_1, ..., p_n) \in \mathbb{R}^{+n}$ with $p_{max} = \max_{p \in P}$:

$$\begin{aligned}\log \sum_i \exp(p_i) &= p_{max} + \log \sum_{p_i} \exp(p_i - p_{max}) \\ &= p_{max} + \log(1 + \sum_{p_i p} \exp(p_i - p_{max})) \\ &\geq p_{max} + \log(1) \\ &= p_{max} \end{aligned} \tag{5}$$

since $\log$ is an increasing function and $\exp$ is always positive. Then:

$$\begin{aligned}\Delta(\boldsymbol{y}_i, \boldsymbol{y}_i(\boldsymbol{w})) - \epsilon_h H_{\alpha,\beta}(Q^{\boldsymbol{y}_i}_{\boldsymbol{x}_i}; \boldsymbol{w}) &\leq \max_y (\Delta(\boldsymbol{y}_i, \boldsymbol{y}) - \epsilon_h H_{\alpha,\beta}(Q^{\boldsymbol{y}}_{x_i}; \boldsymbol{w})) \\ &\leq \epsilon_y \log \sum_y \exp \frac{1}{\epsilon_y} \Big(\Delta(\boldsymbol{y}_i, \boldsymbol{y}) - \epsilon_h H_{\alpha,\beta}(Q^{\boldsymbol{y}_i}_{\boldsymbol{x}_i}; \boldsymbol{w}) \Big) \end{aligned} \tag{6}$$

□

Proposition 3. *The optimization problem* (2) *can be equivalently written as a difference of convex (DC) functions for any values of* $\alpha \geq 0, \beta \geq 0$ *using the following formulation,*

$$\min_{\boldsymbol{w}} \quad \frac{1}{2}||\boldsymbol{w}||^2 + \frac{C}{n}\epsilon_y \sum_i \Big[\log \sum_y \exp \frac{1}{\epsilon_y} \Big(\Delta(\boldsymbol{y}_i, \boldsymbol{y}) + F^+_{\alpha,\beta}(\boldsymbol{y}, \boldsymbol{w}) - F^-_{\alpha,\beta}(\boldsymbol{y}, \boldsymbol{w}) \Big) + G^+_{\alpha,\beta}(\boldsymbol{y}_i, \boldsymbol{w}) - G^-_{\alpha,\beta}(\boldsymbol{y}_i, \boldsymbol{w}) \Big], \tag{7}$$

where $F^+_{\alpha,\beta}(\boldsymbol{y}, \boldsymbol{w})$, $F^-_{\alpha,\beta}(\boldsymbol{y}, \boldsymbol{w})$, $G^+_{\alpha,\beta}(\boldsymbol{y}_i, \boldsymbol{w})$, *and* $G^-_{\alpha,\beta}(\boldsymbol{y}_i, \boldsymbol{w})$ *are convex.*

Proof. We can write (2) as:

$$\begin{aligned}&\min_{\boldsymbol{w}} \frac{1}{2}||\boldsymbol{w}||^2 + \frac{C}{n}\epsilon_y \sum_i \Big[\log \sum_y \exp \frac{1}{\epsilon_y} \Big(\Delta(\boldsymbol{y}_i, \boldsymbol{y}) - \epsilon_h \frac{1}{1-\alpha} \big(\log \sum_h P(\boldsymbol{y}, \boldsymbol{h}|\boldsymbol{x}_i; \boldsymbol{w})^{\alpha+\beta-1} - \log \sum_h P(\boldsymbol{y}, \boldsymbol{h}|\boldsymbol{x}_i; \boldsymbol{w})^{\beta} \big) \Big) \\ &\qquad + \epsilon_h \frac{1}{1-\alpha} \big(\log \sum_h P(\boldsymbol{y}_i, \boldsymbol{h}|\boldsymbol{x}_i, \boldsymbol{w})^{\alpha+\beta-1} - \log \sum_h P(\boldsymbol{y}_i, \boldsymbol{h}|\boldsymbol{x}_i, \boldsymbol{w})^{\beta} \big) \Big] \\ &= \min_{\boldsymbol{w}} \frac{1}{2}||\boldsymbol{w}||^2 + \frac{C}{n}\epsilon_y \sum_i \Big[\log \sum_y \exp \frac{1}{\epsilon_y} \Big(\Delta(\boldsymbol{y}_i, \boldsymbol{y}) - \epsilon_h \frac{1}{1-\alpha} \log \sum_h P(\boldsymbol{y}, \boldsymbol{h}|\boldsymbol{x}_i; \boldsymbol{w})^{\alpha+\beta-1} + \epsilon_h \frac{1}{1-\alpha} \log \sum_h P(\boldsymbol{y}, \boldsymbol{h}|\boldsymbol{x}_i; \boldsymbol{w})^{\beta} \big) \Big) \\ &\qquad + \epsilon_h \frac{1}{1-\alpha} \log \sum_h P(\boldsymbol{y}_i, \boldsymbol{h}|\boldsymbol{x}_i, \boldsymbol{w})^{\alpha+\beta-1} - \epsilon_h \frac{1}{1-\alpha} \log \sum_h P(\boldsymbol{y}_i, \boldsymbol{h}|\boldsymbol{x}_i, \boldsymbol{w})^{\beta} \Big] \end{aligned} \tag{8}$$

The log-sum-exp function summing over the output variable $\boldsymbol{y}$ is convex with respect to $\boldsymbol{w}$. Let's assume $\alpha > 1$, then $\dfrac{1}{1-\alpha} < 0$, and $-\dfrac{1}{1-\alpha} > 0$. In this case:

- $F^+_{\alpha,\beta}(\boldsymbol{y}, \boldsymbol{w}) = -\epsilon_h \dfrac{1}{1-\alpha} \log \sum_h P(\boldsymbol{y}, \boldsymbol{h}|\boldsymbol{x}_i; \boldsymbol{w})^{\alpha+\beta-1}$ is convex with respect to $\boldsymbol{w}$.
- $F^-_{\alpha,\beta}(\boldsymbol{y}, \boldsymbol{w}) = -\epsilon_h \dfrac{1}{1-\alpha} \log \sum_h P(\boldsymbol{y}, \boldsymbol{h}|\boldsymbol{x}_i; \boldsymbol{w})^{\beta}$ is convex with respect to $\boldsymbol{w}$.
- $G^+_{\alpha,\beta}(\boldsymbol{y}_i, \boldsymbol{w}) = -\epsilon_h \dfrac{1}{1-\alpha} \log \sum_h P(\boldsymbol{y}_i, \boldsymbol{h}|\boldsymbol{x}_i, \boldsymbol{w})^{\beta}$ is convex with respect to $\boldsymbol{w}$.
- $G^-_{\alpha,\beta}(\boldsymbol{y}_i, \boldsymbol{w}) = -\epsilon_h \dfrac{1}{1-\alpha} \log \sum_h P(\boldsymbol{y}_i, \boldsymbol{h}|\boldsymbol{x}_i, \boldsymbol{w})^{\alpha+\beta-1}$ is convex with respect to $\boldsymbol{w}$.

With the opposite results in the case $\alpha < 1$.

The log-sum-exp function summing over the output variable $\boldsymbol{y}$ is convex and non-decreasing with respect to $\boldsymbol{w}$, thus taking the log-sum-exp of a difference of convex is still a difference of convex [3, Corollary 4.3].

□

Unified Framework

$$\min_{\boldsymbol{w}} \frac{1}{2}||\boldsymbol{w}||^2 + \frac{C}{n}\sum_i \Big[\epsilon_y \log \sum_y \exp \frac{1}{\epsilon_y}\Big(\Delta(\boldsymbol{y}_i,\boldsymbol{y}) + \epsilon_h \log \sum_h \exp \frac{1}{\epsilon_h}\big(\boldsymbol{w}^T\phi(\boldsymbol{x_i},\boldsymbol{y},\boldsymbol{h})\big) - \epsilon_h H_{\alpha,\beta}(P^{\boldsymbol{y}}_{\boldsymbol{x}_i};\boldsymbol{w})\Big) - \epsilon_h \log \sum_h \exp \frac{1}{\epsilon_h}\big(\boldsymbol{w}^T\phi(\boldsymbol{x_i},\boldsymbol{y}_i,\boldsymbol{h})\big) + \epsilon_h H_{\alpha,\beta}(P^{\boldsymbol{y}_i}_{\boldsymbol{x}_i};\boldsymbol{w})\Big]$$

$\beta = 1, \alpha > 0, \epsilon_y \to 0, \epsilon_h = 1$

M3E [5]

$$\min_{\boldsymbol{w}} \frac{1}{2}||\boldsymbol{w}||^2 + \frac{C}{n}\sum_i \Big[\max_y \Big(\Delta(\boldsymbol{y}_i,\boldsymbol{y}) + \log \sum_h \exp\big(\boldsymbol{w}^T\phi(\boldsymbol{x_i},\boldsymbol{y},\boldsymbol{h})\big) - H_{\alpha,1}(P^{\boldsymbol{y}}_{\boldsymbol{x}_i};\boldsymbol{w})\Big) - \log \sum_h \exp\big(\boldsymbol{w}^T\phi(\boldsymbol{x_i},\boldsymbol{y}_i,\boldsymbol{h})\big) + H_{\alpha,1}(P^{\boldsymbol{y}_i}_{\boldsymbol{x}_i};\boldsymbol{w})\Big]$$

Optimization algorithm: Linearization of $H_{\alpha,1}(Q^{\boldsymbol{y}}_{\boldsymbol{x}_i};\boldsymbol{w}))$ + CP method

$\beta = 0, \alpha = 2, \epsilon_y = \epsilon_h \in (0,1)$

ϵ-framework [7]

$$\min_{\boldsymbol{w}} \frac{1}{2}||\boldsymbol{w}||^2 + \frac{C}{n}\sum_i \Big[\epsilon \log \sum_y \Big(\Delta(\boldsymbol{y}_i,\boldsymbol{y}) + \sum_h \exp \frac{1}{\epsilon}\big(\boldsymbol{w}^T\phi(\boldsymbol{x_i},\boldsymbol{y},\boldsymbol{h})\big)\Big) - \epsilon \log \sum_h \exp \frac{1}{\epsilon}\big(\boldsymbol{w}^T\phi(\boldsymbol{x_i},\boldsymbol{y}_i,\boldsymbol{h})\big)\Big]$$

Optimization : CCCP-like method + SGD

$\beta = 0, \alpha = 2, \epsilon_y \to 0, \epsilon_h = 1$

MSSVM [6]

$$\min_{\boldsymbol{w}} \frac{1}{2}||\boldsymbol{w}||^2 + \frac{C}{n}\sum_i \Big[\max_y \log \Big(\Delta(\boldsymbol{y}_i,\boldsymbol{y}) + \sum_h \exp\big(\boldsymbol{w}^T\phi(\boldsymbol{x_i},\boldsymbol{y},\boldsymbol{h})\big)\Big) - \log \sum_h \exp\big(\boldsymbol{w}^T\phi(\boldsymbol{x_i},\boldsymbol{y}_i,\boldsymbol{h})\big)\Big]$$

optimization : SGD or CCCP + SGD

$\alpha \to \infty$

$\epsilon_y = \epsilon_h \to 0^+$

LSSVM [2, 8]

$$\min_{\boldsymbol{w}} \frac{1}{2}||\boldsymbol{w}||^2 + \frac{C}{n}\sum_i \Big[\max_y \max_h \Big(\Delta(\boldsymbol{y}_i,\boldsymbol{y}) + \exp\big(\boldsymbol{w}^T\phi(\boldsymbol{x_i},\boldsymbol{y},\boldsymbol{h})\big)\Big) - \max_h \big(\boldsymbol{w}^T\phi(\boldsymbol{x_i},\boldsymbol{y}_i,\boldsymbol{h})\big)\Big]$$

Optimization : CCCP + CP method

Figure 1: *Equivalences between the UF objective and existing models' objective. For each model we quickly explain their specific optimization procedure. SGD means Stochastic Gradient Descent, CCCP stands for Concave Convex Procedure [9] and CP stands for Cutting Plane [4].*

2. Algorithmic details

In this section we detail the algorithmic procedure for training our Unified Framework (UF).

Algorithm 1: Algorithm for trainin UF

Data: $D = \{(\boldsymbol{x}_i, \boldsymbol{y}_i), i = 1...N\}$
Result: Model parameter $\boldsymbol{w}$
initialize $\boldsymbol{w} = \boldsymbol{w_0}, t = 0$;

$$\text{obj}(\boldsymbol{w}, \boldsymbol{w}_t) = \frac{1}{2}||\boldsymbol{w}||^2 + \frac{C}{n}\epsilon_y \sum_i \Big[\log \sum_{\boldsymbol{y}} \exp \frac{1}{\epsilon_y}\Big(\Delta(\boldsymbol{y}_i, \boldsymbol{y}) \quad + F^+_{\alpha,\beta}(\boldsymbol{y}, \boldsymbol{w}) - T^{F^-_{\alpha,\beta}}_{\boldsymbol{y},\boldsymbol{w}_t}(\boldsymbol{w})\Big) + G^+_{\alpha,\beta}(\boldsymbol{y}_i, \boldsymbol{w}) - T^{G^-_{\alpha,\beta}}_{\boldsymbol{y}_i,\boldsymbol{w}_t}(\boldsymbol{w})\Big] \tag{9}$$

while $t \leq T$ *and* $\delta_{obj} \geq C\lambda$ **do**
1 $\quad \boldsymbol{w}_{t+1} \leftarrow \underset{\boldsymbol{w}}{\text{argmin}}\, \text{obj}(\boldsymbol{w}, \boldsymbol{w}_t)$ by gradient descent.
2 $\quad \delta_{obj} \leftarrow \text{obj}(\boldsymbol{w}_t, \boldsymbol{w}_{t-1}) - \text{obj}(\boldsymbol{w}_{t+1}, \boldsymbol{w}_t)$
3 $\quad t \leftarrow t + 1$
4 **end**
5 return $\boldsymbol{w}$

During step 1 of Algorithm 1, we solve the convex optimization problem (10):

$$\boldsymbol{w}_{t+1} = \underset{\boldsymbol{w}}{\text{argmin}}\ \frac{1}{2}||\boldsymbol{w}||^2 + \frac{C}{n}\epsilon_y \sum_i \Big[\log \sum_y \exp \frac{1}{\epsilon_y}\Big(\Delta(\boldsymbol{y}_i, \boldsymbol{y}) + F^+_{\alpha,\beta}(\boldsymbol{y}, \boldsymbol{w}) - T^{F^-_{\alpha,\beta}}_{\boldsymbol{y},\boldsymbol{w}_t}(\boldsymbol{w})\Big) + G^+_{\alpha,\beta}(\boldsymbol{y}_i, \boldsymbol{w}) - T^{G^-_{\alpha,\beta}}_{\boldsymbol{y}_i,\boldsymbol{w}_t}(\boldsymbol{w})\Big]. \tag{10}$$

$F^-_{\alpha,\beta}(\boldsymbol{y}, \boldsymbol{w})$ and $G^-_{\alpha,\beta}(\boldsymbol{y}_i, \boldsymbol{w})$ are replaced by their first order Taylor expansion:

$$\begin{aligned} T^{F^-_{\alpha,\beta}}_{\boldsymbol{y},\boldsymbol{w}_t}(\boldsymbol{w}) &= F_{\alpha,\beta}(\boldsymbol{y}, \boldsymbol{w}_t)^- + (\boldsymbol{w} - \boldsymbol{w}_t)^T \nabla_{\boldsymbol{w}} F^-_{\alpha,\beta}(\boldsymbol{y}, \boldsymbol{w})|_{\boldsymbol{w}_t}, \\ T^{G^-_{\alpha,\beta}}_{\boldsymbol{y}_i,\boldsymbol{w}_t}(\boldsymbol{w}) &= G_{\alpha,\beta}(\boldsymbol{y}_i, \boldsymbol{w}_t)^- + (\boldsymbol{w} - \boldsymbol{w}_t)^T \nabla_{\boldsymbol{w}} G^-_{\alpha,\beta}(\boldsymbol{y}_i, \boldsymbol{w})|_{\boldsymbol{w}_t}. \end{aligned} \tag{11}$$

We denote by $\nabla_{\boldsymbol{w}} F^-_{\alpha,\beta}(\boldsymbol{y}, \boldsymbol{w})|_{\boldsymbol{w}_t}$ the gradient of $F^-_{\alpha,\beta}(\boldsymbol{y}, \boldsymbol{w})$ with respect to $\boldsymbol{w}$ estimated at $\boldsymbol{w}_t$ and similarly $\nabla_{\boldsymbol{w}} G^-_{\alpha,\beta}(\boldsymbol{y}_i, \boldsymbol{w})|_{\boldsymbol{w}_t}$.

We solve the optimization problem (10) by performing gradient descent as explained in Algorithm 2.

Algorithm 2: Gradient descent with line search for solving (10)

Data: $D = \{(\boldsymbol{x}_i, \boldsymbol{y}_i), i = 1...N\})$, current $\boldsymbol{w}_t$

Result: $\boldsymbol{w}_{t+1} = \underset{\boldsymbol{w}}{\operatorname{argmin}} \operatorname{obj}(\boldsymbol{w}, \boldsymbol{w}_t)$

stop = $0, l = 0$
$\boldsymbol{w}_{t+1} = \boldsymbol{w}_t$
$\operatorname{obj}_0 = \operatorname{obj}(\boldsymbol{w}_t, \boldsymbol{w}_t)$
while $l \leq L$ *and stop = 0* **do**
    LINE SEARCH procedure: returns $\boldsymbol{w}_{t+1}$
    $\text{stop}_l = 0, a = 0$
    $b = 0.5, c = 0.1$
    $\eta = \eta_0$
    while $a \leq A$ *and* $stop_l = 0$ **do**
        $\boldsymbol{w}_{\text{tent}} \leftarrow \boldsymbol{w}_t - \eta \nabla_{\boldsymbol{w}} \operatorname{obj}(\boldsymbol{w}, \boldsymbol{w}_t)|_{\boldsymbol{w_t}}$
        if $obj(\boldsymbol{w}_{tent}, \boldsymbol{w}_t) \leq obj_0 + c\eta \nabla_{\boldsymbol{w}} obj(\boldsymbol{w}, \boldsymbol{w}_t)|_{\boldsymbol{w_t}} \nabla_{\boldsymbol{w}} obj(\boldsymbol{w}, \boldsymbol{w}_t)|_{\boldsymbol{w_t}}$ **then**
            $\boldsymbol{w}_{t+1} \leftarrow \boldsymbol{w}_{\text{tent}}$
            $\text{stop}_l \leftarrow 1$
        else
            else
                $\eta \leftarrow b\eta$
            end
        end
        $a \leftarrow A + 1$
    end
    $\delta_{obj} = \operatorname{obj}_0 - \operatorname{obj}(\boldsymbol{w}_{\text{t+1}}, \boldsymbol{w}_t)$
    if $0 \leq \delta_{obj} < C\lambda$ **then**
        stop $\leftarrow 1$
    end
    $l \leftarrow l + 1$
end
return $\boldsymbol{w}_{t+1}$

3. Experiments details

3.1. Binary action classification

During our experiments, for each binary action classification tasks, we reweighed the positive samples by the scalar $\dfrac{|N|}{|P|}$ where N and P are the numbers of negatives and positives samples respectively. This is the weighted loss that we consider in our results analysis.

Table 1 shows the per class test loss mean on the 5 folds for the 10 actions of the "trainval" dataset of the PASCAL VOC 2011 [1] action classification dataset. As explained in our main paper, we see that that the performances of UF as a replication and the corresponding existing model are similar. All models perform equivalently except that the output by marginalizing the output and hidden variable as done by MSSVM is the less accurate criterion. We do not report p-values of left-tailed t-tests on the models' test loss values over the 5 folds since there no statistical significance of outperformance between pairs of models.

Tables 2 and 3 show that in most cases the set of UF parameters chosen by cross-validation boils down to a prediction criterion that maximizes over the output and hidden variables. Similarly, the best α chosen for the M3E models is of high value in that case M3E recovers LSSVM, and the ϵ-framework best parameter ϵ chosen is of small value that also approximates LSSVM.

In terms of computing requirements, all algorithms are comparable. Approximative computing time for all models with cross-validated parameters, averaged on all 10 classes, is of order $\sim$ 5-10 minutes in CPU time on an Intel Xeon X7542 core.

	jump-ing	playing instrument	phoning	riding bike	riding horse	reading	using computer	running	walking	taking photo
LSSVM [2, 8]	45 ± 2.0	57 ± 2.5	55.7 ± 0.82	36 ± 2.6	32 ± 3.0	53 ± 1.5	51 ± 1.0	33 ± 3.9	42 ± 1.9	67 ± 3.6
M3E $\alpha \to \infty$ [5]	45 ± 1.9	57 ± 2.5	54.6 ± 0.93	39 ± 4.6	33 ± 3.1	54 ± 1.6	49 ± 1.9	36 ± 3.4	43 ± 1.9	69 ± 3.6
M3E [5]	45 ± 2.6	56 ± 2.2	55.6 ± 0.73	35 ± 3.5	32 ± 1.4	53 ± 2.1	44 ± 2.1	33 ± 4.0	42 ± 1.8	67 ± 3.3
MSSVM[6]	51 ± 1.4	59 ± 3.9	57 ± 2.2	36 ± 3.5	35 ± 3.5	59 ± 2.4	51 ± 2.1	35 ± 4.2	46 ± 4.6	71 ± 2.6
ϵ-framework [7]	44 ± 2.3	57 ± 2.8	55 ± 1.2	36 ± 1.6	31 ± 2.7	53 ± 1.7	46 ± 2.2	31 ± 3.5	40 ± 1.6	66 ± 4.2
UF	43 ± 1.9	57 ± 2.3	53 ± 1.6	32 ± 1.7	31 ± 2.5	53 ± 1.7	48 ± 3.1	33 ± 3.9	41 ± 1.7	66 ± 3.5
UF $\sim$ LSSVM	44 ± 2.3	57 ± 2.5	54 ± 1.5	35 ± 2.9	35 ± 3.1	53 ± 1.4	50.6 ± 0.71	33 ± 3.7	41 ± 2.0	66 ± 3.5
UF $\sim$ M3E	44 ± 2.3	57 ± 2.6	54 ± 1.5	32 ± 1.8	32 ± 2.8	53 ± 1.4	48 ± 3.1	33 ± 3.9	41 ± 2.0	66 ± 3.5
UF $\sim$ MSSVM	51 ± 1.7	59 ± 3.9	58 ± 2.6	37 ± 3.0	37 ± 4.0	60 ± 2.0	49 ± 1.7	36 ± 3.2	44 ± 2.4	71 ± 2.6

Table 1: *Per class test loss mean on the 5 folds (in %) $\pm$ standard error of the mean (in %) with cross-validated parameters on the PASCAL VOC 11 dataset. The sign $\sim$ means that the parameters of the UF were set to replicate the existing model.*

	jumping	playing instrument	phoning	riding bike	riding horse
LSSVM [2, 8]	$C = 10$	$C = 1$	$C = 1$	$C = 10$	$C = 1$
M3E $\alpha \to \infty$ [5]	$C = 10$	$C = 1$	$C = 1$	$C = 10$	$C = 1$
M3E [5]	$C = 10, \alpha = 100$	$C = 1, \alpha = 10000$	$C = 1, \alpha = 10000$	$C = 10, \alpha = 0.1$	$C = 10, \alpha = 100$
MSSVM[6]	$C = 100$	$C = 1$	$C = 10$	$C = 10$	$C = 100$
ϵ-framework [7]	$C = 10, \epsilon = 0.01$	$C = 1, \epsilon = 0.1$	$C = 10, \epsilon = 1$	$C = 10, \epsilon = 0.1$	$C = 1, \epsilon = 0.001$
UF	$C = 10, \epsilon_h = 0.1, \alpha = 2, \beta = 1$	$C = 1, \epsilon_h = 0.1, \alpha = 2, \beta = 0.5$	$C = 1, \epsilon_h = 0.001, \alpha = 2, \beta = 1$	$C = 10, \epsilon_h = 0.1, \alpha = 0.01, \beta = 1$	$C = 1, \epsilon_h = 0.1, \alpha = 0.1, \beta = 1$
UF $\sim$ LSSVM	$C = 10$	$C = 1$	$C = 1$	$C = 10$	$C = 1$
UF $\sim$ M3E	$C = 10, \alpha = 100$	$C = 1, \alpha = 1000$	$C = 1, \alpha = 10000$	$C = 10, \alpha = 2$	$C = 1, \alpha = 100$
UF $\sim$ MSSVM	$C = 100$	$C = 1$	$C = 10$	$C = 10$	$C = 10$

Table 2: *Cross-validated parameters for each model on first five action classes. The sign $\sim$ means that the parameters of the UF were set to replicate the existing model.*

	reading	using computer	running	walking	taking photo
LSSVM [2, 8]	$C = 1$	$C = 1$	$C = 10$	$C = 1$	$C = 1$
M3E $\alpha \to \infty$ [5]	$C = 1$	$C = 1$	$C = 10$	$C = 10$	$C = 1$
M3E [5]	$C = 1, \alpha = 100$	$C = 10, \alpha = 0.1$	$C = 10, \alpha = 100$	$C = 1, \alpha = 10000$	$C = 1, \alpha = 100$
MSSVM[6]	$C = 10$	$C = 100$	$C = 100$	$C = 10$	$C = 1$
ϵ-framework [7]	$C = 1, \epsilon = 0.001$	$C = 100, \epsilon = 1$	$C = 10, \epsilon = 0.01$	$C = 10, \epsilon = 0.1$	$C = 1, \epsilon = 0.001$
UF	$C = 1, \epsilon_h = 0.1, \alpha = 2, \beta = 1$	$C = 10, \epsilon_h = 1, \alpha = 2, \beta = 1$	$C = 10, \epsilon_h = 0.01, \alpha = 0.01, \beta = 1$	$C = 1, \epsilon_h = 0.1, \alpha = 2, \beta = 1$	$C = 1, \epsilon_h = 0.001, \alpha = 0.1, \beta = 1$
UF $\sim$ LSSVM	$C = 1$	$C = 1$	$C = 10$	$C = 1$	$C = 1$
UF $\sim$ M3E	$C = 1, \alpha = 10000$	$C = 10, \alpha = 2$	$C = 10, \alpha = 100$	$C = 10, \alpha = 10000$	$C = 1, \alpha = 1000$
UF $\sim$ MSSVM	$C = 10$	$C = 100$	$C = 100$	$C = 100$	$C = 1$

Table 3: *Cross-validated parameters for each model on remaining five action classes. The sign $\sim$ means that the parameters of the UF were set to replicate the existing model.*

3.2. Multi-class gesture recognition

Table 4 reports the average loss on the test set for each model with respect to the noise level corrupting the dataset.

	σ = 0cm	σ = 1cm	σ = 5cm	σ = 8cm
LSSVM [2, 8]	11 ± 1.1	11 ± 1.1	16 ± 1.5	22.8 ± 0.92
M3E $\alpha \to \infty$ [5]	11 ± 1.2	10.5 ± 0.40	18.6 ± 0.52	22.8 ± 0.94
M3E [5]	8 ± 1.1	9 ± 1.0	11 ± 2.1	14 ± 1.5
MSSVM[6]	9 ± 1.3	9 ± 1.1	12 ± 1.1	17 ± 1.6
ϵ-framework [7]	10 ± 1.1	11.0 ± 0.74	15 ± 1.6	22.0 ± 0.65
UF	8 ± 1.1	8.6 ± 0.67	11.7 ± 0.85	15 ± 1.6
UF $\sim$ LSSVM	11 ± 1.1	11 ± 1.0	15 ± 1.5	22.8 ± 0.80
UF $\sim$ M3E	8 ± 1.1	9.1 ± 0.59	11.7 ± 0.85	15 ± 1.6
UF $\sim$ MSSVM	8 ± 1.1	9.0 ± 0.69	12 ± 1.0	15 ± 1.2

Table 4: *Test loss mean on the 5 folds (in %) $\pm$ standard error of the mean (in %) with cross-validated parameters on the MSRC-12 dataset, for different noise levels.*

Tables 5 to 8 show the p-values for the statistical left-tailed t-test on the models' test loss values over the 5 folds performed for all pair of models. We can see that when no noise is added to the data, M3E with $\alpha \to \infty$ is outperformed by M3E, the UF, the UF replicating M3E and the UF replicating MSSVM, with statistical significance at level 0.05. The UF replicating LSSVM is also outperformed by the UF, the UF replicating M3E and the UF replicating MSSVM. As the noise level increases, LSSVM and the ϵ-framework are also outperformed by M3E, MSSVM, the UF, the UF replicating M3E and the UF replicating MSSVM.

In terms of computing requirements, all algorithms are comparable. Approximative computing time for all models with cross-validated parameters, averaged on all noise levels, is of order $\sim$ 2-3 hours in CPU time on an Intel Xeon X7542 core.

	LSSVM [2, 8]	M3E $\alpha \to \infty$ [5]	M3E [5]	MSSVM [6]	ϵ-framework [7]	UF	UF $\sim$ LSSVM	UF $\sim$ M3E	UF $\sim$ MSSVM
LSSVM [2, 8]	0.0000	0.3750	0.9355	0.8544	0.7283	0.9494	0.4998	0.9494	0.9494
M3E $\alpha \to \infty$ [5]	0.6250	0.0000	0.9552	0.8984	0.8117	0.9644	0.6268	0.9644	0.9644
M3E [5]	0.0645	**0.0448**	0.0000	0.3474	0.1520	0.5682	0.0613	0.5682	0.5682
MSSVM [6]	0.1456	0.1016	0.6526	0.0000	0.2898	0.7059	0.1422	0.7059	0.7059
ϵ-framework [7]	0.2717	0.1883	0.8480	0.7102	0.0000	0.8796	0.2676	0.8796	0.8796
UF	0.0506	**0.0356**	0.4318	0.2941	0.1204	0.0000	**0.0479**	0.5000	0.5000
UF $\sim$ LSSVM	0.5002	0.3732	0.9387	0.8578	0.7324	0.9521	0.0000	0.9521	0.9521
UF $\sim$ M3E	0.0506	**0.0356**	0.4318	0.2941	0.1204	0.5000	**0.0479**	0.0000	0.5000
UF $\sim$ MSSVM	0.0506	**0.0356**	0.4318	0.2941	0.1204	0.5000	**0.0479**	0.5000	0.0000

Table 5: *p-values for σ = 0cm for statistical left-tailed t-test on the models' test loss values over the 5 folds. Value at indexes (i,j) of the table is the p-value for the left-tailed t-test with alternate hypothesis "model i outperforms model j". The sign $\sim$ means that the parameters of the UF were set to replicate the existing model.*

Table 9 shows cross-validated parameters for each model for each noise level. As explained in the main paper, the best parameters for the UF are never boiling down to either LSSVM, MSSVM or the ϵ-framework. In other words the best parameters combination $(\epsilon_h, \alpha, \beta)$ always take in account the AD entropy of the hidden variable. Moreover, except for σ=1cm, the UF recovers M3E models, that is $\epsilon_h = 1$ and $\beta = 1$.

	LSSVM [2, 8]	M3E $\alpha \to \infty$ [5]	M3E [5]	MSSVM [6]	ϵ-framework [7]	UF	UF $\sim$ LSSVM	UF $\sim$ M3E	UF $\sim$ MSSVM
LSSVM [2, 8]	0.0000	0.7739	0.9203	0.9127	0.6534	0.9683	0.4998	0.9443	0.9493
M3E $\alpha \to \infty$ [5]	0.2261	0.0000	0.8700	0.8527	0.3178	0.9779	0.2051	0.9538	0.9501
M3E [5]	0.0797	0.1300	0.0000	0.5015	0.0970	0.6693	0.0697	0.5009	0.5444
MSSVM [6]	0.0873	0.1473	0.4985	0.0000	0.1100	0.6574	0.0782	0.4991	0.5398
ϵ-framework [7]	0.3466	0.6822	0.9030	0.8900	0.0000	0.9772	0.3354	0.9530	0.9550
UF	**0.0317**	0.0221	0.3307	0.3426	0.0228	0.0000	**0.0236**	0.2760	0.3381
UF $\sim$ LSSVM	0.5002	0.7949	0.9303	0.9218	0.6646	0.9764	0.0000	0.9566	0.9602
UF $\sim$ M3E	0.0557	**0.0462**	0.4991	0.5009	**0.0470**	0.7240	**0.0434**	0.0000	0.5591
UF $\sim$ MSSVM	0.0507	**0.0499**	0.4556	0.4602	**0.0450**	0.6619	**0.0398**	0.4409	0.0000

Table 6: *p-values for σ = 1cm for statistical left-tailed t-test on the models' test loss values over the 5 folds. Value at indexes (i,j) of the table is the p-value for the left-tailed t-test with alternate hypothesis "model i outperforms model j". The sign $\sim$ means that the parameters of the UF were set to replicate the existing model.*

	LSSVM [2, 8]	M3E $\alpha \to \infty$ [5]	M3E [5]	MSSVM [6]	ϵ-framework [7]	UF	UF $\sim$ LSSVM	UF $\sim$ M3E	UF $\sim$ MSSVM
LSSVM [2, 8]	0.0000	0.0771	0.9376	0.9587	0.5495	0.9780	0.5255	0.9780	0.9466
M3E $\alpha \to \infty$ [5]	0.9229	0.0000	0.9875	0.9987	0.9325	0.9999	0.9320	0.9999	0.9991
M3E [5]	0.0624	**0.0125**	0.0000	0.3890	0.0751	0.4763	0.0673	0.4763	0.3238
MSSVM [6]	**0.0413**	**0.0013**	0.6110	0.0000	0.0559	0.6480	**0.0457**	0.6480	0.3948
ϵ-framework [7]	0.4505	0.0675	0.9249	0.9441	0.0000	0.9691	0.4752	0.9691	0.9280
UF	**0.0220**	**0.0001**	0.5237	0.3520	**0.0309**	0.0000	**0.0242**	0.5000	0.2408
UF $\sim$ LSSVM	0.4745	0.0680	0.9327	0.9543	0.5248	0.9758	0.0000	0.9758	0.9409
UF $\sim$ M3E	**0.0220**	**0.0001**	0.5237	0.3520	**0.0309**	0.5000	**0.0242**	0.0000	0.2408
UF $\sim$ MSSVM	0.0534	**0.0009**	0.6762	0.6052	0.0720	0.7592	0.0591	0.7592	0.0000

Table 7: *p-values for σ = 5cm for statistical left-tailed t-test on the models' test loss values over the 5 folds. Value at indexes (i,j) of the table is the p-value for the left-tailed t-test with alternate hypothesis "model i outperforms model j". The sign $\sim$ means that the parameters of the UF were set to replicate the existing model.*

	LSSVM [2, 8]	M3E $\alpha \to \infty$ [5]	M3E [5]	MSSVM [6]	ϵ-framework [7]	UF	UF $\sim$ LSSVM	UF $\sim$ M3E	UF $\sim$ MSSVM
LSSVM [2, 8]	0.0000	0.5026	0.9985	0.9885	0.7603	0.9972	0.5000	0.9972	0.9990
M3E $\alpha \to \infty$ [5]	0.4974	0.0000	0.9985	0.9883	0.7541	0.9972	0.4973	0.9972	0.9989
M3E [5]	**0.0015**	**0.0015**	0.0000	0.1308	**0.0028**	0.4034	**0.0015**	0.4034	0.2900
MSSVM [6]	**0.0115**	**0.0117**	0.8692	0.0000	**0.0205**	0.8100	**0.0113**	0.8100	0.7672
ϵ-framework [7]	0.2397	0.2459	0.9972	0.9795	0.0000	0.9949	0.2209	0.9949	0.9981
UF	**0.0028**	**0.0028**	0.5966	0.1900	**0.0051**	0.0000	**0.0029**	0.5000	0.3935
UF $\sim$ LSSVM	0.5000	0.5027	0.9985	0.9887	0.7791	0.9971	0.0000	0.9971	0.9990
UF $\sim$ M3E	**0.0028**	**0.0028**	0.5966	0.1900	**0.0051**	0.5000	**0.0029**	0.0000	0.3935
UF $\sim$ MSSVM	**0.0010**	**0.0011**	0.7100	0.2328	**0.0019**	0.6065	**0.0010**	0.6065	0.0000

Table 8: *p-values for σ = 8cm for statistical left-tailed t-test on the models' test loss values over the 5 folds. Value at indexes (i,j) of the table is the p-value for the left-tailed t-test with alternate hypothesis "model i outperforms model j". The sign $\sim$ means that the parameters of the UF were set to replicate the existing model.*

	σ = 0cm	σ = 1cm	σ = 5cm	σ = 8cm
LSSVM [2, 8]	$C = 100$	$C = 100$	$C = 100$	$C = 100$
M3E $\alpha \to \infty$ [5]	$C = 100$	$C = 100$	$C = 100$	$C = 100$
M3E [5]	$C = 1000, \alpha = 0.01$	$C = 1000, \alpha = 0.01$	$C = 1000, \alpha = 0.01$	$C = 100, \alpha = 2$
MSSVM[6]	$C = 100$	$C = 1000$	$C = 1000$	$C = 1000$
ϵ-framework [7]	$C = 100, \epsilon = 0.1$	$C = 100, \epsilon = 0.1$	$C = 100, \epsilon = 0.1$	$C = 100, \epsilon = 0.1$
UF	$C = 1000, \epsilon_h = 1, \alpha = 0.01, \beta = 1$	$C = 1000, \epsilon_h = 1, \alpha = 2, \beta = 0.5$	$C = 1000, \epsilon_h = 1, \alpha = 0.1, \beta = 1$	$C = 1000, \epsilon_h = 1, \alpha = 0.1, \beta = 1$
UF $\sim$ LSSVM	$C = 100$	$C = 100$	$C = 100$	$C = 100$
UF $\sim$ M3E	$C = 1000, \alpha = 0.01$	$C = 1000, \alpha = 0.01$	$C = 1000, \alpha = 0.1$	$C = 1000, \alpha = 0.1$
UF $\sim$ MSSVM	$C = 1000$	$C = 1000$	$C = 1000$	$C = 1000$

Table 9: *Cross-validated parameters for each model for each noise level. The sign $\sim$ means that the parameters of the UF were set to replicate the existing model.*

References

[1] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The pascal visual object classes (VOC) challenge. *International Journal of Computer Vision, Volume 88*, 2010.

[2] P. F. Felzenszwalb, D. A. McAllester, and D. Ramanan. A discriminatively trained, multiscale, deformable part model. In *CVPR*, 2008.

[3] R. Horst, P. M. Pardalos, and N. V. Thoai. *Introduction to Global Optimization*. Springer, 2000.

[4] T. Joachims, T. Finley, and C. J. Yu. Cutting-plane training of structural svms. *Machine Learning*, 77(1):27–59, 2009.

[5] K. Miller, M. P. Kumar, B. Packer, D. Goodman, and D. Koller. Max-margin min-entropy models. In *AISTATS*, 2012.

[6] W. Ping, Q. Liu, and A. Ihler. Marginal structured SVM with hidden variables. In *ICML*, 2014.

[7] A. G. Schwing, T. Hazan, M. Pollefeys, and R. Urtasun. Efficient structured prediction with latent variables for general graphical models. In *ICML*, 2012.

[8] C.-N. Yu and T. Joachims. Learning structural SVMs with latent variables. In *ICML*, 2009.

[9] A. L. Yuille and A. Rangarajan. The Concave-Convex Procedure (CCCP). In *NIPS*, 2002.