Risk Minimization for Zero-shot Sequence Labeling

Zechuan Hu[◇‡], Yong Jiang[†*], Nguyen Bach[†], Tao Wang[†], Zhongqiang Huang[†], Fei Huang[†], Kewei Tu[◇*]

[◇]School of Information Science and Technology, ShanghaiTech University
[◇]Shanghai Engineering Research Center of Intelligent Vision and Imaging
[◇]Shanghai Institute of Microsystem and Information Technology, Chinese Academy of Sciences
[◇]University of Chinese Academy of Sciences
[†]DAMO Academy, Alibaba Group
{huzch,tukw}@shanghaitech.edu.cn, yongjiang.jy@alibaba-inc.com
{nguyen.bach,leeo.wangt,z.huang,f.huang}@alibaba-inc.com

Abstract

Zero-shot sequence labeling aims to build a sequence labeler without human-annotated datasets. One straightforward approach is utilizing existing systems (source models) to generate pseudo-labeled datasets and train a target sequence labeler accordingly. However, due to the gap between the source and the target languages/domains, this approach may fail to recover the true labels. In this paper, we propose a novel unified framework for zero-shot sequence labeling with minimum risk training and design a new decomposable risk function that models the relations between the predicted labels from the source models and the true labels. By making the risk function trainable, we draw a connection between minimum risk training and latent variable model learning. We propose a unified learning algorithm based on the expectation maximization (EM) algorithm. We extensively evaluate our proposed approaches on cross-lingual/domain sequence labeling tasks over twenty-one datasets. The results show that our approaches outperform state-of-the-art baseline systems.

1 Introduction

Sequence labeling is an important task in natural language processing. It has many applications such as Part-of-Speech Tagging (POS) (DeRose, 1988; Toutanova et al., 2003) and Named Entity Recognition (NER) (Ratinov and Roth, 2009; Ritter et al., 2011; Lample et al., 2016; Ma and Hovy, 2016; Hu et al., 2020). Approaches to sequence labeling are mostly based on supervised learning, which relies heavily on labeled data. However, the labeled data is generally expensive and hard to obtain (for low-resource languages/domains), which means that these supervised learning approaches fail in many cases.

Learning knowledge from imperfect predictions from other rich-resource sources (such as cross-lingual, cross-domain transfer) (Yarowsky and Ngai, 2001; Guo et al., 2018; Huang et al., 2019; Hu et al., 2021) is a feasible and efficient way to tackle the low-resource problem. It transfers knowledge from rich-resource languages/domains to low-resource ones. One typical approach to this problem is utilizing existing systems to provide predicted results for the zero-shot datasets. However, due to the gap between the source and the target languages/domains, this approach may fail to recover the true labels. Several previous approaches try to alleviate this problem by relying heavily on cross-lingual information (e.g., parallel text (Wang and Manning, 2014; Ni et al., 2017)), labeled data in source languages (Chen et al., 2019), and prior domain knowledge (Yang and Eisenstein, 2015) for different kinds of zero-shot scenarios. However, these approaches are designed to be specific, and might not be generalizable to other kinds of settings where the required resources are expensive to obtain or not available due to data privacy (Wu et al., 2020). Instead, we want a learning framework that can address the zero-shot learning problem in a unified perspective.

In this work, we consider two widely explored settings in which we have access to: 1) the imperfect hard predictions (Rahimi et al., 2019; Lan et al., 2020); 2) the imperfect soft predictions (Wu et al., 2020), produced by one or more source models on target unlabeled data , and propose two novel approaches. We start by introducing a novel approach based on the minimum risk training framework. We design a new decomposable risk function parameterized by a fixed matrix that models the relations between the noisy predictions from the source models and the true labels. We then make the matrix trainable, which leads to further expressiveness and connects minimum risk training to learning latent

*Corresponding authors. ‡Work was done when Zechuan Hu was interning at Alibaba DAMO Academy.

Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, pages 4909–4920
August 1–6, 2021.

variable models. We propose a learning algorithm based on the EM algorithm, which alternates between updating a posterior distribution and optimizing model parameters.

To empirically evaluate our proposed approaches, we extensively conduct experiments on four sequence labeling tasks of twenty-one datasets. Our two proposed approaches, especially the latent variable model, outperform several strong baselines.

2 Background

2.1 Sequence Labeling

Given a sentence $\mathbf{x} = x_1, \ldots, x_n$, its word representations are extracted from the pre-trained embeddings and passed into a sentence encoder such as BiLSTM, Convolutional Neural Networks (CNN) and multilingual BERT (Devlin et al., 2019) to obtain a sequence of contextual features. Without considering the dependencies between predicted labels, the Softmax layer computes the conditional probability as follows,

$$P_{\boldsymbol{\theta}}(\mathbf{y}|\mathbf{x}) = \prod_{i=1}^{n} P_{\boldsymbol{\theta}}(y_i|\mathbf{x})$$

Given the gold sequence $\mathbf{y}^* = y_1^*, \ldots, y_n^*$, the general training objective is to minimize the negative log-likelihood of the sequence,

$$\mathcal{J}(\boldsymbol{\theta}) = -\log P_{\boldsymbol{\theta}}(\mathbf{y}^*|\mathbf{x}) = -\sum_{i=1}^{n} \log P_{\boldsymbol{\theta}}(y_i^*|\mathbf{x})$$

For simplicity, throughout this paper, we assume that all the sequence labelers are based on the Softmax method.

2.2 Cross-Lingual/Domain Transfer

Supervised models fail when labeled data are absent. Learning from imperfect predictions from rich-resource sources is a viable approach to tackle the problem. Generally speaking, there are two settings to obtain the imperfect predictions from: single source and multi source. The simplest single-source approach is to train a single-source model on one source language/domain and use the source model to directly predict labels on the target test data. We name this approach as direct single-source transfer (DT). Another single-source approach is to use the predictions of the source model on a set of unlabeled target data to supervise the training of a target model. With imperfect hard predictions from the source model, the corresponding objective function is the cross-entropy loss between the imperfect hard predictions and the target model's soft predictions,

$$\mathcal{J}(\boldsymbol{\theta}) = -\log P_{\boldsymbol{\theta}}(\hat{\mathbf{y}}|\mathbf{x}) = -\sum_{i=1}^{n} \log P_{\boldsymbol{\theta}}(\hat{y}_i|\mathbf{x})$$

where $\hat{\mathbf{y}}$ denotes the pseudo label sequence of $\mathbf{x}$ predicted by the source model and $\hat{y}_i$ is the pseudo label for position i. With imperfect soft predictions from the source model, the corresponding objective function is the KL-divergence (KL) or mean square error (MSE) loss between the imperfect soft predictions and the target model's soft predictions (knowledge distillation, KD) (Wu et al., 2020).

For multi-source setup, a simple approach contains the following two steps. The first step is to apply DT with each source language to produce predictions on unlabeled target data. The second step is to mix the predictions from all the source models and perform supervised learning of a target model on the mixed pseudo-labeled dataset. However, the mixed pseudo-labeled dataset can be very noisy because predictions from different source models may contradict each other. Similar to single-source setting, a more effective way is aggregating the soft predictions from multiple sources and doing KD (Wu et al., 2020).

3 Methodology

3.1 Minimum Risk Training

In supervised learning, minimum risk training aims to minimize the expected error (risk) concerning the conditional probability,

$$\mathcal{J}(\boldsymbol{\theta}) = \sum_{\mathbf{y} \in \mathcal{Y}(x)} P_{\boldsymbol{\theta}}(\mathbf{y}|\mathbf{x}) R(\mathbf{y}^*, \mathbf{y})$$

where $R(\mathbf{y}^*, \mathbf{y})$ is the risk function that measures the distance between the gold sequence $\mathbf{y}^*$ and the candidate sequence $\mathbf{y}$, and $\mathcal{Y}(\mathbf{x})$ denotes the collection of all the possible label sequences given the sentence $\mathbf{x}$. The risk function can be defined in many ways depending on specific applications, such as the BLEU score in machine translation (Shen et al., 2016). However, in our setting, there are no gold labels to compute $R(\mathbf{y}^*, \mathbf{y})$. Instead, we assume there are multiple pretrained source models which can be used to predict hard labels, and we define the risk function as $R(\hat{\mathbf{y}}, \mathbf{y})$ to measure the difference between pseudo label sequence

$\hat{\mathbf{y}}$ predicted by source models and the candidate sequence $\mathbf{y}$. The objective function becomes,

$$\begin{aligned}\mathcal{J}(\boldsymbol{\theta}) &= \mathbb{E}_{P_{\boldsymbol{\theta}}(\mathbf{y}|\mathbf{x})}[R(\hat{\mathbf{y}}, \mathbf{y})] \\ &= \sum_{\mathbf{y}\in\mathcal{Y}(x)} P_{\boldsymbol{\theta}}(\mathbf{y}|\mathbf{x}) R(\hat{\mathbf{y}}, \mathbf{y})\end{aligned}$$

Conventional minimum risk training is intractable which is mainly due to the combination of two reasons: first, the set of candidate label sequences $\mathcal{Y}(\mathbf{x})$ is exponential in size and intractable to enumerate; second, the risk function is hard to decompose (or indecomposable). To tackle the problem, we define the risk function as a negative probability $-P(\hat{\mathbf{y}}|\mathbf{y})$ that can be fully decomposed by position. The objective function becomes,

$$\begin{aligned}\mathcal{J}(\boldsymbol{\theta}) &= \sum_{\mathbf{y}\in\mathcal{Y}(x)} P_{\boldsymbol{\theta}}(\mathbf{y}|\mathbf{x}) R(\hat{\mathbf{y}}, \mathbf{y}) \\ &= -\sum_{\mathbf{y}\in\mathcal{Y}(x)} P_{\boldsymbol{\theta}}(\mathbf{y}|\mathbf{x}) P_{\boldsymbol{\psi}}(\hat{\mathbf{y}}|\mathbf{y}) \qquad (1) \\ &= -\prod_{i=1}^{n} \sum_{y_i} P_{\boldsymbol{\theta}}(y_i|\mathbf{x}) P_{\boldsymbol{\psi}}(\hat{y}_i|y_i)\end{aligned}$$

We introduce a matrix $\boldsymbol{\psi} \in \mathbb{R}^{K\times K}$ to model $P_{\boldsymbol{\psi}}(\hat{y}_i|y_i)$, where K is the number of labels. Notice that $\boldsymbol{\psi}$ here is a fixed matrix that does not change in training. In the general imperfect predictions learning, it is often implicitly assumed that the prediction from a source model is generally better than uniformly selecting a candidate label at random. Given this prior knowledge, we require $P_{\boldsymbol{\psi}}(\hat{y}_i = k|y_i = k) > \frac{1}{K}$. Therefore, we empirically define matrix $\boldsymbol{\psi}$ as,

$$\boldsymbol{\psi}_{ij} = \begin{cases} \mu & \text{if } i = j\,, \\ \frac{1-\mu}{K-1} & \text{if } i \neq j \end{cases}$$

where $\mu > \frac{1}{K}$ is a hyper-parameter. In the implementation, for convenience, we multiply an identity matrix by a hyper-parameter τ and then apply Softmax operation to every column to obtain the matrix $\boldsymbol{\psi}$.

To further explain $\boldsymbol{\psi}$, we give an example from the perspective of prediction in Table 1. Given a sentence $\mathbf{x}$ = "I cried", a label distribution $P_{\boldsymbol{\theta}}(\mathbf{y}|\mathbf{x})$ for the sentence, a pseudo label sequence $\hat{y} = \{\textbf{Pron}, \textbf{Adj}\}$ predicted by the source model, and two settings μ_1=0.4 and μ_2=1 for $\boldsymbol{\psi}^{(1)}$ and $\boldsymbol{\psi}^{(2)}$ respectively, we compute $P_{\boldsymbol{\theta}}(y_i|\mathbf{x}) \times P_{\boldsymbol{\psi}}(\hat{y}_i|y_i)$ as shown in the table.

$\mathbf{x}$	$P_{\boldsymbol{\theta}}(\mathbf{y}|\mathbf{x})$			$\hat{\mathbf{y}}$
I	0.6	0.3	0.1	**Pron**
cried	0.1	0.6	0.3	**Adj**

Case 1	$\boldsymbol{\psi}^{(1)}$	**Pron**	**Verb**	**Adj**	$P_{\boldsymbol{\theta}}(y_i\|\mathbf{x}) \times P_{\boldsymbol{\psi}}(\hat{y}_i^{(1)}\|y_i)$		
	Pron	0.4	0.3	0.3	0.24	0.09	0.03
	Verb	0.3	0.4	0.3	0.03	0.18	0.12
	Adj	0.3	0.3	0.4	$\mathbf{y}_{pred}$: [**Pron**, **Verb**]		

Case 2	$\boldsymbol{\psi}^{(2)}$	**Pron**	**Verb**	**Adj**	$P_{\boldsymbol{\theta}}(y_i\|\mathbf{x}) \times P_{\boldsymbol{\psi}}(\hat{y}_i^{(2)}\|y_i)$		
	Pron	1	0	0	0.6	0	0
	Verb	0	1	0	0	0	0.3
	Adj	0	0	1	$\mathbf{y}_{pred}$: [**Pron**, **Adj**]		

Table 1: An example of prediction results on two different $\boldsymbol{\psi}$s. Case1 with a less sparse matrix than Case2 obtains a better prediction. $\mathbf{y}_{pred}$ denotes the predictions by sequence labeler using corresponding matrix $\boldsymbol{\psi}$.

Since $\boldsymbol{\psi}^{(2)}$ is an identity matrix, it predicts the label with the largest value at each position. It assigns the wrong label **Adj** to the word "cried" as a consequence. On the contrary, $\boldsymbol{\psi}^{(1)}$ introduces some uncertainties by providing smoothing over the pseudo labels. As a result, it correctly predicts the word "cried" as **Verb**. From the perspective of training, which minimizes $\mathcal{J}(\boldsymbol{\theta})$, if $\boldsymbol{\psi}$ is an identity matrix, then it is a supervised model with $\hat{\mathbf{y}}$ as the supervision signal; on the other hand, if $\boldsymbol{\psi}$ is a uniform matrix, then the supervision signal becomes random and training becomes meaningless.

Extending to Leverage Soft Predictions Previous works shows that the soft predictions from source models can provide more information than the hard predictions (Hinton et al., 2015; Wu et al., 2020). Our novel approach can also easily leverage this information by simply replacing the one-hot pseudo labels with soft probability distributions from source models. The training objective becomes,

$$\mathcal{J}(\boldsymbol{\theta}) = -\prod_{i=1}^{n} \sum_{y_i} P_{\boldsymbol{\theta}}(y_i|\mathbf{x}) \sum_{\hat{y}_i} P_s(\hat{y}_i|\mathbf{x}) P_{\boldsymbol{\psi}}(\hat{y}_i|y_i)$$

where P_s is the source model's soft predictions.

For simplicity, in the rest of this section, we introduce our approaches based on the setup of using one-hot pseudo labels, but all the approaches can be extended to leverage soft predictions in a similar way.

3.2 Minimum Risk Training: A Latent Variable Model Perspective

In this subsection, we instead use a trainable matrix $\boldsymbol{\sigma}$ to model $P_{\boldsymbol{\sigma}}(\hat{\mathbf{y}}|\mathbf{y})$. We initialize $\boldsymbol{\sigma}$ in the same way as $\boldsymbol{\psi}$. Assuming that conditioning on $\mathbf{y}$, $\mathbf{x}$ and $\hat{\mathbf{y}}$ are independent with each other, we find that the non-negative term of equation (1) is a conditional marginal probability defined by a latent variable model in which $\mathbf{y}$ is the latent variable.

$$\sum_{\mathbf{y}\in\mathcal{Y}(x)} P_{\boldsymbol{\theta}}(\mathbf{y}|\mathbf{x})P_{\boldsymbol{\sigma}}(\hat{\mathbf{y}}|\mathbf{y}) = P_{\boldsymbol{\theta},\boldsymbol{\sigma}}(\hat{\mathbf{y}}|\mathbf{x})$$

In latent variable model training, we generally optimize the negative conditional log-likelihood, and the objective function becomes,

$$\begin{aligned}\mathcal{J}(\boldsymbol{\theta},\boldsymbol{\sigma}) &= -\log P_{\boldsymbol{\theta},\boldsymbol{\sigma}}(\hat{\mathbf{y}}|\mathbf{x}) \\ &= -\sum_{i=1}^{n}\log\sum_{y_i} P_{\boldsymbol{\theta}}(y_i|\mathbf{x})P_{\boldsymbol{\sigma}}(\hat{y}_i|y_i)\end{aligned}$$

Interpolation In practice, given a pre-defined hyper-parameter μ, we combine the fixed $P_{\boldsymbol{\psi}}(\hat{y}_i|y_i)$ with the trainable $P_{\boldsymbol{\sigma}}(\hat{y}_i|y_i)$ to get a new probability,

$$P_{\boldsymbol{\phi}}(\hat{y}_i|y_i) = \lambda P_{\boldsymbol{\psi}}(\hat{y}_i|y_i) + (1-\lambda)P_{\boldsymbol{\sigma}}(\hat{y}_i|y_i)$$

where $\lambda \in [0,1]$ is a hyper-parameter, $\boldsymbol{\phi}$ is the combined matrix. If $\lambda = 1$, it denotes the minimum risk training. Otherwise, it denotes the latent variable model.

3.3 From Single-source to Multi-source Setup

By modeling the joint distribution over the pseudo labels which are predicted by U source models on the target unlabeled data, we can easily extend our latent variable model to the multi-source setting. The objective function becomes,

$$\mathcal{J}(\boldsymbol{\theta},\boldsymbol{\phi}) = -\sum_{i=1}^{n}\log\sum_{y_i} P_{\boldsymbol{\theta}}(y_i|\mathbf{x})\prod_{u=1}^{U} P_{\boldsymbol{\phi}}(\hat{y}_i^{(u)}|y_i,u)$$

Our overall architecture of the latent variable model is depicted in Figure 1.

3.4 Optimization

In this section, we propose a unified optimization scheme, which is based on the EM algorithm (Dempster et al., 1977) [1], to learn the parameters

[1] Another approach is to perform direct gradient descent optimization, which we find weaker results. We have a discussion on that in the analysis section.

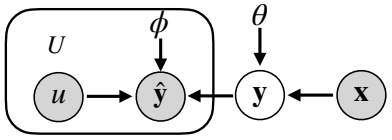

Figure 1: Directed graphical model of our latent variable model.

of the two proposed approaches. The EM algorithm is widely applied to learn parameters in a large family of models with latent variables such as the Gaussian mixture models. It is an iterative approach that has two steps in every iteration, which are the E-step and the M-step. In the E-step, it optimizes a posterior distribution of the latent variables. In the M-step, it estimates the parameters of the latent variable model according to the posterior distribution. As the single-source setup can be seen as a special case, we focus on the multi-source setup to derive the equations. We first introduce $Q(\mathbf{y}) = \prod_i Q(\mathbf{y}_i)$ as a distribution over the latent variable $\mathbf{y}$, and then we derive the upper bound of $\mathcal{J}(\boldsymbol{\theta},\boldsymbol{\phi})$ as follows,

$$\begin{aligned}\mathcal{J}(\boldsymbol{\theta},\boldsymbol{\phi}) &= -\sum_{i=1}^{n}\log\sum_{y_i} P_{\boldsymbol{\theta}}(y_i|\mathbf{x})\prod_{u=1}^{U} P_{\boldsymbol{\phi}}(\hat{y}_i^{(u)}|y_i,u) \\ &= -\sum_{i=1}^{n}\log\sum_{y_i} Q(y_i)\frac{P_{\boldsymbol{\theta}}(y_i|\mathbf{x})\prod_{u=1}^{U} P_{\boldsymbol{\phi}}(\hat{y}_i^{(u)}|y_i,u)}{Q(y_i)} \\ &\le -\sum_{i=1}^{n}\sum_{y_i} Q(y_i)\log\frac{P_{\boldsymbol{\theta}}(y_i|\mathbf{x})\prod_{u=1}^{U} P_{\boldsymbol{\phi}}(\hat{y}_i^{(u)}|y_i,u)}{Q(y_i)} \\ &= -\sum_{i=1}^{n}\mathbb{E}_{Q(\mathbf{y}_i)}\log P_{\boldsymbol{\theta}}(y_i|\mathbf{x})\prod_{u=1}^{U} P_{\boldsymbol{\phi}}(\hat{y}_i^{(u)}|y_i,u) + \mathbf{C}\end{aligned} \tag{2}$$

where $\mathbf{C}$ is a residual term, and $Q(y_i)$ stands for $Q(\mathbf{y}_i = y_i)$. The inequation above is derived from Jensen's inequality. To make the bound tight for particular $\boldsymbol{\theta}$ and $\boldsymbol{\phi}$, we derive $Q(y_i)$ as,

$$Q(y_i) \propto P_{\boldsymbol{\theta}}(y_i|\mathbf{x})\prod_{u=1}^{U} P_{\boldsymbol{\phi}}(\hat{y}_i^{(u)}|y_i,u) \tag{3}$$

We sketch our strategy of parameter update in the t-th iteration as follows,

- **E step**, we compute $Q(y_i)$ using parameters $\boldsymbol{\theta}$ and $\boldsymbol{\phi}$ from the $(t-1)$-th iteration;

- **M step**, we update parameters $\boldsymbol{\theta}$ and $\boldsymbol{\phi}$ together using a gradient-based approach by minimizing the upper bound above. $Q(y_i)$ is fixed in this step and hence we minimize

$$-\sum_{i=1}^{n}\mathbb{E}_{Q(\mathbf{y}_i)}\log P_{\boldsymbol{\theta}}(y_i|\mathbf{x})\prod_{u=1}^{U}P_{\boldsymbol{\phi}}(\hat{y}_i^{(u)}|y_i,u)$$

we repeat the two steps alternately until convergence. We give an overall process for multi-source setup with unlabeled target data in Algorithm 1.

Algorithm 1 Multi-source transfer with latent variable model

Input: unlabeled dataset of target T, U pretrained source models $\{\mathcal{M} = M^{(1)},\ldots,M^{(U)}\}$, U trainable matrices $\{\boldsymbol{\Sigma} = \boldsymbol{\sigma}^{(1)},\ldots,\boldsymbol{\sigma}^{(U)}\}$ and U fixed matrices $\{\boldsymbol{\Psi} = \boldsymbol{\psi}^{(1)},\ldots,\boldsymbol{\psi}^{(U)}\}$, hyper-parameter μ and λ, maximal iterations E for the EM algorithm.
Initialize: initialize $\boldsymbol{\Sigma}$ and $\boldsymbol{\Psi}$ with the same hyper-parameter μ. Initialize $\{\boldsymbol{\Phi} = \boldsymbol{\phi}^{(1)},\ldots,\boldsymbol{\phi}^{(U)}\}$ using λ, $\boldsymbol{\Sigma}$ and $\boldsymbol{\Psi}$. Initialize an empty pseudo label list $\hat{\mathcal{Y}}$, an upper bound loss $l^m = +\infty$, and an overall loss $l^e = +\infty$.
for $u = 1,\ldots,U$ **do**
    Use $M^{(u)}$ to obtain the hard/soft label sequence of the unlabeled data T and append the predictions to the list of pseudo label sequences. $\hat{\mathcal{Y}}$.
end for
Concatenate the unlabeled data T with all pseudo label collections $\hat{\mathcal{Y}}$ to form a new training dataset $\hat{T}$.
for $e = 1,\ldots,E$ **do**
    Compute posterior distribution $Q(\mathbf{y}_i)$ according to formula 3 for each sample $\mathbf{x}$. ▷ E step
    Compute the loss $l^e = \mathcal{J}(\boldsymbol{\phi},\boldsymbol{\theta})$.
    if l^e has no improvement **do**
        End training.
    end if
    repeat ▷ M step
        Compute l^m according to Eq. 2.
        Update $\boldsymbol{\phi}$ and $\boldsymbol{\theta}$.
    Until l^m has no improvement.
end for

3.5 Inference

For inference, we use $Q(\mathbf{y})$ to obtain $\mathbf{y}_{pred}$ [2],

$$\mathbf{y}_{pred} = \underset{\mathbf{y}\in\mathcal{Y}(\mathbf{x})}{\operatorname{argmax}}\, P_{\boldsymbol{\theta}}(\mathbf{y}|\mathbf{x})\prod_{u=1}^{U}P_{\boldsymbol{\phi}}(\hat{\mathbf{y}}^{(u)}|\mathbf{y},u)$$

4 Experiments

We use the multilingual BERT (mBERT) as our word representations[3] as the sentence encoder. Following Wu et al. (2020), the source model are previously trained on its corresponding training data. We use the BIO scheme for CoNLL and OntoNotes NER tasks and Aspect Extraction. We run each model three times and report the average accuracy for the POS tagging task and F1-score for the other tasks.

4.1 Datasets

Cross-Lingual Sequence Labeling We choose three tasks to conduct the cross-lingual sequence labeling task, which are POS tagging, NER, and Aspect Extraction. For the POS tagging task, we use Universal Dependencies treebanks (UD) **v2.4**[4] and randomly select five anguages together with the English dataset. The whole datasets are English (En), Catalan (Ca), Indonesian (Id), Hindi (Hi), Finnish (Fi), and Russian (Ru). For the Aspect Extraction task, we select the restaurant domain over subtask 1 in the SemEval-2016 shared task (Pontiki et al., 2016). For the NER task, we evaluate our models on the CoNLL 2002 and 2003 shared tasks (Tjong Kim Sang, 2002; Tjong Kim Sang and De Meulder, 2003).

Cross-Domain Sequence Labeling We use English portion of the OntoNotes (v5) (Hovy et al., 2006), which contains six domains: broadcast conversation (bc), broadcast news (bn), magazine (mz), newswire (nw), and web (wb).

More details can be found in the Appendix A.1.

4.2 Approaches

Single-source Setup The following approaches are applicable for single-source setup,

- DT: we use the pre-trained source model to directly predict the pseudo labels on the target unlabeled data.
- Hard: we use the pseudo labels from DT on the target unlabeled data to train a new model.

Multi-source Setup The following approaches are applicable for multi-source setup,

- Hard-Cat: we apply DT with all the source languages/domains, mix the resulting pseudo labels from all the sources on the unlabeled target data, and train a new model.
- Hard-Vote: we do majority voting at the token level on the pseudo labels from DT with each source and train a new model.

[2] Another choice is to use $P_{\boldsymbol{\theta}}(\mathbf{y}|\mathbf{x})$, however, we found that utilizing $Q(\mathbf{y})$ generally achieves better performance.

[3] Following previous work (Wu and Dredze, 2019; Wu et al., 2020), we fine-tune mBERT's parameters.

[4] https://universaldependencies.org/

	CoNLL NER					Aspect Extraction					
	English	German	Dutch	Spanish	Avg.	English	Spanish	Dutch	Russian	Turkish	Avg.
Single-source:											
The following approaches have access to hard predictions:											
DT	—	72.17	79.54	75.13	75.61	—	62.48	53.15	46.35	36.42	49.6
Hard	—	72.37	80.01	75.75	76.04	—	63.76	58.28	48.36	40.13	52.63
MRT	—	73.15	80.38	75.87	76.47	—	64.53	59.63	49.89	45.79	54.96
LVM	—	73.36	80.34	76.01	76.57	—	65.03	60.55	50.59	46.40	55.64
The following approaches have access to soft predictions:											
KD-re	—	73.77	80.64	76.02	76.81	—	64.44	58.68	49.54	43.37	54.01
MRT	—	73.67	80.56	76.07	76.77	—	**65.81**	60.91	50.55	45.97	55.81
LVM	—	**73.96**	**80.79**	76.29	**77.01**	—	65.77	60.44	**50.79**	**46.69**	**55.92**
Wu et al. (2020)†	—	73.22	80.89	76.94	77.02	—	—	—	—	—	—
Multi-source:											
The following approaches have access to hard predictions:											
Hard-Vote	77.46	73.52	78.05	76.60	76.41	57.66	65.03	57.23	49.11	45.17	54.84
Hard-Cat	77.13	73.22	78.32	76.81	76.37	55.91	63.13	56.01	49.33	**46.23**	54.12
MRT	77.56	73.81	79.12	76.99	76.87	58.65	65.78	58.56	50.90	43.77	55.53
LVM	78.14	74.17	79.60	77.69	77.40	**61.69**	67.49	59.76	**52.19**	41.93	56.61
The following approaches have access to soft predictions:											
KD-re	78.57	75.25	80.58	77.45	77.96	59.25	65.97	59.70	51.71	44.86	56.30
MRT	78.65	75.83	80.52	**77.74**	78.18	60.66	67.57	59.91	51.59	42.97	56.54
LVM	**79.09**	**76.00**	**83.03**	77.66	**78.94**	60.87	**68.72**	60.14	51.88	42.81	**56.88**
Wu et al. (2020)†	—	74.97	80.70	77.75	—	—	—	—	—	—	—

Table 2: Results on the CoNLL NER and Aspect Extraction tasks. KD-re is our re-implementation for the KD approach (Wu et al., 2020). Their reported results are denoted as † for reference.

	OntoNotes						
	bc	bn	mz	nw	tc	wb	**Avg.**
The following approaches have access to hard predictions:							
Hard-Vote	75.90	84.62	81.93	82.41	68.44	77.65	78.49
Hard-Cat	75.27	84.66	81.88	82.60	71.33	77.12	78.81
MRT	77.03	84.48	84.02	82.90	68.93	77.29	79.11
LVM	75.93	84.76	83.37	83.26	70.56	78.34	79.37
The following approaches have access to soft predictions:							
KD-re	76.20	84.75	82.64	82.92	70.36	**78.49**	79.23
MRT	76.88	84.60	84.01	83.51	70.00	77.71	79.45
LVM	**77.56**	**85.58**	**84.32**	**83.88**	**72.47**	78.03	**80.31**
Lan et al. (2020)†	71.47	79.66	70.71	71.31	52.72	34.06	63.32

Table 3: Multi-source cross-domain results on OntoNotes. KD-re is our re-implementation for the KD approach (Wu et al., 2020). The reported results from Lan et al. (2020) are denoted as † for reference.

Both Setups The following approaches are applicable for both single-/multi-source setups,

- KD-re: to fairly compare with the the KD approach (Wu et al., 2020) in the same settings (such as source model's cross-lingual ability), we re-implement the KD approach and adapt it to all tasks.
- MRT: our minimum risk training approach with a fixed matrix ψ with soft or hard predictions.
- LVM: our latent variable model with parameter ϕ (containing the fixed matrix ψ and the trainable matrix σ) with soft or hard predictions.

We also provide the reported results from existing approaches for reference. Due to different experiment configuration reasons, directly comparing our approaches to their reported results is generally not fair. For the CoNLL NER tasks, we provide the reported results from Wu et al. (2020). For the cross-domain sequence labeling tasks, we provide the reported results from Lan et al. (2020) who learns a consensus network to aggregate predictions from multiple sources.

4.3 Hyper-parameters

Hyper-parameter selection in transfer learning is difficult as no labeled dataset is available for the target language. We select the hyper-parameters only on the development set over the English language and directly use the selected hyper-parameters for the other languages. This may result in sub-optimal

	SINGLE SOURCE							MULTI-SOURCE						
	CA	ID	HI	FI	RU	Avg.		EN	CA	ID	HI	FI	RU	Avg.
The following approaches have access to hard predictions:														
DT	86.65	84.37	67.14	76.03	88.02	80.44	Hard-Vote	82.90	86.21	85.87	74.10	78.86	89.77	82.95
Hard	86.73	84.52	67.34	76.32	88.21	80.62	Hard-Cat	83.04	85.80	86.13	74.55	78.95	90.22	83.11
MRT	86.78	84.61	67.63	76.97	88.36	80.87	MRT	82.72	85.64	86.14	74.48	78.91	89.90	82.97
LVM	86.80	84.64	67.65	77.04	88.37	80.90	LVM	83.08	85.76	86.11	75.35	79.12	89.98	83.23
The following approaches have access to soft predictions:														
KD-re	86.84	84.93	67.62	76.51	**88.53**	80.89	KD-re	83.81	86.46	86.25	74.46	79.01	**90.56**	83.43
MRT	86.57	84.65	**68.44**	77.51	88.40	81.11	MRT	83.60	85.54	**86.60**	75.07	**79.89**	90.24	83.49
LVM	86.78	84.89	68.31	**77.68**	88.45	**81.22**	LVM	83.85	**86.76**	86.50	**75.41**	79.60	90.23	**83.73**

Table 4: Results on the POS tagging tasks. KD-re is our re-implementation for the KD approach (Wu et al., 2020).

performance but is more realistic. In latent variable model training, the latent variable is generally very flexible, which may result in sub-optimal performance. Therefore, the initialization of the latent variable is very crucial. In practice, we find that the best strategy is to initialize μ of $\boldsymbol{\psi}$ with a large value (e.g., 0.9) and μ of $\boldsymbol{\sigma}$ with a small value (e.g., 0.3), and anneal λ from 1 to 0. At the early stage of training, this initialization offers a strong prior for the encoder which can keep the encoder from going in a bad direction; and at later stages of training, the warmed-up encoder can better guide the training of $\boldsymbol{\phi}$ and vice versa. In this way, the encoder and $\boldsymbol{\phi}$ can achieve a good balance during training. More details of the hyper-parameters can be found in the Appendix A.2.

4.4 Results and Observations

For the single-source setting, we use English as the source language and the others as the unlabeled target languages. In the multi-source setting, we repeat our experiments multiple times, each time with a language as the target and the others as the sources. We evaluate all approaches on the CoNLL, Aspect Extraction, OntoNotes, and POS tagging. We report the results in Table 2, 3 and 4 [5].

Observation #1 Our two approaches outperform several strong baselines on all the tasks and all the scenarios (single-/multi-source scenarios with soft/hard predictions), especially the multi-source scenario, which demonstrates the effectiveness of the two proposed approaches. It shows that modeling this kind of relation is fairly important, which helps to recover the true labels from noisy data. Meanwhile, introducing uncertainties for the relations between the predicted labels from the source models and the true labels in both training and prediction processes significantly benefit our approaches.

Observation #2 Our LVM approach achieves overall improvements over the MRT approach on all tasks. It suggests that our LVM approach learns the relations between predicted labels from the source models and true labels better than MRT.

Other Minor Observations First, all the approaches that use unlabeled target data for training outperform DT. It suggests that leveraging the unlabeled target data (which may contain knowledge of the target language/domain) in training for zero-shot transfer learning does help. Comparing the approaches that leverage soft instead of hard predictions from sources, the former generally outperform the latter. It suggests that soft predictions can still provide useful knowledge for samples with incorrect hard predictions. The reported results from Lan et al. (2020) are significantly worse. We speculate the reason is that they leverage poor embeddings and different encoders (BiLSTM-CRF). KD-re outperforms our approaches on Ca and Id of POS tagging task on the single-source setting, but its advantage is not statistically significant.

5 Analysis

We conduct the analysis on the multi-source setting with soft predictions from sources for its better performance.

Big Data Performance We experiment with our two models and the KD-re baseline on big target training data on the POS tagging task. We ran-

[5] We utilize almost stochastic dominance (ASD) test (Dror et al., 2019) to compare the best score of our approaches and the score of the best performing baselines. We mark the the highest score as **bold** if its superiority is significant ($p < 0.05$) and underline otherwise.

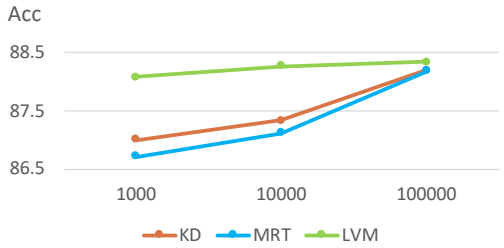

Figure 2: The multi-source performance of Ca datasets by varying different sizes on the POS tagging task.

		EN	DE	NL	ES	Avg.
MRT	Direct‡	**78.83**	75.27	80.22	**77.76**	78.02
	EM†	78.65	**75.83**	**80.52**	77.74	**78.19**
LVM	Direct‡	78.79	75.48	81.29	**77.93**	78.37
	EM†	**79.09**	**76.00**	**83.03**	77.66	**78.95**

Table 5: Results on comparisons between EM algorithm and direct gradient-based strategy. ‡ denotes the results of direct gradient-based strategy and † denotes the results of EM algorithm that are from Table 2.

domly select 100000 sentences (without labels) for the Wikipedia-003 section of the Ca language on the CoNLL 2017 shared task (Ginter et al., 2017). We randomly select 1000, 10000, and 100000 sentences to train these three approaches, evaluate on the UD test set for each of the three languages respectively, and show the results in Figure 2. It shows that our latent variable model outperforms the other two approaches over all the settings. Though KD outperform MRT with less than 10000 sentences, but MRT has comparable result with enough unlabeled data. Besides, with more unlabeled data used for training, each model further gains a considerable boost.

Comparison to Direct Gradient Optimization Our two proposed approaches can also be optimized directly by any gradient-based approach, such as the AdamW optimizer (Loshchilov and Hutter, 2018). We use the two proposed approaches to compare the performance of the direct gradient-based training strategy and the EM algorithm. We conduct the experiments on our two proposed approaches on CoNLL NER task on the multi-source setting. We show the results in Table 5. It shows that the EM algorithm outperforms direct gradient-based training for our approaches, which is slightly different from previous findings (Berg-Kirkpatrick et al., 2010).

		EN	DE	NL	ES	Avg.
MRT	hard-EM	**79.65**	75.02	80.26	77.00	77.98
	soft-EM	78.65	**75.83**	**80.52**	**77.74**	**78.19**
LVM	hard-EM	78.36	**76.01**	81.98	77.46	78.45
	soft-EM	**79.09**	76.00	**83.03**	**77.66**	**78.95**

Table 6: Results on hard-EM experiments. The results of soft-EM are from Table 2 of the body.

Comparison to Hard EM In this part, we compare our optimization strategy (soft-EM) with the hard-EM approach. Instead of computing a dense vector for $Q(\mathbf{y}_i)$, hard-EM computes a one-hot vector. We conduct the experiments on our two proposed approaches on the CoNLL NER task on the multi-source setting. The results are shown in Table 6. It shows that soft-EM gains slightly improvement over hard-EM on the MRT approach, but differs significantly from hard-EM on our LVM approach.

Impact of Matrix $\boldsymbol{\psi}$ We analyze the relation between the performance and different initialization of $\boldsymbol{\psi}$. We experiment with the MRT approach in the single-source setup with soft predictions on NER tasks and Figure 3 shows the results. The best value of τ is 2 for De and 3 for the others (resulting in $\mu = 0.43$ and 0.67 respectively[6]), which shows that the uncertainties introduced by a smooth $\boldsymbol{\psi}$ can effectively boost the model's performance. On the other hand, setting $\boldsymbol{\psi}$ to a nearly identity matrix with $\tau = 10$ leads to worse scores.

6 Related Work

Cross-lingual/domain Sequence Labeling Recent works on cross-lingual transfer mainly have two scenarios: the single-source cross-lingual transfer (Yarowsky and Ngai, 2001; Wang and Manning, 2014; Huang et al., 2019) and the multi-source cross-lingual transfer (Täckström et al., 2012; Guo et al., 2018; Rahimi et al., 2019; Hu et al., 2021). Wu et al. (2020) propose a knowledge distillation approach to further leveraging unlabeled target data and achieve the state-of-the-art results. Hu et al. (2021) propose a multi-view framework to selectively transfer knowledge from multiple sources by utilizing a small amount of labeled dataset. Cross-domain adaption is widely studied (Steedman et al.,

[6]The CoNLL NER datasets have 11 labels (9 entity labels, a padding label and an ending label).

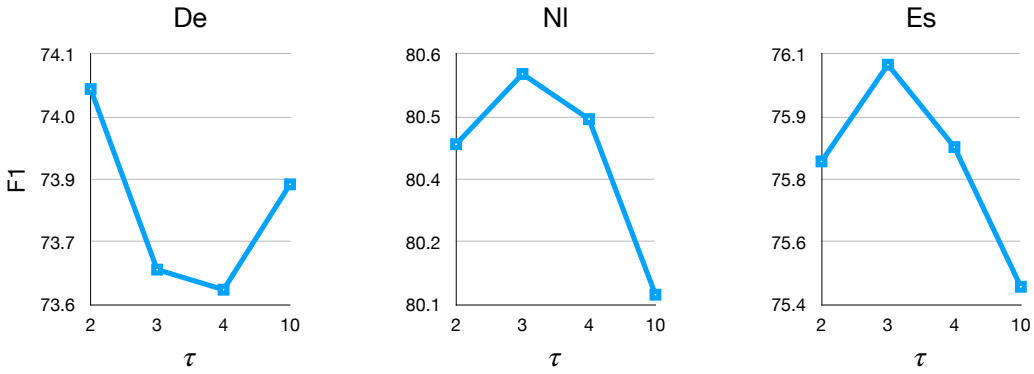

Figure 3: The performance of MRT approach in single-source setup with soft predictions on three NER datasets by varying different τ.

2003). Existing works include bootstrapping approaches (Ruder and Plank, 2018), mixture-of-experts (Guo et al., 2018; Wright and Augenstein, 2020), and consensus network (Lan et al., 2020). Other previous work (Kim et al., 2017; Guo et al., 2018; Huang et al., 2019) utilized labeled data in the source domain to learn desired information. However, our proposed approaches do not require any source labeled data or parallel texts.

Contextual Multilingual Embeddings Embeddings like mBERT (Devlin et al., 2019), XLM (CONNEAU and Lample, 2019) and XLM-R (Conneau et al., 2020) which are trained on many languages, make great progress on cross-lingual learning for multiple NLP tasks. Recent works (Wu and Dredze, 2019; Pires et al., 2019) show the strong cross-lingual ability of the contextual multilingual embeddings.

7 Conclusion

In this paper, we propose two approaches to the zero-shot sequence labeling problem. Our MRT approach uses a fixed matrix to model the relations between the predicted labels from the source models and the true labels. Our LVM approach uses trainable matrices to model these label relations. We extensively verify the effectiveness of our approaches on both single-source and multi-source transfer over both cross-lingual and cross-domain sequence labeling problems. Experiments show that MRT and LVM generally bring significant improvements over previous state-of-the-art approaches on twenty-one datasets.

Acknowledgement

This work was supported by the National Natural Science Foundation of China (61976139) and by Alibaba Group through Alibaba Innovative Research Program.

References

Taylor Berg-Kirkpatrick, Alexandre Bouchard-Côté, John DeNero, and Dan Klein. 2010. Painless unsupervised learning with features. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 582–590, Los Angeles, California. Association for Computational Linguistics.

Xilun Chen, Ahmed Hassan Awadallah, Hany Hassan, Wei Wang, and Claire Cardie. 2019. Multi-source cross-lingual model transfer: Learning what to share. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3098–3112, Florence, Italy. Association for Computational Linguistics.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. In *ACL*.

Alexis CONNEAU and Guillaume Lample. 2019. Cross-lingual language model pretraining. In *Advances in Neural Information Processing Systems 32*, pages 7059–7069. Curran Associates, Inc.

A. P. Dempster, N. M. Laird, and D. B. Rubin. 1977. Maximum likelihood from incomplete data via the em algorithm. In *Journal of the Royal Statistical Society. Series B*, volume 39, pages 1–38.

Steven J. DeRose. 1988. Grammatical category disambiguation by statistical optimization. *Computational Linguistics*, 14(1):31–39.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference*

of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Rotem Dror, Segev Shlomov, and Roi Reichart. 2019. Deep dominance - how to properly compare deep neural models. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2773–2785, Florence, Italy. Association for Computational Linguistics.

Filip Ginter, Jan Hajič, Juhani Luotolahti, Milan Straka, and Daniel Zeman. 2017. CoNLL 2017 shared task - automatically annotated raw texts and word embeddings. LINDAT/CLARIAH-CZ digital library at the Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University.

Jiang Guo, Darsh Shah, and Regina Barzilay. 2018. Multi-source domain adaptation with mixture of experts. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4694–4703, Brussels, Belgium. Association for Computational Linguistics.

Geoffrey Hinton, Oriol Vinyals, and Jeffrey Dean. 2015. Distilling the knowledge in a neural network. In *NIPS Deep Learning and Representation Learning Workshop*.

Eduard Hovy, Mitchell Marcus, Martha Palmer, Lance Ramshaw, and Ralph Weischedel. 2006. OntoNotes: The 90% solution. In *Proceedings of the Human Language Technology Conference of the NAACL, Companion Volume: Short Papers*, pages 57–60, New York City, USA. Association for Computational Linguistics.

Zechuan Hu, Yong Jiang, Nguyen Bach, Tao Wang, Zhongqiang Huang, Fei Huang, and Kewei Tu. 2020. An investigation of potential function designs for neural CRF. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 2600–2609, Online. Association for Computational Linguistics.

Zechuan Hu, Yong Jiang, Nguyen Bach, Tao Wang, Zhongqiang Huang, Fei Huang, and Kewei Tu. 2021. Multi-View Cross-Lingual Structured Prediction with Minimum Supervision. In *the Joint Conference of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (**ACL-IJCNLP 2021**)*. Association for Computational Linguistics.

Lifu Huang, Heng Ji, and Jonathan May. 2019. Cross-lingual multi-level adversarial transfer to enhance low-resource name tagging. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 3823–3833, Minneapolis, Minnesota. Association for Computational Linguistics.

Joo-Kyung Kim, Young-Bum Kim, Ruhi Sarikaya, and Eric Fosler-Lussier. 2017. Cross-lingual transfer learning for POS tagging without cross-lingual resources. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2832–2838, Copenhagen, Denmark. Association for Computational Linguistics.

Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. 2016. Neural architectures for named entity recognition. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 260–270, San Diego, California. Association for Computational Linguistics.

Ouyu Lan, Xiao Huang, Bill Yuchen Lin, He Jiang, Liyuan Liu, and Xiang Ren. 2020. Learning to contextually aggregate multi-source supervision for sequence labeling. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2134–2146, Online. Association for Computational Linguistics.

Ilya Loshchilov and Frank Hutter. 2018. Fixing weight decay regularization in adam.

Xuezhe Ma and Eduard Hovy. 2016. End-to-end sequence labeling via bi-directional LSTM-CNNs-CRF. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1064–1074, Berlin, Germany. Association for Computational Linguistics.

Jian Ni, Georgiana Dinu, and Radu Florian. 2017. Weakly supervised cross-lingual named entity recognition via effective annotation and representation projection. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1470–1480, Vancouver, Canada. Association for Computational Linguistics.

Telmo Pires, Eva Schlinger, and Dan Garrette. 2019. How multilingual is multilingual BERT? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4996–5001, Florence, Italy. Association for Computational Linguistics.

Maria Pontiki, Dimitris Galanis, Haris Papageorgiou, Ion Androutsopoulos, Suresh Manandhar, Mohammad AL-Smadi, Mahmoud Al-Ayyoub, Yanyan Zhao, Bing Qin, Orphée De Clercq, Véronique Hoste, Marianna Apidianaki, Xavier Tannier, Natalia Loukachevitch, Evgeniy Kotelnikov, Nuria Bel, Salud María Jiménez-Zafra, and Gülşen Eryiğit. 2016. SemEval-2016 task 5: Aspect based sentiment analysis. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 19–30, San Diego, California. Association for Computational Linguistics.

Afshin Rahimi, Yuan Li, and Trevor Cohn. 2019. Massively multilingual transfer for NER. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 151–164, Florence, Italy. Association for Computational Linguistics.

Lev Ratinov and Dan Roth. 2009. Design challenges and misconceptions in named entity recognition. In *Proceedings of the Thirteenth Conference on Computational Natural Language Learning (CoNLL-2009)*, pages 147–155, Boulder, Colorado. Association for Computational Linguistics.

Alan Ritter, Sam Clark, Mausam, and Oren Etzioni. 2011. Named entity recognition in tweets: An experimental study. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 1524–1534, Edinburgh, Scotland, UK. Association for Computational Linguistics.

Sebastian Ruder and Barbara Plank. 2018. Strong baselines for neural semi-supervised learning under domain shift. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1044–1054, Melbourne, Australia. Association for Computational Linguistics.

Shiqi Shen, Yong Cheng, Zhongjun He, Wei He, Hua Wu, Maosong Sun, and Yang Liu. 2016. Minimum risk training for neural machine translation. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1683–1692, Berlin, Germany. Association for Computational Linguistics.

Mark Steedman, Rebecca Hwa, Stephen Clark, Miles Osborne, Anoop Sarkar, Julia Hockenmaier, Paul Ruhlen, Steven Baker, and Jeremiah Crim. 2003. Example selection for bootstrapping statistical parsers. In *Proceedings of the 2003 Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*, pages 236–243.

Oscar Täckström, Ryan McDonald, and Jakob Uszkoreit. 2012. Cross-lingual word clusters for direct transfer of linguistic structure. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 477–487, Montréal, Canada. Association for Computational Linguistics.

Erik F. Tjong Kim Sang. 2002. Introduction to the CoNLL-2002 shared task: Language-independent named entity recognition. In *COLING-02: The 6th Conference on Natural Language Learning 2002 (CoNLL-2002)*.

Erik F. Tjong Kim Sang and Fien De Meulder. 2003. Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition. In *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003*, pages 142–147.

Kristina Toutanova, Dan Klein, Christopher D. Manning, and Yoram Singer. 2003. Feature-rich part-of-speech tagging with a cyclic dependency network. In *Proceedings of the 2003 Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*, pages 252–259.

Mengqiu Wang and Christopher D. Manning. 2014. Cross-lingual projected expectation regularization for weakly supervised learning. *Transactions of the Association for Computational Linguistics*, 2:55–66.

Dustin Wright and Isabelle Augenstein. 2020. Transformer based multi-source domain adaptation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7963–7974, Online. Association for Computational Linguistics.

Qianhui Wu, Zijia Lin, Börje Karlsson, Jian-Guang Lou, and Biqing Huang. 2020. Single-/multi-source cross-lingual NER via teacher-student learning on unlabeled data in target language. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6505–6514, Online. Association for Computational Linguistics.

Shijie Wu and Mark Dredze. 2019. Beto, bentz, becas: The surprising cross-lingual effectiveness of BERT. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 833–844, Hong Kong, China. Association for Computational Linguistics.

Yi Yang and Jacob Eisenstein. 2015. Unsupervised multi-domain adaptation with feature embeddings. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 672–682, Denver, Colorado. Association for Computational Linguistics.

David Yarowsky and Grace Ngai. 2001. Inducing multilingual POS taggers and NP bracketers via robust projection across aligned corpora. In *Second Meeting of the North American Chapter of the Association for Computational Linguistics*.

A Experimental details

A.1 Datasets

CoNLL CoNLL is a dataset for the NER task. We evaluate our models on the CoNLL 2002 and 2003 shared tasks (Tjong Kim Sang, 2002; Tjong Kim Sang and De Meulder, 2003), which contain four languages: English, German, Dutch, and Spanish. Every dataset contains four types of named

entities: Organization, Location, Person, and Miscellaneous.

Aspect Extraction We select the restaurant domain over subtask 1 in the SemEval-2016 shared task (Pontiki et al., 2016).

OntoNotes We use English portion of the OntoNotes (v5) (Hovy et al., 2006), which contains six domains: broadcast conversation (bc), broadcast news (bn), magazine (mz), newswire (nw), and web (wb). It is a NER task which contains 18 entity types.

A.2 Hyper-parameter setting

We select the hyper-parameters according to the strategy which is described in the main paper. For multi-source cross-lingual/domain tasks, we select hyper-parameters based on the performance on the English development set and apply them to other target languages. For single-source cross-lingual/domain tasks, we simply use the same hyper-parameter as multi-source setting. In the inference step, We use $P_{\boldsymbol{\theta}}(\mathbf{y}|\mathbf{x})$ in single-source cross-lingual/domain and $Q(\mathbf{y})$ in multi-source cross-lingual/domain to predict the label sequence. We empirically set the learning rate of mBERT as 2e-5 and the learning rate of ϕ and $\boldsymbol{\phi}$ as 2e-4 for multi-source setup and 2e-5 for single-source setup. We train each model for three epochs. We tune the following hyper-parameters.

τ and $\hat{\tau}$ for initializing matrices τ and $\hat{\tau}$ are used to initialize the matrices $\boldsymbol{\psi}$ and $\boldsymbol{\sigma}$ in our minimum risk training and latent variable model approaches respectively. Due to different sizes of the label sets for different tasks, the range of selection is different. Take the CoNLL NER tasks for example, we tune it in the range of $\{1, 2, 3, 4, 10\}$ for $\hat{\tau}$ in $\boldsymbol{\psi}$ in MRT and LVM, and $\{1, 2, 3, 4, 10\}$ for τ in $\boldsymbol{\sigma}$ in LVM. The CoNLL NER tasks have 11 labels (9 entity labels, a padding label and an ending label), which means $\mu \in \{0.21, 0.43, 0.67, 0.85, 1.0\}$. We list the value we select for each task below:

- CoNLL NER: $\tau = 3$ and $\hat{\tau} = 2$ for single-source setup; $\tau = 2$ and $\hat{\tau} = 10$ for multi-source setup.
- AE: $\tau = 3$ and $\hat{\tau} = 4$ for single-source setup;; $\tau = 2$ and $\hat{\tau} = 10$ for multi-source setup.
- POS: $\tau = 4$ and $\hat{\tau} = 2$ for single-source setup; $\tau = 2$ and $\hat{\tau} = 10$ for multi-source setup.
- OntoNotes: $\tau = 4$ and $\hat{\tau} = 10$.