

## Research Article

# A Unified Model Using Distantly Supervised Data and Cross-Domain Data in NER

Yun Hu <sup>1,2</sup>, Hao He,<sup>1</sup> Zhengfei Chen,<sup>3</sup> Qingmeng Zhu,<sup>1</sup> and Changwen Zheng <sup>1</sup>

<sup>1</sup>*Institute of Software, Chinese Academy of Sciences, Haidian, Beijing 100190, China*

<sup>2</sup>*University of Chinese Academy of Sciences, Haidian, Beijing 100190, China*

<sup>3</sup>*Shenzhen Power Supply Bureau Co., Ltd., Shenzhen 518001, China*

Correspondence should be addressed to Yun Hu; [huyun2016@iscas.ac.cn](mailto:huyun2016@iscas.ac.cn) and Changwen Zheng; [changwen097@gmail.com](mailto:changwen097@gmail.com)

Received 2 December 2021; Accepted 29 April 2022; Published 29 May 2022

Academic Editor: Shawkat ali

Copyright © 2022 Yun Hu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Named entity recognition (NER) systems are often realized by supervised methods that require large hand-annotated data. When the hand-annotated data is limited, distantly supervised (DS) data and cross-domain (CD) data are usually used separately to improve the performance. The distantly supervised data can provide in-domain dictionary information, and the hand-annotated cross-domain information can be provided by cross-domain data. These two types of information are complementary. However, there are two problems required to be solved before using directly. First, the distantly supervised data may contain a lot of noise. Second, directly using cross-domain data may degrade performance due to the distribution mismatching problem. In this paper, we propose a unified model named PARE (PARTial learning and REinforcement learning). The PARE model can simultaneously use distantly supervised data and cross-domain data as external data. The model uses the partial learning method with a new label strategy to better handle the noise in distantly supervised data. The reinforcement learning method is used to alleviate the distribution mismatching problem in cross-domain data. Experiments in three datasets show that our model outperforms other baseline models. Besides, our model can be used in the situation where no hand-annotated in-domain data is provided.

## 1. Introduction

Name entity recognition [1] as a fundamental natural language processing (NLP) task has received significant attention. The NER system tries to label each word in sentences with predefined types, such as brand and product. The results of NER can be used in many downstream NLP tasks, such as coreference resolution [2], relation extraction [3], and question answering [4]. The supervised methods are often used to realize the NER system [5–7]. However, supervised methods often require large annotated data. In practice, obtaining hand-annotated data can be a very expensive and laborious process. When limited hand-annotated data is accessible, distantly supervised data and cross-domain data are widely used as external resources [8–11].

The distantly supervised data is obtained by the matching method. An example is shown in Figure 1. The word “Samsung” is matched by the entity dictionary. More

in-domain data can be obtained through distant supervision. Cross-domain data can provide compensative information. The cross-domain learning methods utilize the high resource domain information to improve the low resource domain performance [10]. It is natural to combine these two types of data.

However, there are some problems with directly using two types of data. First, in distantly supervised data, when a word is not matched by the entity dictionary, the word has no clear label. The matching method can lead to many errors. In Figure 1 (we use Chinese as a case of study; however, in order to better make readers understand, Chinese characters are not allowed in the figure, and we translate Chinese to English), the label of the word “mobile” should be product. Previous work uses a word-level method and a sentence-level method to alleviate errors [8]. In the word-level method, the partial learning method is used. In the sentence-level method, the reinforcement

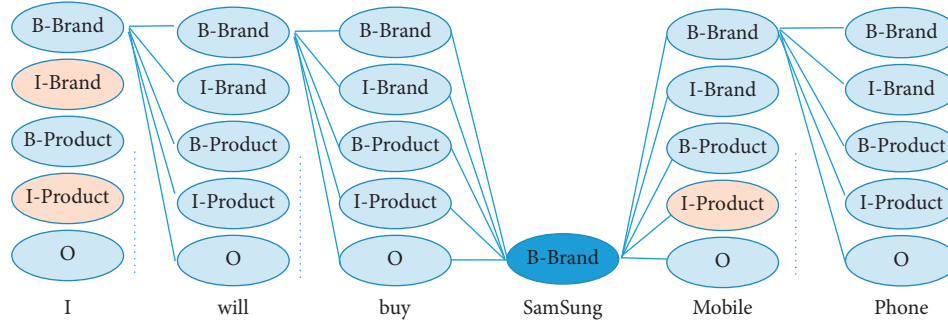


FIGURE 1: An example of the improved label strategy in partial learning method. The NER system tries to recognize brands and products in e-commerce data. The brown blocks are impossible tags. “B-Brand” means beginning of a brand, “I-Brand” means inside of a brand, “B-Product” means beginning of a product, “I-Product” means inside of a product, and “O” means outside of entity.

learning method is used. In the traditional partial learning method, the labels of the words are the whole label set when the words are not contained in the entity dictionary. For example, the label of “mobile” has 5 candidate labels ( $\{\text{“B-Brand,” “I-Brand,” “B-Product,” “I-Product,” “O”}\}$ ). However, the traditional label strategy does not consider label features in NER. For example, the label of “mobile” cannot be “I-Brand.” Second, in cross-domain learning, the parameter sharing method is a widely used method [10, 12]. However, directly using the parameter sharing method may lead to performance reduction due to the distribution mismatching [13]. Previous works mainly focus on using different network architectures to capture private features and shared features [14]. In addition, these works cannot handle the situation where no hand-annotated domain data is provided.

In this paper, we explore a PARE model which can simultaneously use distantly supervised data and cross-domain data as external data. For distantly supervised data, we use a new label strategy considering label features of NER. The new label strategy can reduce redundant labels in the partial learning method. In cross-domain data, the reinforcement learning method is used to alleviate the distribution mismatching problem in the parameter sharing method. We evaluate our PARE model in three datasets and show that our methods outperform other baseline models. Besides, our PARE model can be used in the situation where no hand-annotated in-domain data is provided. We only use distantly data and cross-domain data as inputs and achieve competitive results. Contributions of our work can be summarized as follows:

- (i) We propose a PARE model. The PARE model can use distantly supervised annotated data and cross-domain data as external data simultaneously in Chinese NER.
- (ii) We propose an improved partial learning method to process distantly supervised NER data, which can better process the noise in distantly supervised data.
- (iii) We use the reinforcement learning method to process cross-domain NER data, which can alleviate the distribution mismatching problem in the parameter sharing method.

## 2. PARE Model

The architecture of the PARE model is shown in Figure 2. The model can be divided into three parts: core NER part, DS data selector part, and CD data selector part. The improved partial learning method is used in the core NER part to reduce the noise in distant supervision. The DS data selector part and CD data selector part are similar to those in [8, 13]. The DS data selector removes the noise sentence by reinforcement learning. The CD data selector selects the relevant sentence in the cross-domain dataset, which reduces the domain distribution mismatching. Through reinforcement learning, we can use the distantly supervised data and cross-domain data as external data simultaneously.

**2.1. Core NER.** The core NER is based on the LSTM-CRF model [7], which is widely used in named entity recognition. The core NER part contains three subparts: embedding layer, Bi-LSTM layer, and CRF layer. The input of the model is a sentence  $x = \{x_1, x_2, \dots, x_n\}$ .  $x_i$  is the word in sentence and  $x$  can be from hand-annotated data, distantly supervised data, or cross-domain data. The output of the model is  $y = \{y_1, y_2, \dots, y_n\}$ , which is the label of sentence  $x$ . To better process the distantly supervised data, a new label strategy is proposed in core NER.

**2.1.1. Embedding Layer.** We use the embedding layer as the first step of a neural network model. The embedding layer maps the word into a low dimensional dense vector, which contains the semantics of the word. The embedding vector  $e_i$  can be obtained as

$$e_i = \text{Embedding}(x_i; \theta_e), \quad (1)$$

where  $x_i$  is the input word and  $\theta_e$  is the embedding table. The embedding layer can also use an additional contextualized word embedding such as BERT [15].  $e_i$  is concatenated with the output of BERT.

**2.1.2. Feature Extractor.** The feature extractor uses the output of word embedding. We use the Bi-LSTM as the feature extractor. The LSTM can handle gradient vanishing/

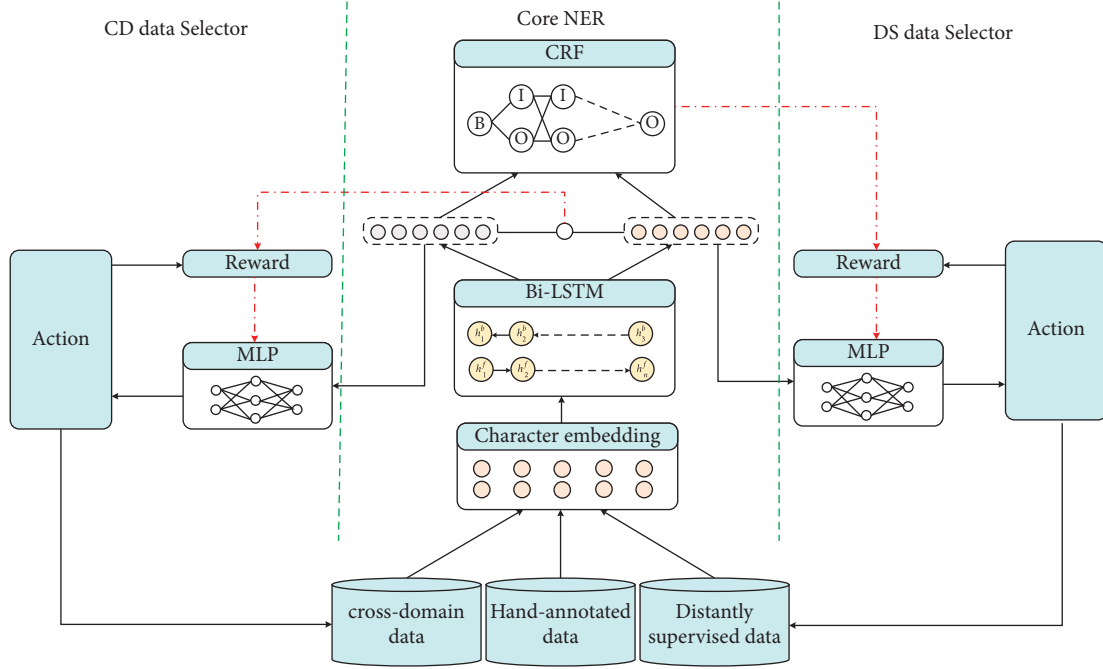


FIGURE 2: The architecture of the PARE model.

exploding problems well as shown in previous work [16]. A common LSTM unit is composed of a cell, an input gate, an output gate, and a forget gate. The Bi-LSTM concatenates forward LSTM output and backward LSTM output to capture information from context [17]. The output of Bi-LSTM  $h_i$  can be represented as

$$h_i = \text{Bi-LSTM}(e_i; \theta_l), \quad (2)$$

where  $\theta_l$  is the parameters in Bi-LSTM.

**2.1.3. CRF.** We use Conditional Random Fields (CRF) to predict the label sequence, because the neighborhood information can be considered in CRF. For example, CRF considers that the “I-Product” cannot be behind the “B-Brand.” For a sentence  $x$  and one possible label sequence  $y$ , we define the score to be

$$\text{score}(x, y) = \sum_{i=0}^n A_{y_i, y_{i+1}} + \sum_{i=1}^n P_{i, y_i}, \quad (3)$$

where  $A$  is a matrix of transition score.  $A_{i,j}$  represents the score of a transition from the tag  $i$  to tag  $j$ .  $P$  is the matrix of the output score.  $P$  matrix can be obtained by using a feed-forward neural network after the Bi-LSTM output. The probability of the sequence  $y$  can be obtained by

$$p(y | x) = \frac{e^{\text{score}(x, y)}}{\sum_{\bar{y} \in A(y)} e^{\text{score}(x, \bar{y})}}, \quad (4)$$

where  $A(y)$  is all candidate label sequences.

The traditional CRF can process hand-annotated data and cross-domain data well because every word in hand-annotated data and cross-domain data has an explicit label.

For distantly annotated data, if the words are in the dictionary, the words have explicit labels. In traditional label strategy, if the words are not in the entity dictionary, the labels are the whole label set. For example, the word “mobile” will have 5 labels. The partial learning method can better process these label strategies. For the input sentence  $x$ , there will be a set of possible label sequence  $C(y)$ , because every word will have several labels. We compute the probability of the all possible predicted sequence  $C(y)$  by summing the probability of each possible label sequence  $\hat{y}$ :

$$p(C(y) | x) = \frac{\sum_{\hat{y} \in C(y)} e^{\text{score}(x, \hat{y})}}{\sum_{\bar{y} \in A(y)} e^{\text{score}(x, \bar{y})}}. \quad (5)$$

However, the previous label strategy does not consider label features in NER. Consider that the whole label set may lead to many redundant labels. The redundant labels may harm the model performance. In this paper, we use a new label strategy. The new label strategy removes some impossible labels. Some label features are shown as follows:

- (i) The start label cannot be “I-XXX.” “XXX” represents an entity type, like “Brand.” For example, the label of “I” cannot be “I-Brand” or “I-Product” in Figure 1.
- (ii) The label of punctuation is “O.”
- (iii) The label “I-YYY” cannot be after the label “B-XXX” or “I-XXX.” “YYY” represents another entity type. For example, the label of “mobile” cannot be “I-Product” in Figure 1.

By using the new label strategy, the number of possible label sequences ( $|C(y)|$ ) may reduce, and the performance may increase. During training, the log probability of the

correct sequence  $\log(p(C(y)|x))$  is maximized. During decoding, the output sequence is given by

$$y^* = \arg \max_{y \in A(y)} \text{score}(x, y). \quad (6)$$

**2.2. Data Selector.** The data selector uses reinforcement learning method to select proper sentences. The overall training procedure is shown in Algorithm 1. In each epoch, we first select the cross-domain data and then select the distantly supervised data. Before doing the selection process, the hand-annotated data is merged with distantly supervised data and cross-domain data to obtain the merged data, respectively. To obtain more feedback, we divide the merged data into many random-size bags. Every sentence in bags obtains a state. During the selection process, for each sentence from distantly supervised data, the DS data selector has an action to decide whether to select the sentence or not. The CD data selector has an action to decide whether to select the cross-domain sentences or not. We directly select the sentences from the hand-annotated dataset. After doing the selection process in a bag, different selections will lead to different rewards. The DS reward is used as the feedback of the DS data selector and updates the DS data selector. The CD data selector is also updated through the domain reward. The goal of data selectors is to maximize the reward when the data selectors take action. After doing the whole selection process in dataset, the selected sentences are used to train the core NER, which can obtain the new state and reward in reinforcement learning. After the training processing, the core NER can be used as the NER system, which combines the distantly supervised data and cross-domain data information.

Some details of the reinforcement learning method are shown as follows. We use superscript  $t$  to discriminate different data and different data selectors.  $t$  can be  $\{ds, do\}$ , where  $ds, do$  mean distantly supervised and cross-domain, respectively.

**2.2.1. State.** The state  $s^t$  can be the input of the selector network. For different data inputs, the model has the same pattern to obtain the state. The state  $s^t$  of the sentence contains the following information: (a) the serialized feature representation, which is extracted by the Bi-LSTM; (b) the label score is equal to  $P$  in equation (3).

**2.2.2. Policy Network.** For each state  $s^t$ , the action space  $a^t$  is  $\{0, 1\}$ . “1” represents selecting the sentence, and “0” represents not selecting the sentence. For DS annotated data, the action  $a_i^{ds}$  is obtained from the DS data selector. The action  $a_i^{do}$  is obtained from the CD data selector in cross-domain data. The data selector  $\pi_{\theta}^t(s_i^t, a_i^t)$  is a multilayer perceptron with the parameter  $\theta^t$ . A logistic function is used as the policy function:

$$\pi_{\theta}^t(s_i^t, a_i^t) = a_i \sigma(W^t * s_i^t + b^t) + (1 - a_i^t)(1 - \sigma(W^t * s_i^t + b^t)), \quad (7)$$

where  $W^t$  and  $b^t$  are the parameters of selector.  $\sigma$  is the sigmoid function.

**2.2.3. Reward.** The reward is used to evaluate the quality of the selection. The reinforcement learning method in cross-domain has a different reward compared with reinforcement learning method in distant supervision because the goal of the selection is different. The goal of the DS data selector is to select the sentences which have less noise. The goal of the CD data selector is to select sentences that are relevant to hand-annotated data.

For distantly supervised data, when all the selections in the current merged data bag are finished, a delayed average reward is obtained. The selected distantly supervised sentences and hand-annotated sentences are used to compute the reward:

$$r^{ds} = \frac{1}{|\widehat{X}^{ds}| + |X^{ha}|} \left( \sum \log p(C(y) | \widehat{x}^{ds}) + \sum \log p(y | x^{ha}) \right), \quad (8)$$

where  $\widehat{x}^{ds}$  is the selected sentences,  $|\widehat{X}^{ds}|$  is the number of the selected sentences,  $x^{ha}$  is the hand-annotated sentences, and  $|X^{ha}|$  is the number of the selected sentences.  $\log p(C(y) | \widehat{x}^{ds})$  and  $\log p(y | x^{ha})$  are computed by equation (5).

The goal of the CD data selector is to select relevant sentences that fit the distribution of the target domain. For cross-domain data, we also obtained a delayed average reward. A reward is the distance between the selected domain data and hand-annotated data. We describe a simple distance in this paper, and the complex and better distance will be discussed in the future. The reward is set as

$$r^{do} = -d\left(\Phi\left(\widehat{X}^{do}\right), \Phi\left(X^{ha}\right)\right). \quad (9)$$

The negative sign means that the distance value is small when the selected sentences are similar to the hand-annotated sentences. We use the maximum mean discrepancy to measure the distance.  $\Phi\left(\widehat{X}^{do}\right)$  is the elementwise average of selected cross-domain sentences’ state  $\widehat{s}^{do}$ .  $\Phi\left(X^{ha}\right)$  is the elementwise average of hand-annotated sentences’ state  $s^{ha}$ .

Selector training: the policy gradient method [18] is used to optimize the policy network. The agents obtain the reward when all the actions have been done. For each bag, the feedback of every action  $r^t(a_i^t)$  is equal to the average reward  $r^t$ . The selector is updated as follows:

$$\theta^t = \theta^t + \alpha \sum_{i=1}^{|\widehat{X}^t|} r^t(a_i^t) \nabla_{\theta} \log \pi_{\theta}^t(s_i^t, a_i^t), \quad (10)$$

where  $|\widehat{X}^t|$  is the number of selected sentences from distantly supervised data or cross-domain data.

**2.3. No Hand-Annotated In-Domain Data.** Our model can be used in the situation where only distantly supervised data and cross-domain data are provided. The motivation is that

```

(i) Input: Hand data, cross-domain data, and distantly supervised data
(ii) Output: Trained PARE model
(1)   for each epoch do
(2)     Merge hand data and cross-domain data
(3)     Divide merged data into many small bag1s
(4)     for each bag1 in bag1s do
(5)       for each sentence in bag1 do
(6)         Obtain the sentence state  $s^t$ 
(7)         if sentence in Hand data then
(8)            $a^t = 1$ 
(9)         else
(10)          Select cross-domain data through  $\pi_{\theta}^t(s_i^t, a_i^t)$ 
(11)        end if
(12)      end for
(13)      Obtain reward  $r^{\text{do}}$ 
(14)      Optimize CD data selector through (10)
(15)    end for
(16)    Merge hand data and distantly supervised data
(17)    Divide merged data into many small bag2s
(18)    for each bag2 in bag2s do
(19)      for each sentence in bag2 do
(20)        Obtain the sentence state  $s^t$ 
(21)        if sentence in Hand data then
(22)           $a^t = 1$ 
(23)        else
(24)          Select distantly supervised data through  $\pi_{\theta}^t(s_i^t, a_i^t)$ 
(25)        end if
(26)      end for
(27)      Obtain reward  $r^{\text{ds}}$ 
(28)      Optimize DS data selector through (10)
(29)    end for
(30)    Train the core NER using selected data
(31)  end for

```

ALGORITHM 1: Overall training procedure.

we can use cross-domain data instead of in-domain hand-annotated data. The details are shown in Algorithm 2. First, we use the distantly supervised data to select cross-domain data because the cross-domain data selection does not require the hand-annotated in-domain data to obtain the reward. Then, we use the selected cross-domain data to select the distantly supervised data. Finally, we use the selected cross-domain data and selected distantly supervised data to train the core NER part.

### 3. Experiments and Results

**3.1. Datasets.** Three hand-annotated in-domain datasets are used to evaluate our methods: e-commerce (EC) dataset, news dataset, and broadcast conversation (BC) dataset. The number of sentences is shown in Table 1. The EC dataset and news dataset are from [8] (the only difference is that we consider three types of entities (Brand, Product, and Model) in the EC dataset because cross-domain EC data has these common label types). The cross-domain data of the EC dataset is the Taobao dataset in [19]. For news data, the cross-domain data is the Chinese web text domain data from OntoNotes [20]. We also use a new dataset to evaluate our methods. For BC data, we randomly select 3000 sentences as

train data, 500 sentences as development data, and 1000 sentences as test data from the OntoNotes Chinese broadcast conversation dataset [20]. The distantly supervised data is obtained from the rest of the OntoNotes Chinese broadcast conversation dataset. The construction method is similar to that in [8]. The cross-domain data is from OntoNotes Chinese web data [20]. 6000 cross-domain sentences are used.

**3.2. Parameters Setting.** The pretrained embedding is shown to be helpful in previous works [7]. In core NER, the embedding is pretrained using word2vec [21] in a large user-generated text corpus. The embedding dimension is 100. The dimension of the LSTM is set to 100. The optimization method is RMSprop [22]. The initial learning rate is 0.001. The dropout rate is 0.2. The minibatch is 128. The max-epoch iteration is 500. For DS data selector and CD data selector, the dimension of the hidden layer in the multilayer perceptron is 100. The optimization method is Adam [23]. The initial learning rate is 0.001.

**3.3. Results.** For evaluation, the entity-level metrics of Precision (P), Recall (R), and their F1 score are used in this paper.



```

(i) Input: Cross-domain data and distantly supervised data
(ii) Output: Trained PARE model
(1) Merge distantly supervised data and cross-domain data.
(2) for Each epoch do
(3)   Divide the merge data into many small bag1s
(4)   for Each bag1 in bag1s do
(5)     for Each sentence in bag1 do
(6)       Obtain the sentence state  $s^t$ 
(7)       if sentence in distantly supervised data then
(8)          $a^t = 1$ 
(9)       else
(10)        Select cross-domain data through  $\pi_{\theta}^t(s_i^t, a_i^t)$ 
(11)      end if
(12)    end for
(13)    Obtain reward  $r^{do}$ 
(14)    Optimize CD data selector through (10)
(15)  end for
(16)  Merge the selected cross-domain sentences and distantly supervised data
(17)  Divide the merged data into many small bag2s
(18)  for Each bag2 in bag2s do
(19)    for Each sentence in bag2 do
(20)      Obtain the sentence state  $s^t$ 
(21)      if sentence in cross-domain data then
(22)         $a^t = 1$ 
(23)      else
(24)        Select distantly supervised data through  $\pi_{\theta}^t(s_i^t, a_i^t)$ 
(25)      end if
(26)    end for
(27)    Obtain reward  $r^{ds}$ 
(28)    Optimize DS data selector through (10)
(29)  end for
(30)  Train the core NER using selected data
(31) end for

```

ALGORITHM 2: The training procedure in no in-domain hand-annotated data.

TABLE 1: The number of sentences in the different corpora.

	# rain	# Dev	# Test	# DS	# CD
EC	1200	400	800	2560	5999
News	3000	3328	3186	3722	8405
BC	3000	500	1000	3000	8405

3.3.1. *Baseline.* Eight baselines are considered to compare with our proposed model:

- (i) LSTM-CRF (Hand) [7]: the model only uses the hand-annotated in-domain data as input.
- (ii) DS-PA-RL (Hand + DS) [8]: the model uses distantly supervised data as external data. The traditional partial learning method is used to process the distantly supervised data. The reinforcement learning method is used to delete the noise sentences. The code is the same as in [8].
- (iii) CD-SHA (Hand + CD) [24]: the model uses cross-domain data as external data. We mix cross-domain data with hand-annotation in-domain data as input data and then share all model parameters to train the model.

- (iv) CD-ADV (Hand + CD) [14]: the model uses cross-domain data as external data. An adversarial network is used to process the private and shared information between target domain and source domain.
- (v) BERT-CRF (Hand) [15]: the model uses the hand-annotated in-domain data as input. The BERT is used as a context embedding encoder, and CRF is used as a decoder.
- (vi) SoftLexicon (Hand) [25]: the model uses the hand-annotated in-domain data as input and can capture the lexicon information through a segmentation label set.
- (vii) SCDL (Hand + DS) [26]: the distantly supervised data is used as external data, and a self-collaborative denoising learning method is used to handle label noise in distantly supervised data.
- (viii) Multicell (Hand + CD) [27]: the cross-domain data is used as external data, and different label types are processed by different cells in Multicell LSTM.

Four systems can be built based on our method:

TABLE 2: The results of models.

Model	Data	EC			News			BC		
		P	R	F	P	R	F	P	R	F
Word embedding based										
LSTM-CRF [7]	Hand	50.64	54.83	52.65	63.33	68.63	65.87	65.00	58.65	61.66
DS-PA-RL [8]	Hand + DS	58.93	60.69	59.80	75.94	76.10	76.02	73.94	78.95	76.36
CD-SHA [24]	Hand + CD	65.70	65.17	65.44	75.85	71.31	73.51	82.28	73.64	77.72
CD-ADV [14]	Hand + CD	64.09	63.49	63.79	71.03	<b>76.15</b>	73.50	72.79	74.43	73.60
DS-IPA-RL	Hand + DS	62.82	62.53	62.67	78.62	75.80	77.18	76.92	78.43	77.67
CD-RL	Hand + DS	66.82	66.67	66.74	83.63	74.30	78.69	82.30	75.81	78.92
PARE	Hand + DS + CD	<b>66.99</b>	<b>68.78</b>	<b>67.87</b>	<b>83.87</b>	76.12	<b>79.80</b>	<b>82.56</b>	<b>79.06</b>	<b>80.77</b>
BERT based										
BERT-CRF [15]	Hand	80.14	80.30	80.22	98.17	97.36	97.76	94.64	92.99	93.81
SoftLexicon [25]	Hand	81.15	81.53	81.34	98.18	97.53	97.85	94.94	93.34	94.13
SCDL [26]	Hand + DS	81.33	81.64	81.48	98.08	97.74	97.91	94.95	93.54	94.24
Multicell [27]	Hand + CD	81.82	81.69	81.75	98.17	97.76	97.96	94.89	93.65	94.27
PARE-BERT	Hand + DS + CD	<b>81.84</b>	<b>82.32</b>	<b>82.08</b>	<b>98.20</b>	<b>97.76</b>	<b>97.98</b>	<b>94.96</b>	<b>94.36</b>	<b>94.66</b>

- (i) DS-IPA-RL (Hand + DS): the model uses the distantly supervised data as external data. Compared with [8], the model uses the new label strategies in the partial learning method.
- (ii) CD-RL (Hand + CD): the model uses cross-domain data as external data. The reinforcement learning method is used to select the relevant sentences in high resource domain.
- (iii) PARE (Hand + DS + CD): the PARE model uses distantly supervised data and cross-domain data simultaneously.
- (iv) PARE-BERT (Hand + DS + CD): the PARE-BERT uses the output of BERT as contextualized word embedding. Distantly supervised data and cross-domain data are used simultaneously.

The results of the models are shown in Table 2. We first analyze the models using traditional word embedding. From Table 2, the models using external data yield improvement compared with the LSTM-CRF model using hand-annotated data as input. This shows that using external data can be very helpful when hand-annotated data is limited. Different ways of using external resources will lead to different effects. The DS-IPA-RL model outperforms the DS-PA-RL model, showing that the new label strategies are helpful in the partial learning method. The CD-ADV model obtains worse results than the CD-SHA model. The reason may be that the CD-ADV model cannot utilize the CRF information sharing when the output label of the source domain and target domain are the same. The CD-RL model achieves the best performance compared with other cross-domain models. The PARE model achieves the best performance of all models. This indicates that our PARE model can utilize different external data well.

The Pretrained Language Model (PLM) has achieved competitive performance in the NER task, and our model is orthogonal with PLM. The results of models using BERT embedding are shown in Table 2. We can observe that our model can also improve the performance in the BERT situation. The results also show that PARE-BERT outperforms

other baseline models using BERT. However, the improvement rate is smaller compared with the models without BERT. The reason may be that the language model has contained much information and the information gain brought by the unified model is relatively small.

To better analyze the different parts of the model, we divide our PARE model into some small parts to evaluate our methods. First, we will analyze the effect of using the label strategy in the partial learning method. Then, we evaluate the reinforcement learning method in cross-domain learning methods. Finally, we show the results in the situation where no hand-annotated data is provided.

**3.3.2. Improved Partial Learning.** We compare the new label strategy with two label strategies and do not consider the influence of reinforcement learning. The first is the FA label strategy. In the FA label strategy, for the word that is not contained in the entity dictionary, the label is "O." To utilize the FA data, we use the simple LSTM-CRF model [7]. The second is the PA label strategy. For the word that is not contained in the entity dictionary, the label is the whole label sets. We use the LSTM-CRF-PA model to utilize the PA data [8]. The results are shown in Table 3. The results show that the models using external data do not always obtain better results than the model only using hand-annotated data. Different label strategies achieve various results. The model using the IPA label strategy obtains the best results compared with the other models using hand-annotated and distantly supervised data. These facts show that label strategy is very important when distantly supervised data is used, and our new label strategy can be more effective than other label strategies.

**3.3.3. Cross-Domain Learning.** In Table 4, we explore the performance of the model compared with the CD-SHA model. The CD-SHA model uses different amounts of randomly selected domain data. From the table, the CD-RL model outperforms the CD-SHA model using all domain data. This indicates that the reinforcement learning method

TABLE 3: The performance of improved partial learning.

Model	Data	EC	News	BC
LSTM-CRF	Hand	52.65	65.87	61.66
LSTM-CRF	Hand + FA	51.21	61.21	71.31
LSTM-CRF-PA	Hand + PA	51.91	69.54	71.53
LSTM-IPA	Hand + IPA	<b>59.11</b>	<b>75.09</b>	<b>75.18</b>

TABLE 4: The performance of using cross-domain data by reinforcement learning method.

Model	Data	EC	News	BC
CD-SHA	Hand + 0% CD	52.65	65.87	61.66
CD-SHA	Hand + 25% CD	62.01	68.98	71.43
CD-SHA	Hand + 50% CD	64.47	71.44	74.07
CD-SHA	Hand + 75% CD	65.94	73.06	78.86
CD-SHA	Hand + 100% CD	65.44	73.51	78.23
CD-RL	Hand + CD RE	<b>67.46</b>	<b>78.69</b>	<b>78.92</b>

can select relevant sentences and reduce negative transfer. The table can also show that the CD-RL model achieves the best results compared with the CD-SHA model using different amounts of domain data. These facts show that reinforcement learning selection is better than random selection.

**3.3.4. No Hand-Annotated In-Domain Data.** Our PARE model can be extended to no hand-annotated situation. The model only uses distantly supervised data and cross-domain data as input, which greatly reduces the time of manual annotation. The training method is shown in Section 2.3. We compare our methods with three baselines:

- (i) *LSTM-IPA*. The model uses only distantly supervised data as input. The improved label strategy is used to process the distantly supervised data.
- (ii) *SHA*. The model uses cross-domain data as input to train the model, and we directly transfer the model from the source domain to the target domain.
- (iii) *SHA-IPA*. The model uses distantly supervised data and domain data as input. The model is the same as that in [8], and we use domain data instead of hand-annotated in-domain data.

The results in three different datasets are shown in Table 5. From the table, we find that directly using distantly supervised data achieves very low performance. The reason can be that the data only from the distantly supervised method contains a lot of noise, especially the low coverage. The results of the SHA model are slightly better than those of the LSTM-IPA model. The SHA-IPA achieves the best results among all models. These facts show that our methods can work well in the situation where no hand-annotated data is provided.

#### 4. Error Analysis

To show the detail of our PARE model outperforming the baseline model, we do some error analysis. We consider five error types:

TABLE 5: Test results for PARE model in no hand-annotated data.

Model	Data	EC	News	BC
LSTM-PA	DS	21.78	6.30	10.34
LSTM-CRF	CD	51.60	50.98	36.36
SHA-IPA	DS + CD	53.54	62.93	57.54
PARE	DS RE + CD RE	<b>56.41</b>	<b>65.44</b>	<b>60.65</b>

Containing: Gold/PARE model: Men large [ripped jeans] (Product) Error prediction: Men large ripped [jeans] (Product)
Be-contained: Gold/PARE model: Want to buy [sippy cup] (Product) for baby Error prediction: Want to buy [sippy cup for baby] (Product)
Cross: Gold/PARE model: I want to buy [toilet powder] (Product) and baby [toilet powder] (Product) Error prediction: I want to buy [toilet] (Product) [powder] (Product) and baby toilet [powder] (Product)
Type-error: Gold/PARE model: I want to see how [UGREEN] (Brand) is Error prediction: I want to see how [UGREEN] (Product) is
No-cross: Gold/PARE Model: Can I use the coupon together Error prediction: Can I use the [coupon] (Product) together

FIGURE 3: The case of different error types.

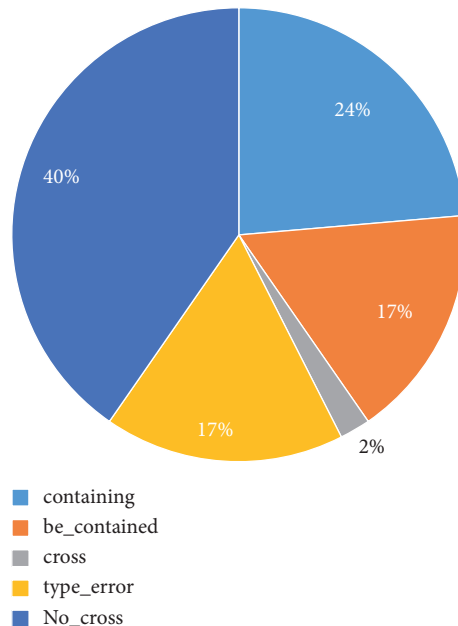


FIGURE 4: The rate of different error types in EC.

- (i) Containing: the gold entity boundary contains the predicted entity boundary.
- (ii) Be-contained: the gold entity boundary is contained by the predicted entity boundary.
- (iii) Cross: the gold entity boundary is crossed with the predicted entity boundary.
- (iv) Type-error: the gold entity boundary is the same as the predicted entity boundary, and the gold entity



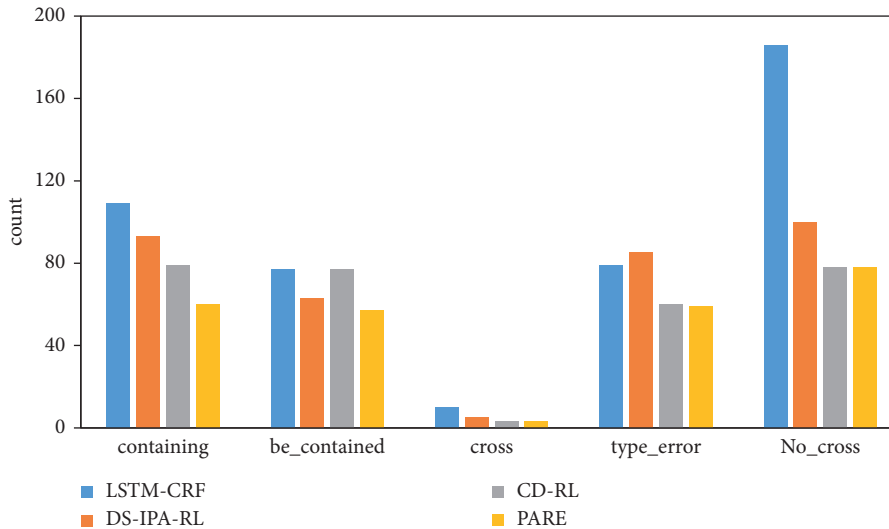


FIGURE 5: The performance of the PARE model in different error types.

has a different type compared with the predicted entity.

- (v) No-cross: there is no boundary crossing between the gold entity and the predicted entity; that is, the model predicts the unlabeled entity or forgets to predict the gold entity.

Some cases in EC are shown in Figure 3, and the rate of the LSTM-CRF model in different error types is shown in Figure 4. The main error comes from no-cross error, accounting for 40%. It may be a good choice for a model to focus on no-cross error. In Figure 5, we show the performance of the PARE model in different error types. The model incorporated distantly supervised data, and cross-domain data can greatly reduce the number of no-cross errors.

## 5. Related Work

Recently, most NER systems are based on supervised methods, such as conditional random field model and neural network model. Conditional random field model relies heavily on hand-crafted features [5, 28]. The features of the sequence can be extracted automatically in the neural network model [6, 7, 25, 29–31]. Recently, the methods of using pretraining language models have become the mainstream for solving named entity recognition. The pretraining language models [15, 32–34] can better capture the context information of the current token. In this paper, we build our models based on the LSTM-CRF model [7] and BERT [15], which are prevalent neural network models.

In the named entity recognition task, the distantly supervised data is a widely used external data [8]. A lot of methods focus on reducing the noise in the distantly supervised NER data [8, 9, 19, 35]. Yang et al. introduced a partial learning method to process the distantly supervised data [8]. Recently, Peng et al. have explored a positive-unlabeled method to process distantly supervised NER data

in English [9]. Jie et al. proposed a cross-training model that can process the incomplete annotation situation [19]. A structural causal model is introduced to solve the dictionary biased problem in [35]. Zhang et al. proposed self-collaborative denoising learning methods, which jointly train two teacher-student networks in a mutually beneficial manner to iteratively perform noisy label refinery [26]. However, all these methods do not consider the NER label features. In this paper, we build our model based on [8] and propose a new label strategy using the NER label feature.

Parameters sharing method is a widely used method in cross-domain learning methods [10, 12, 27, 36]. Directly transferring cross-domain data information may lead to performance decline [37, 38]. Data selection can be useful in cross-domain learning methods for preventing negative transfer from irrelevant sentences [39, 40]. Besides, Jia and Zhang proposed the Multicell Compositional LSTM for domain adaptation. Motivated by [13], we introduce the cross-domain data selection to NER and use distantly supervised data as another external data.

In recent years, the reinforcement learning method has been used widely in natural language processing [8, 13, 41]. Feng et al. used instance selection as a reinforcement learning process in relation classification [41]. Yang et al. used the reinforcement learning method to select sentences from the distantly supervised data in NER [8]. Liu et al. used the reinforcement learning method for domain adaptation in POS tagging, dependency parsing, and sentiment analysis [13]. In this paper, we explore a PARE model that used distantly supervised data and cross-domain data simultaneously by the reinforcement learning method.

## 6. Conclusion

In this paper, we propose a PARE model that can simultaneously use distantly supervised data and cross-domain data as external data in NER. In the PARE model, first, we introduce a new label strategy to traditional partial

annotated learning. The strategy can reduce the redundancy path to improve performance. Then, we introduce reinforcement learning to reduce the noise in distantly supervised data and distribution differences in cross-domain data. Finally, the model can be used in the situation where no hand-annotated data is provided. The experiments in three datasets show that our method can work perfectly.

## Data Availability

The dataset used in this work can be found at “rainarch/DSNER/tree/master/data,” “[https://github.com/allanj/ner\\_in\\_complete\\_annotation/tree/master/data](https://github.com/allanj/ner_in_complete_annotation/tree/master/data),” and “<https://catalog.ldc.upenn.edu/LDC2013T19>.”

## Conflicts of Interest

The authors declare no conflicts of interest.

## Acknowledgments

This research was supported by the Key Deployment Projects of the Chinese Academy of Sciences (Grant no. ZDRW-XH-2021-3).

## References

- [1] E. F. T. Kim Sang and F. De Meulder, “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*, Edmonton Canada, May 2003.
- [2] K. Lee, L. He, M. Lewis, and L. Zettlemoyer, “End-to-end neural coreference resolution,” in *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pp. 188–197, Association for Computational Linguistics, Copenhagen, Denmark, September 2017.
- [3] R. Bunescu and R. Mooney, “A shortest path dependency kernel for relation extraction,” in *Proceedings of the Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*, Vancouver British Columbia Canada, October 2005.
- [4] X. Yao and B. Van Durme, “Information extraction over structured data: question answering with freebase,” in *Proceedings of the Fifty Second Annual Meeting of the Association for Computational Linguistics*, vol. 1, pp. 956–966, Association for Computational Linguistics, Baltimore, MD, USA, June 2014.
- [5] J. D. Lafferty, A. McCallum, and F. C. N. Pereira, “Conditional random fields: probabilistic models for segmenting and labeling sequence data,” in *Proceedings of the Eighteenth International Conference on Machine Learning*, pp. 282–289, Williams College, Williamstown, MA, USA, July 2001.
- [6] R. Collobert, J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu, and P. Kuksa, “Natural language processing (almost) from scratch,” *Journal of Machine Learning Research*, vol. 12, pp. 2493–2537, 2011.
- [7] G. Lample, M. Ballesteros, S. Subramanian, K. Kawakami, and C. Dyer, “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*, pp. 260–270, Association for Computational Linguistics, San Diego, CA, USA, March 2016.
- [8] Y. Yang, W. Chen, Z. Li, Z. He, and M. Zhang, “Distantly supervised NER with partial annotation learning and reinforcement learning,” in *Proceedings of the Twenty Seventh International Conference on Computational Linguistics*, pp. 2159–2169, Association for Computational Linguistics, Santa Fe, NM, USA, August 2018.
- [9] M. Peng, X. Xing, Qi Zhang, J. Fu, and X. Huang, “Distantly supervised named entity recognition using positive-unlabeled learning,” in *Proceedings of the Fifty Seventh Annual Meeting of the Association for Computational Linguistics*, pp. 2409–2419, Association for Computational Linguistics, Florence, Italy, July 2019.
- [10] J. Y. Lee, F. Dernoncourt, and S. Peter, “Transfer Learning for Named-Entity Recognition with Neural Networks,” 2017, <https://arxiv.org/abs/1705.06273>.
- [11] C. Jia, X. Liang, and Y. Zhang, “Cross-domain NER using cross-domain language modeling,” in *Proceedings of the Fifty Seventh Annual Meeting of the Association for Computational Linguistics*, pp. 2464–2474, Association for Computational Linguistics, Florence, Italy, July 2019.
- [12] B. Y. Lin and W. Lu, “Neural adaptation layers for cross-domain named entity recognition,” in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pp. 2012–2022, Association for Computational Linguistics, Brussels, Belgium, November 2018.
- [13] M. Liu, Y. Song, H. Zou, and T. Zhang, “Reinforced training data selection for domain adaptation,” in *Proceedings of the Fifty Seventh Annual Meeting of the Association for Computational Linguistics*, pp. 1957–1968, Association for Computational Linguistics, Florence, Italy, July 2019.
- [14] P. Cao, Y. Chen, K. Liu, J. Zhao, and S. Liu, “Adversarial transfer learning for Chinese named entity recognition with self-attention mechanism,” in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pp. 182–192, Association for Computational Linguistics, Brussels, Belgium, November 2018.
- [15] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “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, NAACL-HLT 2019*, J. Burstein, C. Doran, and T. Solorio, Eds., vol. 1, pp. 4171–4186, Association for Computational Linguistics, Minneapolis, MN, USA, June 2019.
- [16] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.
- [17] A. Graves and J. Schmidhuber, “Framewise phoneme classification with bidirectional lstm networks,” in *Proceedings of the IEEE International Joint Conference on Neural Networks, 2005. IJCNN '05. Proceedings*, vol. 4, pp. 2047–2052, Montreal, Canada, July 2005.
- [18] R. S. Sutton, D. A. McAllester, S. P. Singh, and Y. Mansour, “Policy gradient methods for reinforcement learning with function approximation,” in *Proceedings of the Advances in Neural Information Processing Systems*, pp. 1057–1063, Denver CO, USA, November 2000.
- [19] Z. Jie, P. Xie, W. Lu, R. Ding, and L. Li, “Better modeling of incomplete annotations for named entity recognition,” in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, vol. 1, pp. 729–734, Association for Computational Linguistics, Minneapolis, Minnesota, June 2019.

- [20] R. Weischedel, M. Palmer, M. Marcus et al., *Ontonotes Release 5.0 Ldc2013t19*, Linguistic Data Consortium, Philadelphia, PA, USA, 2013.
- [21] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Proceedings of the Advances in Neural Information Processing Systems*, pp. 3111–3119, Lake Tahoe NV, USA, December 2013.
- [22] T. Tieleman and G. Hinton, "Lecture 6.5-rmsprop: divide the gradient by a running average of its recent magnitude," *COURSERA: Neural networks for machine learning*, vol. 4, no. 2, pp. 26–31, 2012.
- [23] D. P. Kingma and B. Jimmy, "Adam: A Method for Stochastic Optimization," *Computer Science*, 2014.
- [24] Z. Yang, R. Salakhutdinov, and W. W. Cohen, "Transfer Learning for Sequence Tagging with Hierarchical Recurrent Networks," 2016, <https://arxiv.org/abs/1703.06345>.
- [25] R. Ma, M. Peng, Qi Zhang, Z. Wei, and X. Huang, "Simplify the usage of lexicon in Chinese NER," in *Proceedings of the Fifty Eighth Annual Meeting of the Association for Computational Linguistics, ACL 2020*, D. Jurafsky, J. Chai, N. Schluter, and J. R. Tetreault, Eds., pp. 5951–5960, Association for Computational Linguistics, July 2020.
- [26] X. Zhang, B. Yu, T. Liu et al., "Improving distantly-supervised named entity recognition with self-collaborative denoising learning," in *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021*, M. Francine Moens, X. Huang, L. Specia, and S. Wen-Tau Yih, Eds., Article ID 10746, November, 2021.
- [27] C. Jia and Y. Zhang, "Multi-cell compositional LSTM for NER domain adaptation," in *Proceedings of the Fifty Eighth Annual Meeting of the Association for Computational Linguistics, ACL 2020*, D. Jurafsky, J. Chai, N. Schluter, and J. R. Tetreault, Eds., pp. 5906–5917, Association for Computational Linguistics, July 2020.
- [28] W. Chen, Y. Zhang, and H. Isahara, "Chinese named entity recognition with conditional random fields," in *Proceedings of the Fifth SIGHAN Workshop on Chinese Language Processing*, pp. 118–121, Association for Computational Linguistics, Sydney, Australia, July 2006.
- [29] C. Dong, J. Zhang, C. Zong, M. Hattori, and H. Di, "Character-based lstm-crf with radical-level features for Chinese named entity recognition," in *Natural Language Understanding and Intelligent Applications*, pp. 239–250, Springer, Berlin, Germany, 2016.
- [30] H. He and Xu Sun, "F-score driven max margin neural network for named entity recognition in Chinese social media," in *Proceedings of the Fifteenth Conference of the European Chapter of the Association for Computational Linguistics*, vol. 2, pp. 713–718, Association for Computational Linguistics, Valencia, Spain, April 2017.
- [31] W. Liu, X. Fu, Y. Zhang, and W. Xiao, "Lexicon enhanced Chinese sequence labeling using BERT adapter," in *Proceedings of the Fifty Ninth Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP 2021*, Edited by C. Zong, F. Xia, W. Li, and R. Navigli, Eds., vol. 1, pp. 5847–5858, Association for Computational Linguistics, August 2021.
- [32] Y. Liu, M. Ott, N. Goyal et al., "Roberta: A robustly optimized BERT pretraining approach," 2019, <https://arxiv.org/abs/1907.11692>.
- [33] X. Jiao, Y. Yin, L. Shang et al., "Tinybert: distilling BERT for natural language understanding," in *Proceedings of the Findings of the Association for Computational Linguistics: EMNLP 2020*, T. Cohn, Y. He, and Y. Liu, Eds., pp. 4163–4174, Association for Computational Linguistics, November 2020.
- [34] Y. Cui, W. Che, T. Liu, B. Qin, and Z. Yang, "Pre-training with whole word masking for Chinese BERT," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 29, pp. 3504–3514, 2021.
- [35] W. Zhang, H. Lin, X. Han, and Le Sun, "Deiasing distantly-supervised named entity recognition via causal intervention," in *Proceedings of the Fifty Ninth Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP 2021*, Edited by C. Zong, F. Xia, W. Li, and R. Navigli, Eds., vol. 1, pp. 4803–4813, Association for Computational Linguistics, August 2021.
- [36] D. Singh Sachan, P. Xie, M. Sachan, and E. P. Xing, "Effective use of bidirectional language modeling for transfer learning in biomedical named entity recognition," 2017, <https://arxiv.org/abs/1711.07908>.
- [37] S. Ruder and B. Plank, "Strong baselines for neural semi-supervised learning under domain shift," 2018, <https://arxiv.org/abs/1804.09530>.
- [38] M. Liu, J. Han, H. Zhang, and Y. Song, "Domain adaptation for disease phrase matching with adversarial networks," in *Proceedings of the BioNLP 2018 workshop 2018 workshop*, pp. 137–141, Melbourne, Australia, July 2018.
- [39] M. T. Rosenstein, Z. Marx, L. P. Kaelbling, and T. G. Dietterich, "To transfer or not to transfer," in *Proceedings of the NIPS 2005 workshop on transfer learning*, vol. 898, p. 3, Vancouver, British Columbia, Canada, December 2005.
- [40] S. Ruder and B. Plank, "Learning to Select Data for Transfer Learning with Bayesian Optimization," 2017, <https://arxiv.org/abs/1707.05246>.
- [41] J. Feng, M. Huang, L. Zhao, Y. Yang, and X. Zhu, "Reinforcement Learning for Relation Classification from Noisy Data," in *Proceedings of the AAAI, New Orleans LA USA*, February 2018.