2021 IEEE INTERNATIONAL WORKSHOP ON MACHINE LEARNING FOR SIGNAL PROCESSING, OCT. 25–28, 2021, GOLD COAST, AUSTRALIA

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, Beijing, China

ABSTRACT

Some semi-supervised learning (SSL) methods heavily rely on domain-specific data augmentations. Recently, semi-supervised learning (SSL) via energy-based models (EBMs) has been studied and is attractive from the perspective of being domain-agnostic, since it inherently does not require data augmentations. There exist two different methods for EBM based SSL - joint-training and pre-training. Joint-training estimates the joint distribution of observations and labels, while pre-training is taken over observations only and followed by fine-tuning. Both joint-training and pre-training are previously known in the literature, but it is unclear which one 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, by conducting a suite of experiments across a variety of domains such as image classification and natural language labeling. It is found that joint-training EBMs outperform pre-training EBMs marginally but nearly consistently, presumably because the optimization of joint-training is directly related to the targeted task, while pre-training does not.

***Index Terms*—** semi-supervised learning, energy-based models, neural random fields, conditional random fields, joint random fields

1. INTRODUCTION

A plethora of semi-supervised learning (SSL) methods have emerged to leverage both labeled and unlabeled data to train deep neural networks (DNNs) [1, 2, 3, 4, 5, 6], spanning over various domains such as image classification, natural language labeling and so on. Roughly speaking, recent SSL methods with DNNs could be divided into two classes[1] - based on generative models or discriminative models, which are referred to as generative SSL and discriminative SSL respectively. Discriminative SSL methods often assume that the outputs from the discriminative classifier are smooth with respect to local and random perturbations of the inputs. These SSL methods thus heavily rely on domain-specific data augmentations [8], which are tuned intensively for images and lead to impressive performance in some image domains. But discriminative SSL is often 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].

Generative SSL methods exploit unsupervised learning of generative models over unlabeled data, which inherently does not require data augmentations and generally can be applied to a wider range of domains. Considering observation x and label y, there exist two different methods for the generative SSL approach - joint-training [9, 10] and pre-training [11]. In 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 $\log p(x)$. In pre-training, we perform unsupervised representation learning on unlabeled data, which is followed by supervised training (called fine-tuning) on labeled data.

Among existing generative SSL methods, a class of generative models - energy-based models (EBMs) have been shown with promising results for semi-supervised learning across various domains. Early studies date back to the pre-training of restricted Boltzmann machines (RBMs) [11] (which are a simple type of EBMs) and the joint-training with classification RBMs. More encouraging SSL results have been shown recently for joint-training via more advanced EBMs, which are defined by using DNN-based energy functions. In [12, 13, 14], state-of-the-art SSL results are reported based on EBMs and across different data modalities (images, natural languages, an protein structure prediction and year prediction from the UCI dataset repository) and in different data settings (fix-dimensional and sequence data). Although both joint-training and pre-training of EBMs have been used for SSL in the literature, previous studies have not yet evaluate and compare the two methods. The results from previous individual works are not directly comparable to each other, since they are not evaluated in a common experimental setup.

This work is supported by NSFC 61976122 and Tsinghua-China Mobile Joint Institute. Corresponding author: Zhijian Ou (ozj@tsinghua.edu.cn). Code is available at https://github.com/thu-spmi/semi-EBM

[1] We mainly discuss the SSL methods for using DNNs. General discussion of SSL can be referred to [7].

978-1-7281-6338-3/21/$31.00 ©2021 IEEE

In this paper, we conduct a suite of experiments to systematically compare joint-training and pre-training for EBM-based SSL. As suggested in [15], we vary both the amounts of labeled and unlabeled data to give a realistic whole picture of the performances of the two methods for SSL.

2. RELATED WORK

Discriminative and generative SSL. Semi-supervised learning is a heavily studied problem. Discriminative SSL works by discriminating between different augmentations from a given unlabeled sample, such as in recent FixMatch [4], SimCLR [5] methods. They rely on a rich set of domain-specific data augmentations, e.g., RandAugment [8]. Although there are some efforts to use data-independent model noises, e.g., by dropout [16], domain-specific data augmentations is indispensable.

Recent progress in learning with deep generative models stimulates the generative SSL research, which usually involves blending unsupervised learning and supervised learning. These methods make fewer domain-specific assumptions and tend to be domain-agnostic. The performance comparisons between generative and discriminative SSL methods are mixed. It is found that consistency based discriminative SSL methods often outperform generative SSL methods in image domain. However, in text domain, the generative SSL methods such as those based on pre-training word vectors are more successful and widely used.

EBM based generative SSL. Pre-training of RBMs once received attention in the early stage of training DNNs [11]. Recently, it is shown in [12] that joint-training via EBMs produces state-of-the-art SSL results on images (MNIST, SVHN and CIFAR-10), compared to previous generative SSL methods based on Variational AutoEncoders (VAEs) and Generative Adversarial Networks (GANs). It is also shown in [13] that joint-training via EBMs outperforms VAT (virtual adversarial training) [1] on tabular data from the UCI dataset repository. Further, joint-training via EBMs has been extended to modeling sequences and consistently outperforms conditional random fields (CRFs) (the supervised baseline) and self-training (the classic semi-supervised baseline) on natural language labeling tasks such as POS (part-of-speech) tagging, chunking and NER (named entity recognition). Despite these previous works showing the advantage of EBMs in domain-agnostic SSL, a direct and fair comparison of joint-training and pre-training for EBM-based SSL, however, has not been known in the literature, to the best of our knowledge.

3. SEMI-SUPERVISED LEARNING VIA EBMS

In this section, we review the methods of joint-training and pre-training for EBM-based SSL across different data modalities, which are scattered in previous individual works, using consistent notations summarized in Table 1.

3.1. Background

An energy-based model (EBM) [17], also known as a random field [18], defines a probability distribution for a collection of random variables $x \in \mathcal{X}$ with parameter θ in the form:

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

where $\mathcal{X}$ denotes the space of all possible values of x, and $Z(\theta) = \int \exp\left[u_\theta(x)\right] dx$ is the normalizing constant. $u_\theta(x) : \mathcal{X} \to \mathbb{R}$ is called the potential function which assigns a scalar value to each configuration of x in $\mathcal{X}$ and can be very flexibly defined (e.g., through DNNs of different architectures). For different applications, $\mathcal{X}$ could be discrete or continuous, and x could be fix-dimensional or trans-dimensional (i.e., sequences of varying lengths). For example, images are fix-dimensional continuous data (i.e., $\mathcal{X} = \mathbb{R}^D$), and natural languages are sequences taking discrete tokens (i.e., $\mathcal{X} = \bigcup_l \mathbb{V}^l$ where $\mathbb{V}$ is the vocabulary of tokens).

Training EBMs is challenging, because the gradient in maximizing the data log-likelihood $\log p_\theta(x)$ for observed x involves expectation w.r.t. the model distribution $p_\theta(x)$, as shown below:

$$\begin{aligned} \nabla_\theta \log p_\theta(x) &= \nabla_\theta u_\theta(x) - \nabla_\theta \log Z(\theta) \\ &= \nabla_\theta u_\theta(x) - E_{p_\theta(x')}\left[\nabla_\theta u_\theta(x')\right]. \end{aligned} \tag{2}$$

Considerable progress has been made recently to successfully train large-scale EBMs parameterized by DNNs [12, 19, 14, 20] for different types of data from various domains, which lays the foundation to use EBMs, as a unified framework, to achieve domain-agnostic SSL.

- For training EBMs for continuous data such as images, the inclusive approach, as detailed in [12], has been shown to yield superior results in unsupervised and semi-supervised training, by introducing inclusive-divergence minimized auxiliary generators and utilizing stochastic gradient sampling (such as SGLD) to approximate the model expectation in Eq. (2).
- For training EBMs for discrete sequence data such as natural languages, the DNCE approach, as detailed in [20, 14], avoids the model expectation in Eq. (2) and has achieved superior results in unsupervised and semi-supervised training, with the use of dynamic noise distribution to improve training efficiency of NCE (noise-contrastive estimation) [21].

3.2. Pre-training via EBMs for SSL

Pre-training via EBMs for SSL consists of two stages. The first stage is pre-training an EBM on unlabeled data. It is followed by a fine-tuning stage, where we can easily use the pre-trained EBM to initialize a discriminative model and further train over labeled data.

Consider **pre-training of an EBM for semi-supervised image classification**, which essentially involves estimating $p_\theta(x)$ as defined in Eq. (1) from unlabeled images. For the potential function $u_\theta(x)$, we can use a multi-layer feed-forward neural network $\Phi_\theta(x) : \mathbb{R}^D \to \mathbb{R}$, which, in the final layer, calculates a scalar via a linear layer, $u_\theta(x) = w^T h$. Here $h \in \mathbb{R}^H$ denotes the activation from the last hidden layer and $w \in \mathbb{R}^H$ the weight vector in the final linear layer. For simplicity, we omit the bias in describing linear layers throughout the paper.

In fine-tuning, we throw away w and fed h into a new linear output layer, followed by $softmax(Wh)$, to predict y, where $W \in \mathbb{R}^{K \times H}$ denotes the new trainable weight parameters and $y \in \{1, \cdots, K\}$ the class label. *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.*

The above procedure can be similarly applied to **pre-training of an EBM for semi-supervised natural language labeling**, e.g., POS tagging. In pre-training, we estimate an EBM-based language model $p_\theta(x)$ from unlabeled text corpus. Neural networks with different architectures can be used to implement the potential function $\Phi_\theta(x) : \mathbb{V}^l \to \mathbb{R}$ given length l. With abuse of notation, here $x = (x_1, \ldots, x_l)$ denotes a token sequence of length l, and $x_i \in \mathbb{V}, i = 1, \cdots, l$. We use the bidirectional LSTM based potential function in [20] as follows:

$$u_\theta(x) = \sum_{i=1}^{l-1} h_{f,i}^T e_{i+1} + \sum_{i=2}^{l} h_{b,i}^T e_{i-1} \tag{3}$$

where $e_i, h_{f,i}$ and $h_{b,i}$ are of the same dimensions, denoting the output embedding vector, the last hidden vectors of the forward and backward LSTMs respectively at position i.

In fine-tuning, we add a CRF, as the discriminative model, on top of the extracted representations $\{(h_{f,i}, h_{b,i}), i = 1, \cdots, l\}$ to do sequence labeling, i.e., to predict a sequence of labels $y = (y_1, \ldots, y_l)$ with one label for one token at each position, where $y_i \in \{1, \cdots, K\}$ denotes the label at position i. Specifically, we concatenate $h_{f,i}$ and $h_{b,i}$, add a linear output layer to define the node potential, and add a matrix $A \in \mathbb{R}^{K \times K}$ to define the edge potential, as in recent neural CRFs [22, 23]. The parameters to be fine-tuned are the weights in the linear output layer and the edge potential matrix A.

3.3. Joint-training via EBMs for SSL

The above pre-training via EBMs for SSL considers the modeling of only observations x without labels y. The joint-training refers to the joint modeling of x and y:

$$p_\theta(x, y) = \frac{1}{Z(\theta)} \exp\left[u_\theta(x, y)\right] \tag{4}$$

Then, it can be easily seen that the conditional density $p_\theta(y|x)$ implied by the joint density Eq. (4) is:

$$p_\theta(y|x) = \frac{p_\theta(x, y)}{p_\theta(x)} = \frac{\exp(u_\theta(x, y))}{\sum_{y'} \exp(u_\theta(x, y'))} \tag{5}$$

And the implied marginal density is $p_\theta(x) = \frac{1}{Z(\theta)} \exp(u_\theta(x))$, where, with abuse of notation, $u_\theta(x) \triangleq log \sum_y \exp\left[u_\theta(x, y)\right]$. *Different from pre-training, the unsupervised objective $p_\theta(x)$ depends on the targeted task.* The key for EBM based joint-training for SSL is to choose suitable $u_\theta(x, y)$ such that both $p_\theta(y|x)$ and $p_\theta(x)$ can be tractably optimized.

In **joint-training of an EBM for semi-supervised image classification**, we consider a neural network $\Psi_\theta(x) : \mathbb{R}^D \to \mathbb{R}^K$, which accepts the image x and outputs an vector, whose size is equal to the number of class labels, K. Then we define $u_\theta(x, y) = \Psi_\theta(x)[y]$, where $[y]$ denotes the y-th element of a vector. With the above potential definition, it can be easily seen that the implied conditional density $p_\theta(y|x)$ is exactly a standard K-class $softmax$ based classifier, using the K logits calculated by the neural network $\Psi_\theta(x)$ from the input x. And we do not need to calculate $Z(\theta)$ for classification. Therefore, we can conduct SSL over a mix of labeled and unlabeled data by maximizing the (weighted) sum of $\log p_\theta(y|x)$ and $\log p_\theta(x)$, where both optimizations are tractable as detailed in [12].

The above procedure can be similarly applied to **joint-training of an EBM for semi-supervised natural language labeling** with $x = (x_1, \ldots, x_l)$ and $y = (y_1, \ldots, y_l)$, $x_i \in \mathbb{V}, y_i \in \{1, \cdots, K\}, i = 1, \cdots, l$. We consider a neural network $\Psi_\theta(x) : \mathbb{V}^l \to \mathbb{R}^{l \times K}$ and define

$$u_\theta(x, y) = \sum_{i=1}^{l} \Psi_\theta(x)[i, y_i] + \sum_{i=1}^{l} A[y_{i-1}, y_i] \tag{6}$$

where $[\cdot, \cdot]$ denotes the element of a matrix and $A \in \mathbb{R}^{K \times K}$ models the edge potential for adjacent labels. With the above potential definition, it can be easily seen that the conditional density $p_\theta(y|x)$ implied by the joint density Eq.(4) is exactly a CRF with node potentials $\Psi_\theta(x)[i, y_i]$ and edge potentials $A[y_{i-1}, y_i]$, and the implied marginal density $p_\theta(x)$ is exactly a trans-dimensional random field (TRF) language model [24, 25, 26]. Training of both models are tractable as detailed in [14, 20].

4. EXPERIMENTS

SSL experiments are conducted on standard benchmark datasets in different domains, including the CIFAR-10 and SVHN datasets [12] for image classification and the POS, chunking and NER datasets [27, 14] for natural language labeling. We use the standard data split for training and testing. When we vary the amount of labeled and unlabeled data for training, we select varying proportions (e.g., 10%,

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	$u_\theta(x) = w^T h$	$u_\theta(x)$ in Eq.(3)
Joint-training	$u_\theta(x, y) = \Psi_\theta(x)[y]$	$u_\theta(x, y)$ in Eq.(6)

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

Methods	error (%)
CatGAN [28]	19.58±0.46
Ladder network [29]	20.40±0.47
Improved-GAN [30]	18.63±2.32
BadGAN [31]	14.41±0.30
Sobolev-GAN [32]	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

100%) of labels from the original full set of labeled data. Throughout the paper, the amount of labels is thus described in terms of proportions. "100% labeled" means 50,000 and 73,257 images for CIFAR-10 and SVHN, and 56K, 7.4K, 14K sentences for POS, chunking and NER, respectively.

4.1. SSL for Image Classification

First, we experiment with CIFAR-10 and compare different generative SSL methods. As in previous works, we randomly sample 4,000 labeled images for training. The remaining images are treated as unlabeled. We use the network architectures and hyper-parameter settings in [12]. It can be seen from Table 2 that semi-supervised EBMs, especially the joint-training EBMs, produce strong results on par with state-of-art generative SSL methods[2]. Furthermore, joint-training EBMs outperform pre-training+fine-tuning EBMs by a large margin in this task. Note that some discriminative SSL methods, as listed in the lower block in Table 2, also produce superior results but heavily utilize domain-specific data augmentations, and thus are not directly compared to the generative SSL methods.

[2]As discussed in [12], Bad-GANs could hardly be classified as a generative SSL method.

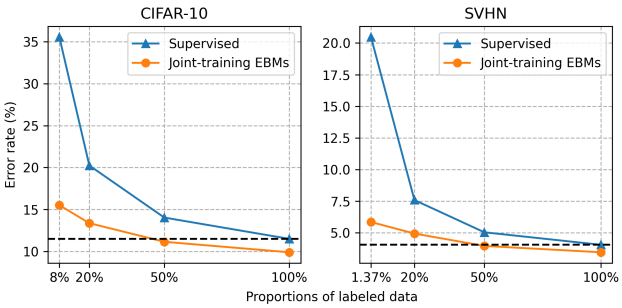

Fig. 1. Error rates of the supervised baseline and the joint-training EBMs as the amount of labels varies on SVHN and CIFAR-10 datasets. The dash line is the supervised result trained with 100% labeled data.

Second, we experiment with CIFAR-10 and SVHN, and examine the effects of varying amount of labels. We sample varying proportions of labels as labeled training data and use the remaining as unlabeled training data (i.e., we do not add external unlabeled data). From the plot of error rates w.r.t. labeling proportions in Fig. 1 , we can see how many labels can be reduced by using joint-training EBMs. The joint-training EBMs obtain 11.14% on CIFAR-10 and 3.95% on SVHN using only 50% labels, which are better than 11.49% and 4.04% respectively, obtained by the supervised baseline using 100% labels. This indicates that we can reduce 50% of labels without losing performance on these two tasks. Additionally, it is interesting to observe that in the case of using 100% labels, the joint-training EBMs outperform the supervised baseline with 13.9% and 14.6% reductions in error rates. This is because the generative loss $p_\theta(x)$ provides regularization for the pure discriminative loss $p_\theta(y|x)$, as discussed in [33].

4.2. SSL for Natural Language Labeling

In this experiment, we evaluate different methods for natural language labeling, through three tasks - POS tagging, chunking and NER. The following benchmark datasets are used - PTB POS tagging, CoNLL-2000 chunking and CoNLL-2003 English NER, as in [23, 6, 27, 14]. We sample varying proportions of labels as labeled training data and use the Google one-billion-word dataset [34] as the large pool of unlabeled sentences. In [14], joint-training EBM based experiments are conducted, using the labeling proportions of 10% and 100% with "U/L" (the ratio between the amount of unlabeled and labeled) of 50. In this paper, a larger scale of experiments are conducted, covering the labeling proportions of 2%, 10% and 100% with "U/L" of 50, 250 and 500 for three tasks, which consist of a total of 27 settings. We use the network architectures in [14]. After some empirical search, we fix hyper-parameters (tuned separately for different methods), which are used for all the 27 settings.

From the comparison results in Table 3 and 4, the main observations are as follows. 1) The joint-training EBMs out-

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

perform the supervised baseline in 25 out of the 27 settings. Since we perform one run for each setting, this indicates 2 outliers. 2) For a fixed labeling size (as given by the labeling proportion), increasing "U/L" enables the joint-training EBMs to perform better, except in one outlier. 3) The effects of increasing the labeling sizes on the improvements of the joint-training EBMs over the supervised baseline with a fixed "U/L" are mixed. For POS/chunking/NER, the largest improvements are achieved under 2%/10%/100% labeled, respectively. It seems that the working point where an SSL method brings the largest improvement over the supervised baseline is task dependent. Suppose that the working point is indicated by the performance of the supervised baseline, then the SSL method brings the largest effect when the performance of the supervised baseline is moderate, i.e., neither too low nor too high. 4) Joint-training EBMs outperform pre-training EBMs in 23 out of the 27 settings marginally but nearly consistently. A possible explanation is that pre-training is not aware of the labels for the targeted task and is thus weakened for representation learning. In contrast, the marginal likelihood optimized in joint-training is directly related to the targeted task. 5) It seems that the degrees of improvements of the joint-training EBMs over the pre-training EBMs are not much affected when varying the labeling sizes and the "U/L" ratios.

5. CONCLUSION

In this paper, 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. It is revealed that 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.

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

6. REFERENCES

[1] Takeru Miyato, Shin-ichi Maeda, Masanori Koyama, and Shin Ishii, "Virtual adversarial training: a regularization method for supervised and semi-supervised learning," *IEEE transactions on pattern analysis and machine intelligence*, vol. 41, no. 8, pp. 1979–1993, 2018.

[2] Samuli Laine and Timo Aila, "Temporal ensembling for semi-supervised learning," in *ICLR*, 2017.

[3] Antti Tarvainen and Harri Valpola, "Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results," in *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.

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

[7] Xiaojin Zhu, "Semi-supervised learning literature survey," *Technical report, University of Wisconsin-Madison*, 2006.

[8] Ekin D Cubuk, Barret Zoph, Jonathon Shlens, and Quoc V Le, "RandAugment: Practical automated data augmentation with a reduced search space," in *CVPR*, 2020.

[9] Hugo Larochelle, Michael I Mandel, Razvan Pascanu, and Yoshua Bengio, "Learning algorithms for the classification restricted Boltzmann machine," *Journal of Machine Learning Research*, 2012.

[10] Diederik P. Kingma, Danilo Jimenez Rezende, Shakir Mohamed, and Max Welling, "Semi-supervised learning with deep generative models," in *NIPS*, 2014.

[11] Geoffrey E Hinton, Simon Osindero, and Yee Whye Teh, "A fast learning algorithm for deep belief nets," *Neural Computation*, 2006.

[12] Yunfu Song and Zhijian Ou, "Learning neural random fields with inclusive auxiliary generators," *arXiv:1806.00271*, 2018.

[13] Stephen Zhao, Jörn-Henrik Jacobsen, and Will Grathwohl, "Joint energy-based models for semi-supervised classification," in *ICML Workshop on Uncertainty and Robustness in Deep Learning*, 2020.

[14] Yunfu Song, Zhijian Ou, Zitao Liu, and Songfan Yang, "Upgrading CRFs to JRFs and its benefits to sequence modeling and labeling," in *ICASSP*, 2020.

[15] Avital Oliver, Augustus Odena, Colin Raffel, Ekin D Cubuk, and Ian J Goodfellow, "Realistic evaluation of semi-supervised learning algorithms," in *ICLR*, 2018.

[16] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," *The journal of machine learning research*, 2014.

[17] Yann LeCun, Sumit Chopra, Raia Hadsell, M Ranzato, and F Huang, "A tutorial on energy-based learning," *Predicting structured data*, vol. 1, no. 0, 2006.

[18] Daphne Koller and Nir Friedman, *Probabilistic graphical models: principles and techniques*, MIT press, 2009.

[19] Yilun Du and Igor Mordatch, "Implicit generation and generalization in energy-based models," *arXiv:1903.08689*, 2019.

[20] Bin Wang and Zhijian Ou, "Improved training of neural trans-dimensional random field language models with dynamic noise-contrastive estimation," in *SLT*, 2018.

[21] Michael Gutmann and Aapo Hyvärinen, "Noise-contrastive estimation: A new estimation principle for unnormalized statistical models," in *AISTATS*, 2010.

[22] Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer, "Neural architectures for named entity recognition," in *NAACL-HLT*, 2016.

[23] Xuezhe Ma and Eduard Hovy, "End-to-end sequence labeling via bi-directional LSTM-CNNs-CRF," in *ACL*, 2016.

[24] Bin Wang, Zhijian Ou, and Zhiqiang Tan, "Learning trans-dimensional random fields with applications to language modeling," *IEEE transactions on pattern analysis and machine intelligence*, vol. 40, no. 4, pp. 876–890, 2017.

[25] Bin Wang and Zhijian Ou, "Language modeling with neural trans-dimensional random fields," in *ASRU*, 2017.

[26] Bin Wang and Zhijian Ou, "Learning neural trans-dimensional random field language models with noise-contrastive estimation," in *ICASSP*, 2018.

[27] Kai Hu, Zhijian Ou, Min Hu, and Junlan Feng, "Neural CRF transducers for sequence labeling," in *ICASSP*, 2019.

[28] Jost Tobias Springenberg, "Unsupervised and semi-supervised learning with categorical generative adversarial networks," in *ICML*, 2016.

[29] Antti Rasmus, Harri Valpola, Mikko Honkala, Mathias Berglund, and Tapani Raiko, "Semi-supervised learning with ladder networks," in *NIPS*, 2015.

[30] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen, "Improved techniques for training GANs," in *NIPS*, 2016.

[31] Zihang Dai, Zhilin Yang, Fan Yang, William W Cohen, and Ruslan R Salakhutdinov, "Good semi-supervised learning that requires a bad GAN," in *NIPS*, 2017.

[32] Youssef Mroueh, Chun-Liang Li, Tom Sercu, Anant Raj, and Yu Cheng, "Sobolev GAN," in *ICLR*, 2018.

[33] Andrew Y Ng and Michael I Jordan, "On discriminative vs. generative classifiers: A comparison of logistic regression and naive bayes," in *NIPS*, 2002.

[34] Ciprian Chelba, Tomas Mikolov, Mike Schuster, Qi Ge, Thorsten Brants, Phillipp Koehn, and Tony Robinson, "One billion word benchmark for measuring progress in statistical language modeling," in *INTERSPEECH*, 2014.