Efficient Learning Losses for Deep Hinge-Loss Markov Random Fields

Charles Dickens[1] **Connor Pryor**[1] **Eriq Augustine**[1] **Alon Albalak**[2] **Lise Getoor**[1]

[1]Computer Science and Engineering Dept., UC Santa Cruz, Santa Cruz, California, USA
[2]Department of Computer Science, UC Santa Barbara, Santa Barbara, California, USA

Abstract

In this work, we examine the learning process for Neural Probabilistic Soft Logic (NeuPSL) [Pryor et al., 2022]. NeuPSL is a novel neuro-symbolic (NeSy) framework that unites state-of-the-art symbolic reasoning with the low-level perception of deep neural networks to create a tractable probabilistic model that supports end-to-end learning via back-propagation. We investigate two common learning losses, *Energy-based* and *Structured Perceptron*. We provide formal definitions, and identify and propose principled fixes to degenerate solutions. We then perform an extensive evaluation over a canonical NeSy task.

1 INTRODUCTION

Neural Probabilistic Soft Logic (NeuPSL) [Pryor et al., 2022] is a recently introduced Neuro-symbolic (NeSy) framework that extends the expressive power of the *Probabilistic Soft Logic* (PSL) programming language [Bach et al., 2017] with neural models. NeuPSL instantiates a tractable class of graphical models, a *Deep Hinge-loss Markov random field* (*Deep HL-MRF*), to integrate low-level neural perception with symbolic reasoning. Deep-HL-MRFs admit log-concave density functions with a structure that supports highly-efficient maximum-a-posteriori (MAP) inference. Moreover, end-to-end training of both neural and graphical model parameters is possible via the standard back-propagation algorithm.

Pryor et al. (2022) introduces *neuro-symbolic energy-based models (NeSy-EBMs)*, a general family of EBMs that connects neural and symbolic components. NeSy-EBMs include NeuPSL and other prominent NeSy frameworks, including DeepProbLog [Manhaeve et al., 2018], and Logic Tensor Networks [Badreddine et al., 2022]. Training a NeSy-EBMs is the process of finding both symbolic and neural parameters that minimize an EBM learning objective.

In this work, we examine both the *energy loss*, used by Pryor et al. (2022), and the *structured perceptron loss* [Collins, 2002, LeCun et al., 1998] as training objectives for a NeuPSL model. We provide formal definitions and identify both theoretical and practical issues. Both learning losses require solving a subproblem to compute gradients for back-propagation. Moreover, both learning losses have degenerate solutions, which leads to minimizers of the learning problem that have low prediction performance.

In this work, we examine NeSy-EBM learning loss in the context of NeuPSL. 1) We describe the structured perceptron and energy learning losses, 2) We identify degenerate solutions for both losses and propose methods for removing them from the problem's feasible set while maintaining tractability, and 3) We analyze the runtime and performance of the learning losses on a canonical NeSy dataset.

2 NEUPSL

NeSy-EBM frameworks integrate neural architectures with encodings of symbolic relations to define an energy function with an explicit neural and symbolic interface. Specifically, input variables are organized into neural, $\mathbf{x}_{nn} \in \mathcal{X}^{nn}$, and symbolic $\mathbf{x} = [x_i]_{i=1}^{n_x} \in [0,1]^{n_x}$, vectors. Likewise, model parameters are organized into neural, $\mathbf{w}_{nn} \in \mathcal{W}^{nn}$, and symbolic $\mathbf{w} = [w_i]_{i=1}^{r} \in \mathbb{R}_+^r$, vectors. Then, one or more neural networks $\mathbf{g}_{nn} = [g_i]_{i=1}^{n_g}$, where each $g_i : \mathcal{X}_{nn} \to \mathbb{R}^{n_{g,i}}$, are integrated into a symbolic model as inputs of potential functions.

The symbolic model NeuPSL instantiates is a *Deep Hinge-Loss Markov Random Field* (Deep-HL-MRF), a tractable probabilistic graphical model. The potentials of a Deep-HL-MRF are hinge-loss functions over the neural output, the symbolic inputs $\mathbf{x}$, and a vector of symbolic output variables $\mathbf{y} = [y_i]_{i=1}^{n_y} \in [0,1]^{n_y}$. Each potential in NeuPSL represents the dissatisfaction of an arithmetic or logical rule in a PSL program (see Bach et al. (2017) for a complete

Submitted to the 5th Workshop on Tractable Probabilistic Modeling at UAI (TPM 2022). **To be used for reviewing only**.

description of the potential instantiation process).

Definition 1 (Deep Hinge-Loss Markov Random Field)**.** *Let $\mathbf{y} = [y_i]_{i=1}^{n_y}$ and $\mathbf{x} = [x_i]_{i=1}^{n_x}$ be vectors of real valued variables. Let $\mathbf{g}_{nn} = [g_{nn,i}]_{i=1}^{n_g}$ be functions with corresponding parameters $\mathbf{w}_{nn} = [\mathbf{w}_{nn,i}]_{i=1}^{n_g}$ and inputs $\mathbf{x}_{nn}$. A **deep hinge-loss potential** is a function of the form*

$$\phi(\mathbf{y}, \mathbf{x}, \mathbf{x}_{nn}, \mathbf{w}_{nn}) = \max(l(\mathbf{y}, \mathbf{x}, \mathbf{g}_{nn}(\mathbf{x}_{nn}, \mathbf{w}_{nn})), 0)^{\alpha}$$

where $\alpha \in \{1, 2\}$. Let $\mathcal{T} = [t_i]_{i=1}^{r}$ denote an ordered partition of a set of m deep hinge-loss potentials $\{\phi_1, \cdots, \phi_m\}$, then define

$$\Phi_i(\mathbf{y}, \mathbf{x}, \mathbf{x}_{nn}, \mathbf{w}_{nn}) := \sum_{j \in t_i} \phi_i(\mathbf{y}, \mathbf{x}, \mathbf{x}_{nn}, \mathbf{w}_{nn})$$
$$\mathbf{\Phi}(\mathbf{y}, \mathbf{x}, \mathbf{x}_{nn}, \mathbf{w}_{nn}) := [\Phi_i(\mathbf{y}, \mathbf{x}, \mathbf{x}_{nn}, \mathbf{w}_{nn})]_{i=1}^{r}$$

*Further, let $\mathbf{w}_{psl} = [w_i]_{i=1}^{r}$ be a vector of non-negative weights corresponding to the partition $\mathcal{T}$. Then, a **deep hinge-loss energy function** is*

$$E(\mathbf{y}, \mathbf{x}, \mathbf{x}_{nn}, \mathbf{w}_{nn}, \mathbf{w}_{psl}) = \mathbf{w}_{psl}^{T} \mathbf{\Phi}(\mathbf{y}, \mathbf{x}, \mathbf{x_{nn}}, \mathbf{w_{nn}})$$

*Further, let $\mathbf{c} = [c_i]_{i=1}^{q}$ be a vector of q linear constraints in standard form, defining the feasible set $\mathbf{\Omega} = \{\mathbf{y}, \mathbf{x} \,|\, c_i(\mathbf{y}, \mathbf{x}) \leq 0, \forall i\}$. A **deep hinge-loss Markov random field**, $\mathcal{P}$, with random variables $\mathbf{y}$ conditioned on $\mathbf{x}$ and $\mathbf{x}_{nn}$ is a probability density of the form*

$$P(\mathbf{y}|\mathbf{x}, \mathbf{x}_{nn}) = \begin{cases} \frac{\exp(-E(\mathbf{y}, \mathbf{x}, \mathbf{x}_{nn}, \mathbf{w}_{nn}, \mathbf{w}_{psl}))}{Z(\mathbf{x}, \mathbf{x}_{nn}, \mathbf{w}_{nn}, \mathbf{w}_{psl})} & (\mathbf{y}, \mathbf{x}) \in \mathbf{\Omega} \\ 0 & o.w. \end{cases}$$

$$Z(\mathbf{x}, \mathbf{x}_{nn}, \mathbf{w}_{nn}, \mathbf{w}_{psl}) = \int_{\Omega} \exp(-E(\mathbf{y}, \mathbf{x}, \mathbf{x}_{nn}, \mathbf{w}_{nn}, \mathbf{w}_{psl})) d\mathbf{y}$$

Inference in NeuPSL fully integrates neural and symbolic inference. Neural inference requires computing the output of the neural networks given the input $\mathbf{x}_{nn}$, i.e., computing $g_{nn,i}(\mathbf{x}_{nn}, \mathbf{w}_{nn,i})$ for all i, while symbolic inference minimizes the Deep-HL-MRF energy function over $\mathbf{y}$:

$$\mathbf{y}^{*} = \underset{\mathbf{y}|(\mathbf{y},\mathbf{x}) \in \mathbf{\Omega}}{\arg\min}\ E(\mathbf{y}, \mathbf{x}, \mathbf{x}_{nn}, \mathbf{w}_{nn}, \mathbf{w}_{psl}) \tag{1}$$

The energy function and constraints are convex in $\mathbf{y}$. Any scalable convex optimizer can be applied to solve (1).

3 LEARNING IN NESY-EBMS

Learning in NeSy-EBMs is the task of finding both neural parameters and symbolic parameters that minimize an EBM learning objective. Learning objectives are functionals mapping an energy function and a set of training examples $\mathcal{S} = \{(\mathbf{y}_i, \mathbf{x}_i, \mathbf{x}_{i,nn}) : i = 1, \cdots, P\}$ to a real-value. The energy function for NeuPSL is parameterized by the neural parameters $\mathbf{w}_{nn}$ and symbolic parameters $\mathbf{w}_{psl}$, so we can express the learning objective as a function of $\mathbf{w}_{nn}$, $\mathbf{w}_{psl}$, and $\mathcal{S}$: $\mathcal{L}(\mathbf{w}_{nn}, \mathbf{w}_{psl}, \mathcal{S})$. Learning objectives follow the standard empirical risk minimization framework and are separable over the training examples in $\mathcal{S}$ as a sum of per-sample loss functions $L_i(\mathbf{y}_i, \mathbf{x}_i, \mathbf{x}_{i,nn}, \mathbf{w}_{nn}, \mathbf{w}_{psl})$. Concisely, NeuPSL learning is the following minimization:

$$\underset{\mathbf{w}_{nn}, \mathbf{w}_{psl}}{\arg\min}\ \mathcal{L}(\mathbf{w}_{nn}, \mathbf{w}_{psl}, \mathcal{S})$$
$$= \underset{\mathbf{w}_{nn}, \mathbf{w}_{psl}}{\arg\min} \sum_{i=1}^{P} L_i(\mathbf{y}_i, \mathbf{x}_i, \mathbf{x}_{i,nn}, \mathbf{w}_{nn}, \mathbf{w}_{psl})$$

In the learning setting, variables $\mathbf{y}_i$ from the training set S are partitioned into vectors $\mathbf{y}_{i,t}$ and $\mathbf{z}_i$. The variables $\mathbf{y}_{i,t}$ represent variables for which there is a corresponding truth value, while $\mathbf{z}_i$ represent latent variables. Without loss of generality, we write $\mathbf{y}_i = (\mathbf{y}_{i,t}, \mathbf{z}_i)$.

There are multiple losses that one could motivate for optimizing the parameters of an EBM. The losses we present in this work use the following terms:

$$\mathbf{z}_i^{*} = \underset{\mathbf{z}|((\mathbf{y}_{i,t},\mathbf{z}),\mathbf{x}) \in \mathbf{\Omega}}{\arg\min}\ E((\mathbf{y}_{i,t}, \mathbf{z}), \mathbf{x}_i, \mathbf{x}_{i,nn}, \mathbf{w}_{nn}, \mathbf{w}_{psl})$$
$$\mathbf{y}_i^{*} = \underset{\mathbf{y}|(\mathbf{y},\mathbf{x}_i) \in \mathbf{\Omega}}{\arg\min}\ E(\mathbf{y}, \mathbf{x}_i, \mathbf{x}_{i,nn}, \mathbf{w}_{nn}, \mathbf{w}_{psl})$$

In words, $\mathbf{z}_i^{*}$ and $\mathbf{y}_i^{*}$ are the lowest energy states given $(\mathbf{y}_{i,t}, \mathbf{x}_i, \mathbf{x}_{i,nn})$ and $(\mathbf{x}, \mathbf{x}_{i,nn})$, respectively.

3.1 ENERGY LOSS

The simplest energy-based learning loss is the *energy loss*, $\mathcal{L}_{Energy}(\mathbf{w}_{nn}, \mathbf{w}_{psl}, S)$. Energy loss learning for NeuPSL was first presented in Pryor et al. (2022) and minimizes the energy of the MAP states of the Deep-HL-MRF given $(\mathbf{y}_{i,t}, \mathbf{x}_i)$, for all i.

$$\underset{\mathbf{w}_{nn}, \mathbf{w}_{psl}}{\arg\min}\ \mathcal{L}_{Energy}(\mathbf{w}_{nn}, \mathbf{w}_{psl}, S)$$
$$= \underset{\mathbf{w}_{nn}, \mathbf{w}_{psl}}{\arg\min} \sum_{i=1}^{P} E((\mathbf{y}_{i,t}, \mathbf{z}_i^{*}), \mathbf{x}_i, \mathbf{x}_{i,nn}, \mathbf{w}_{nn}, \mathbf{w}_{psl})$$

Notice that inference over the latent variables is necessary to compute the learning objective value. Furthermore, when strong convexity in each component of $\mathbf{z}$ is ensured via regularization, by Danskin (1966), the gradient of the energy loss with respect to $\mathbf{w}_{psl}$ is:

$$\nabla_{\mathbf{w}_{psl}} \mathcal{L}_{energy}(\mathbf{w}_{nn}, \mathbf{w}_{psl}, S)$$
$$= \sum_{i=1}^{P} \mathbf{\Phi}((\mathbf{y}_{i,t}, \mathbf{z}_i^{*}), \mathbf{x}_i, \mathbf{x}_{i,nn}, \mathbf{w}_{nn}, \mathbf{w}_{psl})$$

Then the per-sample energy loss partial derivative with respect to $\mathbf{w}_{nn}[j]$ at $\mathbf{y}_i, \mathbf{x}_i, \mathbf{x}_{i,nn}, \mathbf{w}_{psl}$ is:

$$\frac{\partial L_i(\mathbf{y}_i, \mathbf{x}_i, \mathbf{x}_{i,nn}, \mathbf{w}_{nn}, \mathbf{w}_{psl})}{\partial \mathbf{w}_{nn}[j]}$$
$$= \sum_{r=1}^{R} \mathbf{w}_{psl}[r] \sum_{q \in \tau_r} \frac{\partial \phi_q((\mathbf{y}_{i,t}, \mathbf{z}_i^{*}), \mathbf{x}_i, \mathbf{x}_{i,nn}, \mathbf{w}_{nn})}{\partial \mathbf{w}_{nn}[j]}$$

These gradients are sufficient to perform gradient descent via the backpropagation algorithm described in Section 3.4.

3.2 STRUCTURED PERCEPTRON LOSS

The *structured perceptron loss* [LeCun et al., 1998, Collins, 2002], $\mathcal{L}_{SP}(\mathbf{w}_{nn}, \mathbf{w}_{psl}, S)$, measures the energy difference between the true setting of the variables and the MAP state of the Deep-HL-MRF. Structured perceptron learning minimizes the difference in energies:

$$\begin{aligned}&\arg\min_{\mathbf{w}_{nn},\mathbf{w}_{psl}} \mathcal{L}_{SP}(\mathbf{w}_{nn}, \mathbf{w}_{psl}, S)\\ =&\arg\min_{\mathbf{w}_{nn},\mathbf{w}_{psl}} \sum_{i=1}^{P} E((\mathbf{y}_{i,t}, \mathbf{z}_i^*), \mathbf{x}_i, \mathbf{x}_{i,nn}, \mathbf{w}_{psl}, \mathbf{w}_{nn})\\ &\qquad - E(\mathbf{y}_i^*, \mathbf{x}_i, \mathbf{x}_{i,nn}, \mathbf{w}_{psl}, \mathbf{w}_{nn})\end{aligned}$$

In this loss, inference over both the latent variables, $\mathbf{z}$, and over the complete set of variables, $\mathbf{y}$, is necessary. With regularization, the HL-MRF energy function is strongly convex in all components of $\mathbf{y}$. Thus, the gradient of the structured perceptron loss with respect to $\mathbf{w}$ is:

$$\begin{aligned}&\nabla_{\mathbf{w}}\mathcal{L}_{SP}(\mathbf{w}_{nn}, \mathbf{w}_{psl}, S)\\ =&\sum_{i=1}^{P} \mathbf{\Phi}((\mathbf{y}_{i,t}, \mathbf{z}_i^*), \mathbf{x}_i, \mathbf{x}_{i,nn}, \mathbf{w}_{nn}) - \mathbf{\Phi}(\mathbf{y}_i^*, \mathbf{x}_i, \mathbf{x}_{i,nn}, \mathbf{w}_{nn})\end{aligned}$$

3.3 DEGENERATE SOLUTIONS

Pryor et al. (2022) shows there are two degenerate solutions of the energy learning loss for NeuPSL and propose methods for overcoming them. We show that the same degenerate solutions and methods for overcoming them apply to the structured percetpron loss.

First, note that the symbolic parameters are constrained to be non-negative real numbers. Furthermore, every symbolic potential is of the form:

$$\phi_i(\mathbf{y}, \mathbf{x}, \mathbf{x}_{nn}, \mathbf{w}_{nn}) = \max(l_i(\mathbf{y}, \mathbf{x}, \mathbf{g}_{nn}(\mathbf{x}_{nn}, \mathbf{w}_{nn})), 0)^{\alpha}$$

so we have that $\phi_i(\mathbf{y}, \mathbf{x}, \mathbf{x}_{nn}, \mathbf{w}_{nn}) \geq 0$ for all settings of the variables $\mathbf{y}, \mathbf{x}, \mathbf{x}_{nn}, \mathbf{w}_{nn}$. Thus, $\Phi_i(\mathbf{y}, \mathbf{x}, \mathbf{x}_{nn}, \mathbf{w}_{nn}) := \sum_{j\in t_i} \phi_i(\mathbf{y}, \mathbf{x}, \mathbf{x}_{nn}, \mathbf{w}_{nn}) \geq 0$ and $\mathbf{\Phi}(\mathbf{y}, \mathbf{x}, \mathbf{x_{nn}}, \mathbf{w_{nn}}) := [\Phi_i(\mathbf{y}, \mathbf{x}, \mathbf{x}_{nn}, \mathbf{w}_{nn})]_{i=1}^{r} \succeq \mathbf{0}$. We therefore have that

$$\begin{aligned}&\mathcal{L}_{Energy}(\mathbf{w}_{nn}, \mathbf{w}_{psl}, \mathcal{S})\\ =&\sum_{i=1}^{P} \min_{\mathbf{z}|((\mathbf{y}_{i,t},\mathbf{z}),\mathbf{x})\in\mathbf{\Omega}} \mathbf{w}_{psl}^T \mathbf{\Phi}((\mathbf{y}_{i,t}, \mathbf{z}), \mathbf{x}_i, \mathbf{x}_{i,nn}, \mathbf{w}_{nn}) \geq 0\end{aligned}$$

In fact, $\mathcal{L}_{Energy}(\mathbf{w}_{nn}, \mathbf{w}_{psl}, \mathcal{S}) = 0$, i.e., is minimized when $\mathbf{w}_{psl} = \mathbf{0}$.

Similarly, as $\mathbf{y}^*$ is a MAP state of the energy function, it is a minimizer of the energy function, hence

$$\begin{aligned}&\mathcal{L}_{SP}(\mathbf{w}_{nn}, \mathbf{w}_{psl}, S)\\ =&\sum_{i=1}^{P} \mathbf{w}_{psl}^T \Big(\mathbf{\Phi}((\mathbf{y}_{i,t}, \mathbf{z}_i^*), \mathbf{x}_i, \mathbf{x}_{i,nn}, \mathbf{w}_{nn})\\ &\qquad - \mathbf{\Phi}(\mathbf{y}_i^*, \mathbf{x}_i, \mathbf{x}_{i,nn}, \mathbf{w}_{nn}) \Big) \geq 0\end{aligned}$$

and $\mathcal{L}_{SP}(\mathbf{w}_{nn}, \mathbf{w}_{psl}, \mathcal{S}) = 0$ when $\mathbf{w}_{psl} = \mathbf{0}$. The $\mathbf{w}_{psl} = \mathbf{0}$ configuration results in a *collapsed* energy function, a function that assigns all points $\mathbf{y} \in \mathcal{Y}$ to the same energy.

One way to eliminate the $\mathbf{w}_{psl} = \mathbf{0}$ degenerate solution is by leveraging the invariance of HL-MRF MAP inference to the scale of the symbolic parameters, as shown by Srinivasan et al. (2021). Formally, for all configurations $\mathbf{w}_{psl}$ and scalars $\tilde{c} \in \mathbb{R}_+$,

$$\begin{aligned}&\arg\min_{\mathbf{y}|(\mathbf{y},\mathbf{x})\in\mathbf{\Omega}} E(\mathbf{y}, \mathbf{x}, \mathbf{x}_{nn}, \mathbf{w}_{nn}, \mathbf{w}_{psl})\\ =&\arg\min_{\mathbf{y}|(\mathbf{y},\mathbf{x})\in\mathbf{\Omega}} E(\mathbf{y}, \mathbf{x}, \mathbf{x}_{nn}, \mathbf{w}_{nn}, \tilde{c}\cdot\mathbf{w}_{psl})\end{aligned}$$

For this reason, it is possible to constrain the search space of the symbolic parameters to the unit simplex, $\Delta^r = \{\mathbf{w} \in \mathbb{R}_+^r \big| \|\mathbf{w}\|_1 = 1\}$, without inhibiting the expressivity of the model when the HL-MRF is exclusively used to obtain MAP inference predictions. Adding the simplex constraint discussed in the previous section makes the degenerate solution $\mathbf{w}_{psl} = \mathbf{0}$ infeasible.

The simplex constraint and concavity of the energy loss in the symbolic parameters raises an additional degenerate solution. Precisely, a solution to the problem must exist at corner points of the simplex Δ^r.

Lemma 1. *The structured perceptron and energy loss functions, $\mathcal{L}_{Energy}(\mathbf{w}_{nn}, \mathbf{w}_{psl}, \mathcal{S})$, and $\mathcal{L}_{SP}(\mathbf{w}_{nn}, \mathbf{w}_{psl}, \mathcal{S})$ are concave in $\mathbf{w}_{psl}$.*

Proof. For all i

$$\begin{aligned}&E((\mathbf{y}_{i,t}, \mathbf{z}_i^*), \mathbf{x}_i, \mathbf{x}_{i,nn}, \mathbf{w}_{nn}, \mathbf{w}_{psl})\\ =&\inf_{\mathbf{z}|((\mathbf{y}_{i,t},\mathbf{z}),\mathbf{x})\in\mathbf{\Omega}} \mathbf{w}_{psl}^T \Phi((\mathbf{y}_{i,t}, \mathbf{z}), \mathbf{x}_i, \mathbf{x}_{i,nn}, \mathbf{w}_{nn})\end{aligned}$$

Similarly,

$$\begin{aligned}&E((\mathbf{y}_{i,t}, \mathbf{z}_i^*), \mathbf{x}_i, \mathbf{x}_{i,nn}, \mathbf{w}_{nn}, \mathbf{w}_{psl})\\ &\quad - E(\mathbf{y}_i^*, \mathbf{x}_i, \mathbf{x}_{i,nn}, \mathbf{w}_{nn}, \mathbf{w}_{psl})\\ =&\inf_{\mathbf{z}|((\mathbf{y}_{i,t},\mathbf{z}),\mathbf{x})\in\mathbf{\Omega}} \inf_{\mathbf{y}|(\mathbf{y},\mathbf{x})\in\mathbf{\Omega}}\\ &\quad \mathbf{w}_{psl}^T(\Phi((\mathbf{y}_{i,t}, \mathbf{z}), \mathbf{x}_i, \mathbf{x}_{i,nn}, \mathbf{w}_{nn}) - \Phi(\mathbf{y}, \mathbf{x}_i, \mathbf{x}_{i,nn}, \mathbf{w}_{nn}))\end{aligned}$$

Hence, $\mathcal{L}_{Energy}(\mathbf{w}_{nn}, \mathbf{w}_{psl}, \mathcal{S})$, and $\mathcal{L}_{SP}(\mathbf{w}_{nn}, \mathbf{w}_{psl}, \mathcal{S})$ are both pointwise infimums of a set of affine functions of $\mathbf{w}_{psl}$ and are therefore concave [Boyd and Vandenberghe, 2004]. Therefore, $\mathcal{L}_{Energy}$ and $\mathcal{L}_{SP}$ are sums of concave functions of $\mathbf{w}_{psl}$ and are concave. □

Further, Δ^r is a polyhedron described by the set: $\Delta^r = \{\mathbf{w}_{psl} \mid \mathbf{1}^T\mathbf{w}_{psl} = 1\}$ and is therefore a convex set. A concave function is globally minimized over a polyhedron at one of the vertices, this can be shown by definition of concavity. In this case, we can find a solution to the energy minimization problem by comparing the objective value at each point of the simplex, i.e., setting one of the symbolic parameter components to 1 and the remaining parameters to 0. This solution is however undesirable; we want each of

Model	Learning Method	Noise (%)	Accuracy	Runtime (sec)
NeuPSL	Energy	10	77.1 ± 2.5	120.3 ± 0.7
	Energy	25	75.3 ± 4.6	120.3 ± 0.5
	Energy	50	70.4 ± 4.2	121.2 ± 0.7
	Structured Perceptron	10	71.2 ± 3.9	266.5 ± 2.0
	Structured Perceptron	25	72.0 ± 4.6	281.3 ± 2.4
	Structured Perceptron	50	75.1 ± 3.8	289.9 ± 2.5
Independent Baseline	-	10	81.8 ± 2.5	-
	-	25	59.0 ± 2.7	-
	-	50	30.1 ± 2.5	-

Table 1: Accuracy and runtime over varying noise in the local model.

the symbolic relations corresponding to the parameters to be represented and have influence over the model predictions. For this reason, we propose the use of the negative logarithm as a regularizer to break the concavity of the objective and give the simplex corner solutions infinitely high energy. With negative log regularization and simplex constraints, energy loss parameter learning and structured perceptron parameter learning are:

$$\min_{\mathbf{w}_{nn}\in\mathcal{W}_{nn},\mathbf{w}_{psl}\in\Delta^r} \mathcal{L}_{Energy}(\mathbf{w}_{nn},\mathbf{w}_{psl},\mathcal{S}) - \sum_{i=1}^{r}\log_b(\mathbf{w}_{psl}[i])$$

$$\min_{\mathbf{w}_{nn}\in\mathcal{W}_{nn},\mathbf{w}_{psl}\in\Delta^r} \mathcal{L}_{SP}(\mathbf{w}_{nn},\mathbf{w}_{psl},\mathcal{S}) - \sum_{i=1}^{r}\log_b(\mathbf{w}_{psl}[i])$$

3.4 MIRROR DESCENT

Minimization of the learning losses with respect to the symbolic parameters constrained to the unit simplex is achievable via normalized exponentiated gradient descent [Kivinen and Warmuth, 1997, Shalev-Shwartz, 2012]. The minimization over neural parameters is achievable via standard gradient descent. With an initial step size parameter $\eta > 0$, the parameter updates are

$$\mathbf{w}_{nn}^{k+1} = \mathbf{w}_{nn}^{k} + \eta\nabla_{w_{nn}}\mathcal{L}(\mathbf{w}_{nn}^{k},\mathbf{w}_{psl}^{k},S)$$

$$\mathbf{w}_{psl}^{k+1}[i] = \frac{\mathbf{w}_{psl}^{k}[i]\exp\{-\eta\frac{\partial\mathcal{L}(\mathbf{w}_{nn}^{k},\mathbf{w}_{psl}^{k},S)}{\partial\mathbf{w}_{psl}^{k}[i]}\}}{\sum_{j=1}^{r}\exp\{-\frac{\partial\mathcal{L}(\mathbf{w}_{nn}^{k},\mathbf{w}_{psl}^{k},S)}{\partial\mathbf{w}_{psl}^{k}[j]}\}}, \quad \forall i = 1,\cdots,r$$

This update ensures that at every iterate the symbolic parameter $\mathbf{w}_{psl}$ satisfies simplex constraints. This proposed minimization algorithm and convergence behavior can be analyzed with the mirror descent framework [Shalev-Shwartz, 2012].

4 EMPIRICAL EVALUATION

We evaluate the performance and runtime of the energy and structured perceptron learning losses on **MNIST-Add1** [Manhaeve et al., 2018]. This task extends the classic MNIST image classification problem [LeCun et al., 1998] by constructing addition equations using MNIST images and requiring classification to be performed using only their sum as a label. For example, a **MNIST-Add1** addition is $(\boxed{3} + \boxed{5} = 8)$. Given 300 randomly selected MNIST images, we create 150 **MNIST-Add1** additions. We emphasize that individual MNIST images do not have labels, only the resulting sum. All results are averaged over ten splits and evaluated over 500 randomly sampled MNIST images. The baseline symbolic and neural models are taken from [Pryor et al., 2022]. In order to study the effect of noise, we introduce a (noisy) digit classifier for each MNIST image.

Table 1 shows the average accuracy and standard deviation over ten splits on varying amounts of predictor noise. *Independent Baseline* is the expected accuracy using only the local predictor, i.e., the probability of the predictor labeling of both numbers in the addition correctly. Energy and structured perceptron outperform the *Independent Baseline* with 25% and 50% noise, while both perform slightly worse with 10% noise. The energy loss outperforms structured perceptron in accuracy in most settings, while always providing over a two times speed up in time.

While optimizing directly for accuracy is likely to improve predictive performance, accuracy is non-differentiable and hence less tractable. The differentiablity of both energy and structured perceptron in training makes the models tractable. For future work we intend to explore the trade off in performance versus time for more complex tractable losses.

5 CONCLUSION

In this paper, we explore two efficient learning losses for Deep HL-MRFs, energy and structured perceptron. There are many avenues for future work including exploration of additional efficient energy learning losses, looking into losses that specifically focus on the metric being evaluated over, and the exploration of additional application domains to further understand when each loss is appropriate. Each of these directions is likely to provide insights into creating useful and accurate tractable models.

References

Stephen Bach, Matthias Broecheler, Bert Huang, and Lise Getoor. Hinge-loss Markov random fields and probabilistic soft logic. *JMLR*, 18(1):1–67, 2017.

Samy Badreddine, Artur d'Avila Garcez, Luciano Serafini, and Michael Spranger. Logic tensor networks. *AI*, 303 (4):103649, 2022.

Stephen Boyd and Lieven Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.

Michael Collins. Discriminative training methods for hidden Markov models: Theory and experiments with perceptron algorithms. In *Empirical Methods in Natural Language Processing (EMNLP)*. ACM, 2002.

John Danskin. The theory of max-min, with applications. *SIAM Journal on Applied Mathematics*, 14(4):641–664, 1966.

Jyrki Kivinen and Manfred K. Warmuth. Exponentiated gradient versus gradient descent for linear predictors. *Information and Computation*, 132(1):1–63, 1997.

Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

Robin Manhaeve, Sebastijan Dumancic, Angelika Kimmig, Thomas Demeester, and Luc De Raedt. Deepproblog: Neural probabilistic logic programming. In *NeurIPS*, 2018.

Connor Pryor, Charles Dickens, Eriq Augustine, Alon Albalak, William Wang, and Lise Getooor. Neupsl: Neural probabilistic soft logic. 2022. URL `https://arxiv.org/abs/2205.14268`.

Shai Shalev-Shwartz. Online learning and online convex optimization. *Foundations and Trends in Machine Learning (FTML)*, 4(2):107–194, 2012.

Sriram Srinivasan, Charles Dickens, Eriq Augustine, Golnoosh Farnadi, and Lise Getoor. A taxonomy of weight learning methods for statistical relational learning. *Machine Learning*, 2021.