An empirical comparison of joint-training and pre-training for domain-agnostic semi-supervised learning via energy-based models

Yunfu Song, Huahuan Zheng, Zhijian Ou

Speech Processing and Machine Intelligence (SPMI) Lab, Tsinghua University

http://oa.ee.tsinghua.edu.cn/ouzhijian/

Present at IEEE International Workshop on Machine Learning for Signal Processing (MLSP), October, 2020

Content

1. Related work and Motivation

- Semi-Supervised Learning: Discriminative vs. Generative
- Domain-agnostic
- Probabilistic Graphical Models: Directed vs. Undirected (EBM)

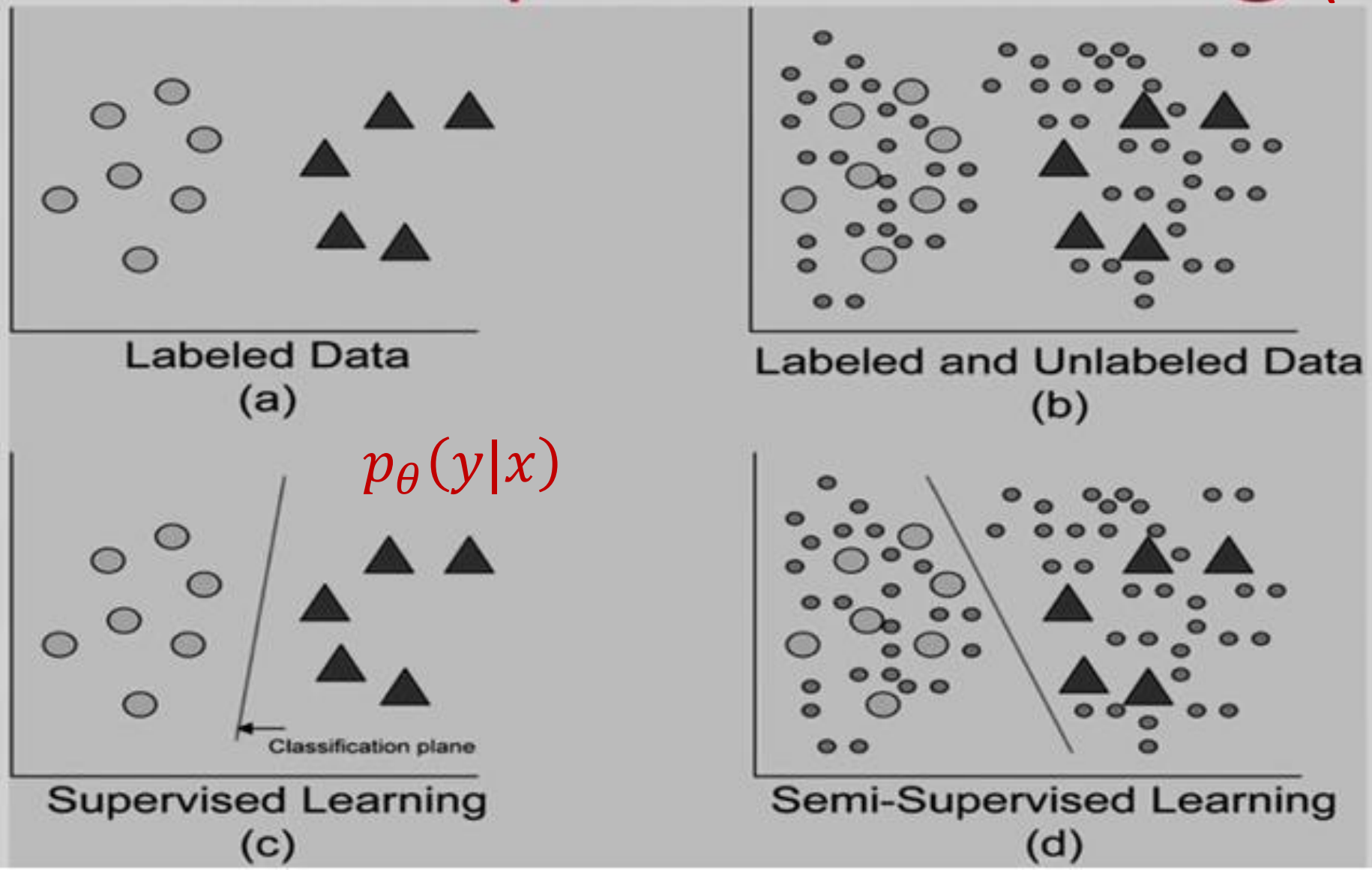
2. Methods and Tasks

- Pre-training vs. Joint-training
- Across domains: image classification and natural language labeling

3. Experiment Results

4. Conclusion

Semi-supervised learning (SSL)



SSL methods (for using DNNs)

- Recent SSL methods with DNNs can be distinguished by the priors they adopt, and, can be divided into two classes.
  - **Generative SSL**
  - **Discriminative SSL**: The outputs from the discriminative classifier are smooth with respect to local and random perturbations of the inputs [1-5].

[1] Takeru Miyato, et al, "Virtual adversarial training: a regularization method for supervised and semi-supervised learning," TPAMI, 2018.
[2] Samuli Laine and Timo Aila, "Temporal ensembling for semisupervised learning," ICLR, 2017.
[3] Antti Tarvainen and Harri Valpola, "Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results," NIPS, 2017.
[4] Kihyuk Sohn, David Berthelot, Chun-Liang Li, and et al, "FixMatch: Simplifying semi-supervised learning with consistency and confidence," arXiv:2001.07685, 2020.
[5] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton, "A simple framework for contrastive learning of visual representations," arXiv:2002.05709, 2020.

Discriminative SSL

- Recent SSL methods with DNNs can be distinguished by the priors they adopt, and, can be divided into two classes.
  - **Generative SSL**
  - **Discriminative SSL**: The outputs from the discriminative classifier are smooth with respect to local and random perturbations of the inputs.

☹ heavily rely on domain-specific data augmentations, which are tuned intensively for images leading to impressive performance in some image domains
☹ less successful for other domains where these augmentations are less effective (e.g., medical images and text). For instance, random input perturbations are more difficult to apply to discrete data like text [6].

[6] Kevin Clark, Minh-Thang Luong, Christopher D Manning, and Quoc Le, "Semi-supervised sequence modeling with cross-view training," in EMNLP, 2018

Generative SSL - Basics

- Exploit unsupervised learning of generative models over unlabeled data, blend unsupervised learning and supervised learning.

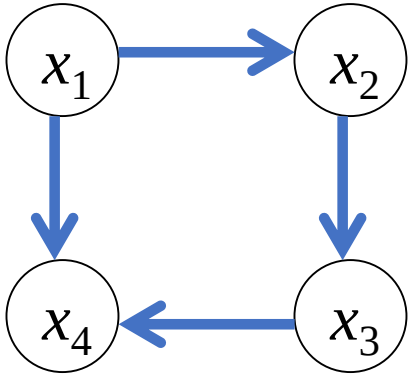
☺ inherently not require data augmentations and generally can be applied to a wider range of domains.
☺ make fewer domain-specific assumptions and tend to be domain-agnostic.

Generative SSL - Two Different Approaches

- **Joint-training**
  - A joint model of p(x,y) is defined.
  - When we have label y, we maximize p(y|x) (the supervised objective), and when the label is unobserved, we marginalize it out and maximize p(x) (the unsupervised objective).
  - Semi-supervised learning over a mix of labeled and unlabeled data is formulated as maximizing the (weighted) sum of log p(y|x) and log p(x).
- **Pre-training**
  - Only define p(x) without y.
  - Perform unsupervised representation learning (called pre-training) on unlabeled data, followed by supervised training (called fine-tuning) on labeled data.
  - This manner of pre-training followed by fine-tuning has received increasing application in natural language processing.

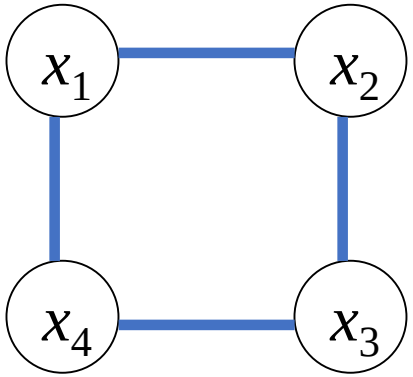
Generative SSL - Two Different Probabilistic Models

- **Directed Graphical Models / Bayesian Networks (BNs)**
  - Self-normalized
  - e.g. Hidden Markov Models (HMMs), Neural network (NN) based classifiers, Variational AutoEncoders (VAEs), Generative Adversarial Networks (GANs), auto-regressive models (e.g. RNNs/LSTMs)



$$P(x_1, x_2, x_3, x_4) = P(x_1)P(x_2|x_1)P(x_3|x_2)P(x_4|x_1, x_3)$$

- **Undirected Graphical Models / Random Fields (RFs) / Energy-based models**
  - Involves the normalizing constant (the partition function) Z
  - e.g. Conditional Random Fields (CRFs)



$$P(x_1, x_2, x_3, x_4) = \frac{1}{Z}\Phi(x_1, x_2)\Phi(x_2, x_3)\Phi(x_3, x_4)\Phi(x_1, x_4)$$

EBM models can be very flexibly defined for SSL, by either of joint-training and pre-training.

… previously known in the literature*, but it is unclear which is better when evaluated in a common experimental setup.

To the best of our knowledge, this paper is the first to systematically compare joint-training and pre-training for EBM-based for SSL, across domains (image classification and natural language labeling).

* EBM based SSL results have been reported across different data modalities (images, natural languages, an protein structure prediction and year prediction from the UCI dataset repository) [12,13,14].

Content

1. Related work and Motivation

- Semi-Supervised Learning: Discriminative vs. Generative
- Domain-agnostic
- Probabilistic Graphical Models: Directed vs. Undirected (EBM)

2. Methods and Tasks

- Pre-training vs. Joint-training
- Across domains: image classification and natural language labeling

3. Experiment Results

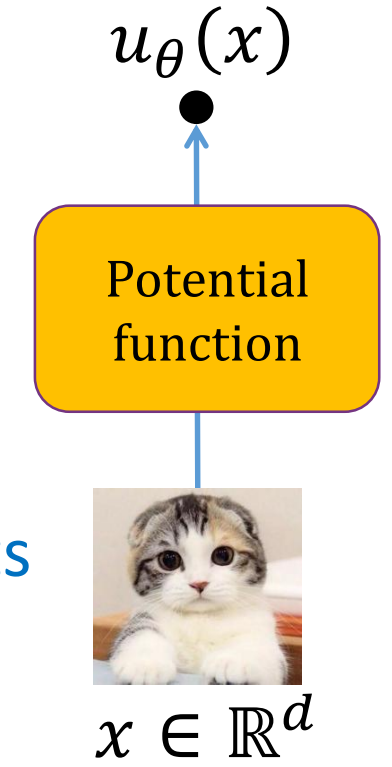
4. Conclusion

Neural Random Fields (NRFs) - Basics

- NRFs are defined by using NNs to implement $u_\theta(x)$: $\mathbb{R}^d \rightarrow \mathbb{R}$

$$p_\theta(x) = \frac{1}{Z(\theta)} exp[u_\theta(x)]$$

  - $u_\theta(x)$ can be very flexibly defined; allows a close connection between p(y|x) and p(x,y).



- This type of RFs has been studied several times in different contexts
  - Deep energy models (DEMs)
    - Ngiam et al., 2012
    - Kim & Bengio, 2016 - includes linear and squared terms in $u_\theta(x)$
  - Descriptive models / Generative ConvNet
    - Xie et al., 2016 / Dai et al., 2014 - defines in the form of exponential tilting of a reference distribution (Gaussian white noise)
  - Neural random field language models
    - Wang & Ou, 2017 - defines over sequences

Learning NRFs - Basics

$$p_\theta(x) = \frac{1}{Z(\theta)} e^{u_\theta(x)}$$

- Maximum-likelihood training

$$\min_\theta KL[\tilde{p}(\tilde{x})||p_\theta(\tilde{x})]$$

$$\nabla_\theta = E_{\tilde{p}(\tilde{x})}[\nabla_\theta log p_\theta(\tilde{x})] = E_{\tilde{p}(\tilde{x})}[\nabla_\theta u_\theta(\tilde{x})] - E_{p_\theta(x)}[\nabla_\theta u_\theta(x)]$$

Expectation under empirical distribution $\tilde{p}(\tilde{x})$

Expectation under model distribution $p_\theta(x)$

- Stochastic maximum likelihood (SML) (Younes, 1989)
  - Approximate the model expectations by Monte Carlo sampling for calculating the gradient.
  - Examples: contrastive divergence (CD) 2002, persistent contrastive divergence (PCD) 2008

L. Younes, “Parametric inference for imperfectly observed gibbsian fields,” Probability Theory and Related Fields, 1989.

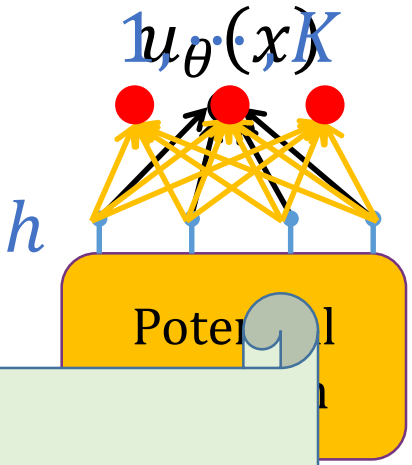
Table 1. Applications of EBMs across different domains: comparison and connection (See text for details).

	Image classification	Natural language labeling
Observation	$x \in \mathbb{R}^D$ continuous, fixed-dimensional	$x \in \bigcup_l \mathbb{V}^l$ discrete, sequence
Label	$y \in \{1, 2, \cdots, K\}$	$y \in \bigcup_l \{1, 2, \cdots, K\}^l$
Pre-training	① $u_\theta(x) = w^T h$	② $u_\theta(x)$ in Eq.(3)
Joint-training	③ $u_\theta(x, y) = \Psi_\theta(x)[y]$	④ $u_\theta(x, y)$ in Eq.(6)

① Pre-training of an EBM for semi-supervised image classification

1) Pre-training: estimate $p_\theta(x)$ over unlabeled images

$$p_\theta(x) = \frac{1}{Z(\theta)} exp[u_\theta(x)]$$



Use a feedforward NN to implement $u_\theta(x): \mathbb{R}^d \to \mathbb{R}$

wh

It can be seen that pre-training aims to learn representations that may be useful for multiple downstream tasks, and any information about the labels is not utilized until the fine-tuning stage.

2)

followed by $\mathrm{softmax}(Wh)$, to predict $y \in \{1, \cdots, K\}$, where $W \in \mathbb{R}^{K \times H}$

[12] Yunfu Song, Zhijian Ou. Learning Neural Random Fields with Inclusive Auxiliary Generators. arXiv:1806.00271, 2018.

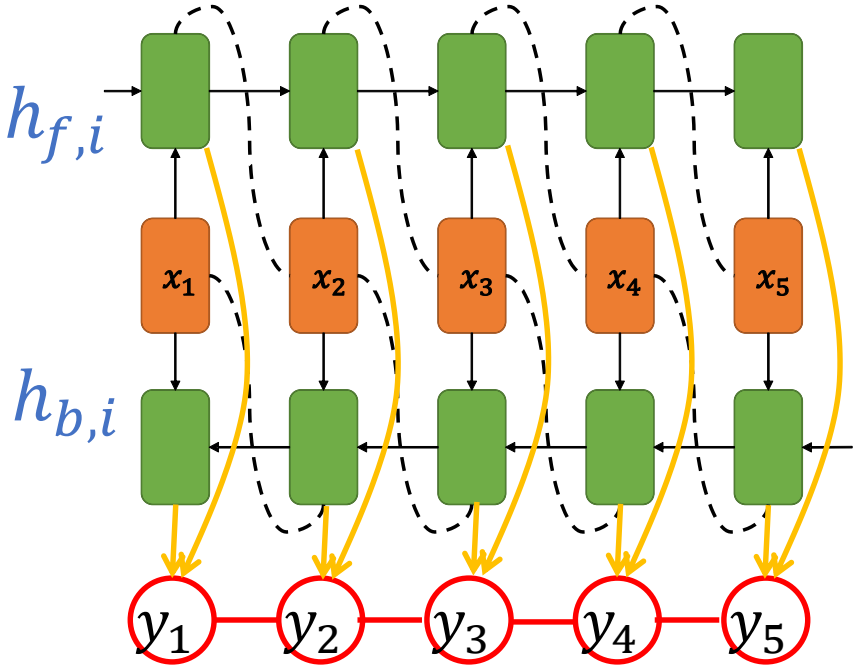
② Pre-training of an EBM for semi-supervised natural language labeling

1) Pre-training: estimate $p_\theta(x)$ over unlabeled sentences $x = (x_1, \cdots, x_l)$

$$p_\theta(x) = \frac{1}{Z(\theta)} exp[u_\theta(x)]$$

Use a B-LSTM to implement $u_\theta(x)$: $\mathbb{V}^l \to \mathbb{R}$

$$u_\theta(x) = \sum_{i=1}^{l-1} h_{f,i}^T e_{i+1} + \sum_{i=2}^{l} h_{b,i}^T e_{i-1}$$



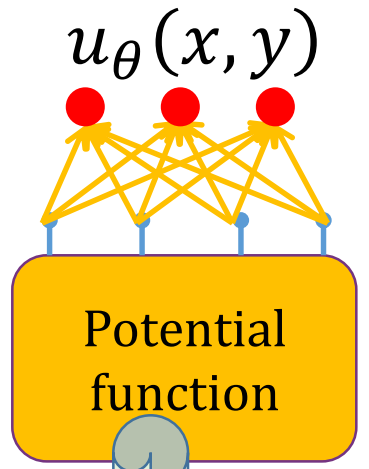
2) Fine-tuning: we add a CRF on top of the extracted representations $\{(h_{f,i}, h_{b,i}), i = 1, \cdots, l\}$ to predict label sequence $y = (y_1, \cdots, y_l)$.

[20] Bin Wang, Zhijian Ou. Improved training of neural trans-dimensional random field language models with dynamic noise-contrastive estimation. IEEE Workshop on Spoken Language Technology (SLT), 2018.

③ Joint-training of an EBM for semi-supervised image classification

- Joint modeling of observation $x \in \mathbb{R}^d$ and class label $y \in \{1, \cdots, K\}$:

$$p_\theta(x, y) = \frac{1}{Z(\theta)} exp[u_\theta(x, y)]$$



- Consider a NN $\Psi_\theta(x)$: $\mathbb{R}^d \to \mathbb{R}^K$ and define:

Different from pre-training, the unsupervised objective $p_\theta(x)$ in joint-training depends on the targeted task.

$$\begin{cases} \min_\theta KL[\tilde{p}(\tilde{x})||p_\theta(\tilde{x})] - \alpha \sum_{(\tilde{x},\tilde{y})\sim\mathcal{L}} log p_\theta(\tilde{y}|\tilde{x}) \\ \min_\phi KL\left[p_\theta(x)||q_\phi(x)\right] \end{cases}$$

[12] Yunfu Song, Zhijian Ou. Learning Neural Random Fields with Inclusive Auxiliary Generators. arXiv:1806.00271, 2018.

④ Joint-training of an EBM for semi-supervised natural language labeling

- JRF: Define a joint distribution over $x = (x_1, \cdots, x_l)$ and $y = (y_1, \cdots, y_l)$

$$p_\theta(l, x^l, y^l) = \pi_l p_\theta(x^l, y^l; l) = \frac{\pi_l}{Z_\theta(l)} \exp\left(u_\theta(x^l, y^l)\right)$$

- Consider a NN $\Psi_\theta(x)$: $\mathbb{V}^l \to \mathbb{R}^{l \times K}$ and define:

$$u_\theta(x, y) = \sum_{i=1}^{l} \Psi_\theta(x)[i, y_i] + \sum_{i=1}^{l} A[y_{i-1}, y_i]$$

- From JRF we have:

$$p_\theta(y^l | x^l) = \frac{1}{\sum_{y^l} \exp\left(u_\theta(x^l, y^l)\right)} \exp\left(u_\theta(x^l, y^l)\right)$$

which is a CRF

- From JRF we have:

$$p_\theta(l, x^l) = \frac{\pi_l}{Z_\theta(l)} \sum_{y^l} \exp\left(u_\theta(x^l, y^l)\right)$$

$$= \frac{\pi_l}{Z_\theta(l)} \exp\left(u_\theta(x^l)\right)$$

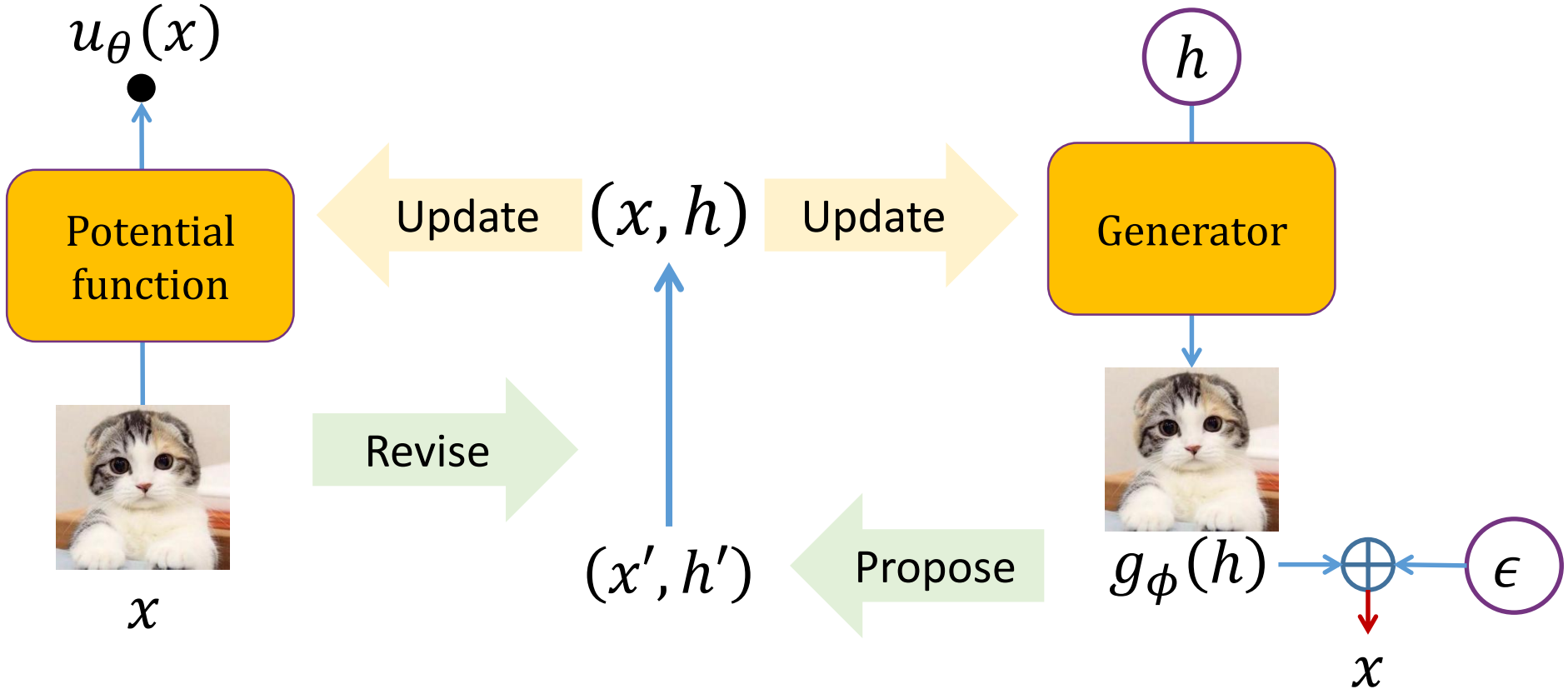
where $u_\theta(x^l) = \log \sum_{y^l} \exp\left(u_\theta(x^l, y^l)\right)$

which is a trans-dimensional random field (TRF)

[14] Yunfu Song, Zhijian Ou, et al. Upgrading CRFs to JRFs and its benefits to sequence modeling and labeling. ICASSP, 2020.

Inclusive-NRF algo. for learning from continuous data, e.g., Images.

simultaneously training a random field and a generator.



$$\begin{cases} \min_\theta KL[\tilde{p}(\tilde{x})||p_\theta(\tilde{x})] \\ \min_\phi KL[p_\theta(x)||q_\phi(x)] \end{cases} \Rightarrow \begin{cases} \nabla_\theta = E_{\tilde{p}(\tilde{x})}[\nabla_\theta log p_\theta(\tilde{x})] = E_{\tilde{p}(\tilde{x})}[\nabla_\theta u_\theta(\tilde{x})] - E_{p_\theta(x)}[\nabla_\theta u_\theta(x)] \\ \nabla_\phi = E_{p_\theta(x)}[\nabla_\phi log q_\phi(x)] = E_{p_\theta(x)q_\phi(h|x)}[\nabla_\phi log q_\phi(x,h)] \end{cases}$$

[12] Yunfu Song, Zhijian Ou. Learning Neural Random Fields with Inclusive Auxiliary Generators. arXiv:1806.00271, 2018.

Dynamic NCE algo. for learning from discrete data, e.g., texts.

Simultaneously train a random field and a generator.

- The target RF model $p_\theta(x) = \frac{1}{Z(\theta)} e^{u_\theta(x)}$
- Treat $\log Z(\theta)$ as a parameter ζ and rewrite $p_{\theta,\zeta}(x) \propto e^{u_\theta(x)-\zeta}$
- Introduce a **noise distribution** $q_n(x)$, and consider a binary classification

$$P(C=0|x) = \frac{p_{\theta,\zeta}(x)}{p_{\theta,\zeta}(x) + \nu q_n(x)}, where\ \nu = \frac{P(C=1)}{P(C=0)}$$

$$P(C=1|x) = 1 - P(C=0|x)$$

- Noise Contrastive Estimation (NCE):

$$\max_{\theta,\zeta} E_{x\sim p_0(x)}[\log P(C=0|x)] + E_{x\sim q_n(x)}[\log P(C=1|x)]$$

☺ $p_\theta \to p_0$ (oracle), under infinite amount of data and infinite capacity of p_θ.

☹ Reliable NCE needs a large $\nu \approx 20$; Overfitting. Dynamic-NCE in (Wang&Ou, SLT 2018).

Content

1. Related work and Motivation

- Semi-Supervised Learning: Discriminative vs. Generative
- Domain-agnostic
- Probabilistic Graphical Models: Directed vs. Undirected (EBM)

2. Methods and Tasks

- Pre-training vs. Joint-training
- Across domains: image classification and natural language labeling

3. Experiment Results

4. Conclusion

Table 2. SSL for image classification over CIFAR-10 with 4,000 labels. The upper/lower blocks show generative/discriminative SSL methods respectively. The means and standard deviations are calculated over ten independent runs with randomly sampled labels.

Methods	error (%)
CatGAN [30]	19.58±0.46
Ladder network [31]	20.40±0.47
Improved-GAN [32]	18.63±2.32
BadGAN [33]	14.41±0.30
Sobolev-GAN [34]	15.77±0.19
Supervised baseline	25.72±0.44
Pre-training+fine-tuning EBM	21.40±0.38
Joint-training EBM	15.12±0.36
Results below this line cannot be directly compared to those above.	
VAT small [1]	14.87
Temporal Ensembling [2]	12.16±0.31
Mean Teacher [3]	12.31±0.28

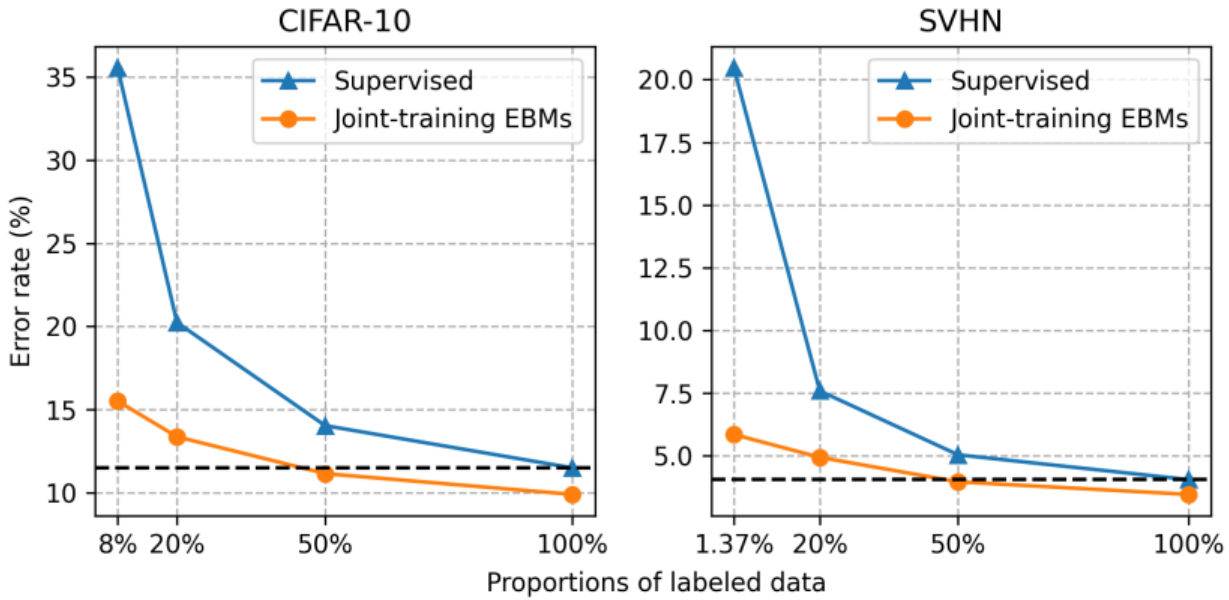


Fig. 1. Error rates of supervised baseline and joint-training EBMs as the amount of labels varies on SVHN and CIFAR-10 datasets. The dash line is the supervised result trained with 100% labeled data.

Table 3. Natural language labeling results. The evaluation metric is accuracy for POS and F_1 for chunking and NER. "Labeled" denotes the amount of labels in terms of the proportions w.r.t. the full set of labels. "U/L" denotes the ratio between the amount of unlabeled and labeled data. "U/L=0" denotes the supervised baseline. "pre." and "joint" denote the results by pre-training+fine-tuning EBMs and joint-training EBMs, respectively.

Labeled	U/L	POS tagging		Chunking		NER	
		pre.	joint	pre.	joint	pre.	joint
2%	0	95.57		78.73		78.19	
	50	95.72	95.92	81.62	82.24	76.74	77.61
	250	95.96	96.13	82.10	82.26	78.49	78.51
	500	96.08	96.24	83.10	83.05	79.47	79.17
10%	0	96.81		90.06		86.93	
	50	96.87	96.99	91.60	91.85	86.37	87.05
	250	96.88	97.00	91.09	91.93	86.86	86.77
	500	96.92	97.08	91.93	92.23	87.57	87.06
100%	0	97.41		94.77		90.74	
	50	97.40	97.49	95.05	95.31	91.24	91.34
	250	97.45	97.54	95.12	95.48	91.19	91.51
	500	97.46	97.57	95.19	95.50	91.30	91.52

Table 4. Relative improvements by joint-training EBMs compared to the supervised baseline (abbreviated as sup.) and pretraining+fine-tuning EBMs respectively. Refer to Table 3 for notations.

		joint over sup.			joint over pre.		
Labeled	U/L	POS	Chunking	NER	POS	Chunking	NER
2%	50	7.9	16.5	-2.7	4.7	3.4	3.7
	250	12.6	16.6	1.5	4.2	0.9	0.1
	500	15.1	20.3	4.5	4.1	-0.3	-1.5
10%	50	5.6	18.0	0.9	3.8	3.0	5.0
	250	6.0	18.3	-1.2	3.8	9.4	-0.7
	500	8.5	21.8	1.0	5.2	3.7	-4.1
100%	50	3.1	10.3	6.5	3.5	5.3	1.1
	250	5.0	13.6	8.3	3.5	7.4	3.6
	500	6.2	14.0	8.4	4.3	6.4	2.5

Conclusions

- We systematically evaluate and compare joint-training and pre-training for EBM-based domain-agnostic SSL, through a suite of experiments across a variety of domains such as image classification and natural language labeling.
- **Joint-training EBMs outperform pre-training EBMs marginally but nearly consistently.**
  - ▶Presumably, this is because that the optimization of joint-training is directly related to the targeted task, but pre-training is not aware of the labels for the targeted task.
- We hope this new finding would be helpful for future work to further explore better methods to leverage unlabeled data.

Thanks for your attention !

Reproducible code is at https://github.com/thu-spmi/semi-EBM

References

1. Zhijian Ou. A Review of Learning with Deep Generative Models from Perspective of Graphical Modeling. arXiv:1808.01630.
2. Yunfu Song, Zhijian Ou. Learning Neural Random Fields with Inclusive Auxiliary Generators. arXiv:1806.00271, 2018.
3. Bin Wang, Zhijian Ou, Zhiqiang Tan. Learning Trans-dimensional Random Fields with Applications to Language Modeling. IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI), 2018, vol.40, no.4, pp.876-890.
4. Bin Wang, Zhijian Ou. Improved training of neural trans-dimensional random field language models with dynamic noise-contrastive estimation. IEEE Workshop on Spoken Language Technology (SLT), Athens, Greece, 2018.
5. Yunfu Song, Zhijian Ou, Zitao Liu, Songfan Yang. Upgrading CRFs to JRFs and its benefits to sequence modeling and labeling. ICASSP, Barcelona, Spain, 2020.
6. Yunfu Song, Huahuan Zheng, Zhijian Ou. An empirical study of domain-agnostic semi-supervised learning via energy-based models: joint-training and pre-training. arxiv:2010.13116, 2020.

[Submitted on 1 Jun 2018 (this version), latest version 22 Jul 2020 (v5)]

Learning Neural Random Fields with Inclusive Auxiliary Generators

Yunfu Song, Zhijian Ou

In this paper we develop Neural Random Field learning with Inclusive-divergence minimized Auxiliary Generators (NRF-IAG), which is under-appreciated in the literature. The contributions are two-fold. First, we rigorously apply the stochastic approximation algorithm to solve the joint optimization and provide theoretical justification. The new approach of learning NRF-IAG achieves superior unsupervised learning performance competitive with state-of-the-art deep generative models (DGMs) in terms of sample generation quality. Second, semi-supervised learning (SSL) with NRF-IAG gives rise to strong classification results comparable to state-of-art DGM-based SSL methods, and simultaneously achieves superior generation. This is in contrast to the conflict of good classification and good generation, as observed in GAN-based SSL.

Published as a conference paper at ICLR 2020

YOUR CLASSIFIER IS SECRETLY AN ENERGY BASED MODEL AND YOU SHOULD TREAT IT LIKE ONE

Will Grathwohl
University of Toronto & Vector Institute
Google Research
wgrathwohl@cs.toronto.edu

Kuan-Chieh Wang*& Jörn-Henrik Jacobsen*
University of Toronto & Vector Institute
wangkua1@cs.toronto.edu
j.jacobsen@vectorinstitute.ai

David Duvenaud
University of Toronto & Vector Institute
duvenaud@cs.toronto.edu

Kevin Swersky & Mohammad Norouzi
Google Research
{kswersky, mnorouzi}@google.com

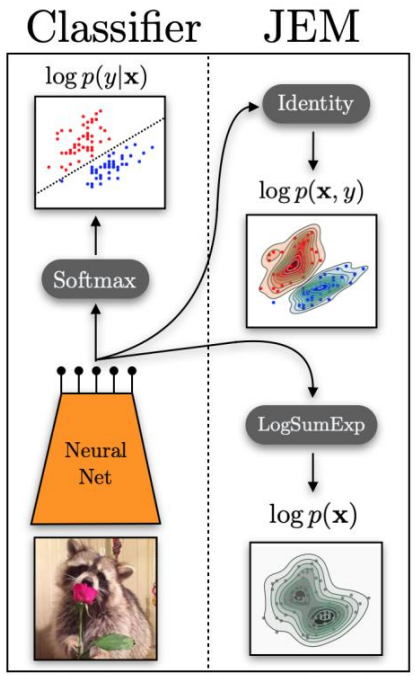


Figure 1: Visualization of our method, JEM, which defines a joint EBM from classifier architectures.

Conditional random field (CRF)

(Linear-chain) CRFs define a conditional distribution y^l given x^l of length l :

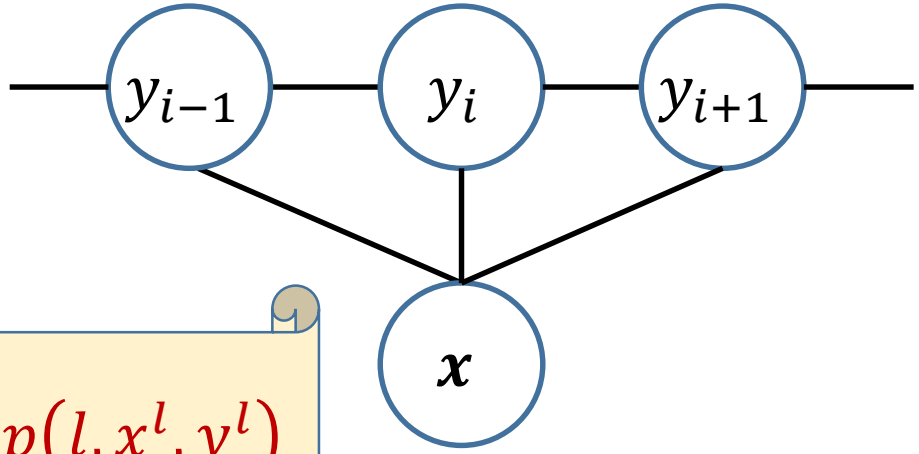
$$p_\theta(y^l|x^l) = \frac{1}{Z_\theta(x^l)} \exp(u_\theta(x^l, y^l)) \qquad Z_\theta(x^l) = \sum_{y^l} \exp(u_\theta(x^l, y^l))$$

Potential function:

Node potential

Edge potential

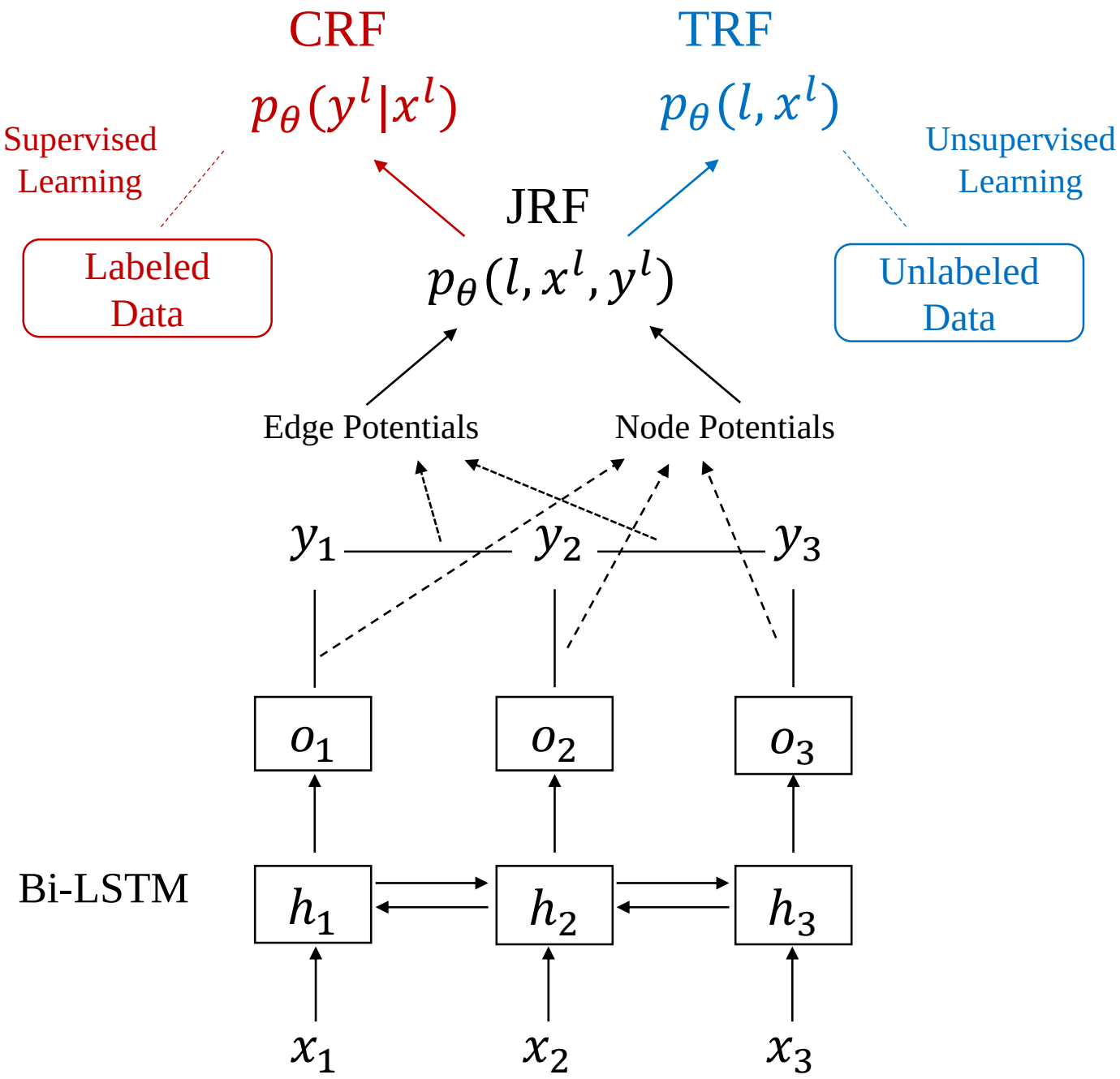
$$u_\theta(x^l, y^l) = \sum_{i=1}^{l} \phi_i(y_i, x^l) + \sum_{i=1}^{l} \psi_i(y_{i-1}, y_i, x^l)$$



- Upgrade CRFs to, a joint generative model of x^l and y^l, $p(l, x^l, y^l)$
  - Use $u(x^l, y^l)$ in the original CRF

Lafferty, John, Andrew McCallum, and Fernando CN Pereira. "Conditional random fields: Probabilistic models for segmenting and labeling sequence data." (2001).

JRF


CRF
$p_\theta(y^l|x^l)$
TRF
$p_\theta(l,x^l)$
Supervised Learning
Unsupervised Learning
JRF
$p_\theta(l,x^l,y^l)$
Labeled Data
Unlabeled Data
Edge Potentials
Node Potentials
y_1
y_2
y_3
o_1
o_2
o_3
Bi-LSTM
h_1
h_2
h_3
x_1
x_2
x_3