

# Distantly Supervised NER with Partial Annotation Learning and Reinforcement Learning

Yaosheng Yang, Wenliang Chen, Zhenghua Li, Zhengqiu He, Min Zhang

Institute of Artificial Intelligence

School of Computer Science and Technology, Soochow University, China

{ysyang, zqhe}@stu.suda.edu.cn

{wlchen, zhli13, minzhang}@suda.edu.cn

## Abstract

A bottleneck problem with Chinese named entity recognition (NER) in new domains is the lack of annotated data. One solution is to utilize the method of distant supervision, which has been widely used in relation extraction, to automatically populate annotated training data without human-cost. The distant supervision assumption here is that if a string in text is included in a predefined dictionary of entities, the string might be an entity. However, this kind of auto-generated data suffers from two main problems: incomplete and noisy annotations, which affect the performance of NER models. In this paper, we propose a novel approach which can partially solve the above problems of distant supervision for NER. In our approach, to handle the incomplete problem, we apply partial annotation learning to reduce the effect of unknown labels of characters. As for noisy annotation, we design an instance selector based on reinforcement learning to distinguish positive sentences from auto-generated annotations. In experiments, we create two datasets for Chinese named entity recognition in two domains with the help of distant supervision. The experimental results show that the proposed approach obtains better performance than the comparison systems on both two datasets.

## 1 Introduction

In recent years, deep learning approaches have achieved great progress in the task of named entity recognition (NER) (Collobert et al., 2011; Chiu and Nichols, 2015). The standardized approach is that using BiLSTMs for encoding and then applying CRF for jointly label decoding (Huang et al., 2015; Lample et al., 2016). In addition, BiLSTMs and CNNs are employed to model character- or word-level representations (Ma and Hovy, 2016).

Most previous studies on NER focus on a certain set of predefined NER types, such as organization, location, person, date, and so on, where a certain amount of labeled data is provided to train the models. However, different applications require particular entity types, such as “Brand” and “Product” in E-commerce domain, and “Company” for finance industry. Considering the high cost of human annotation, it may not be feasible to annotate large amounts of labeled data for each new NER type, but small-scale data is available at some time.

As an alternative solution, distant supervision can automatically generate large-scale labeled data for new-type NER without human-cost. The idea of distant supervision has widely used in the task of relation extraction (Mintz et al., 2009; Riedel et al., 2010; Zeng et al., 2015). For relation extraction, at first we have a knowledge base. If two entities  $e_1$  and  $e_2$  have relation  $r$  according to the knowledge base, then we populate this knowledge and assume the relation between  $e_1$  and  $e_2$  is  $r$  in the sentences that contain the both entities. In this way, we can produce a lot of labeled data for model training.

Similarly, in our task, we first acquire a dictionary containing a list of the new-type entities. Then, we automatically generate large-scale labeled data by assuming that each entity mention in a sentence is a positive instance of the corresponding type according to the dictionary. Figure 1(a) shows an example

The corresponding author is Wenliang Chen.

This work is licenced under a Creative Commons Attribution 4.0 International Licence. Licence details: <http://creativecommons.org/licenses/by/4.0/>

(a) correct annotation	[ <u>衬 衫(Shirt)</u> ] 和(and) [ <u>卫 衣(hoodies)</u> ] 很(well) 合 身(fit)
(b) incomplete annotation	我(I) 想(want to) 买(buy) [ <u>皮 鞋(leather shoes)</u> ] 皮 带(leather belt)
(c) noisy annotation	我(I) 想(want to) 买(buy) [ <u>工 装</u> ] 鞋(work shoes)

Figure 1: Examples generated by distant supervision, where “PDT” means a product name tagged by the distant supervision and one string with underline should be an entity of product.

that is regarded as a positive instance with two “Product” names are correctly matched by the distant supervision method.

However, in practice we find that the automatically labeled NER data suffers from two problems, i.e., *incomplete annotation* and *noisy annotation*, which negatively affect the performance of NER systems. The incomplete annotation problem means that not every entity has been labeled in the way of distant supervision. For example, “皮鞋(leather shoes)” is included in the dictionary, while “皮带(leather belt)” is not included. Thus in Figure 1(b), “皮鞋(leather shoes)” is annotated as a PDT, but “皮带(leather belt)” is not annotated. The noisy annotation problem means that the matched entity does not correspond to entity definition, such as Figure 1(c), where “工装鞋(work shoes)” is a product, but only the first two characters “工装(fatigue clothes)” are matched by the dictionary because “工装鞋(work shoes)” is not included in the dictionary. Obviously, such false labeled examples certainly provide wrong supervision during model training if we directly use the automatically generated data.

In this paper, we propose an approach to handle the two problems of distantly supervised NER data. As for the incomplete annotation problem, we treat the data as partially annotated data based on the extended CRF-PA model that can directly learn from partial annotations (PA) (Tsuboi et al., 2008). The noisy annotation problem is also ubiquitous in distantly supervised for relation extraction, and researchers try to address this issue by using reinforcement learning (RL) technology to select positive instances (Feng et al., 2018). Inspired of their work, we design an instance selector to obtain clean instances from distantly supervised NER data.

In summary, we make the following contributions:

- We propose a novel approach for new-type named entity recognition, which firstly combines the advantages of both partial annotation learning and reinforcement learning, to handle the problems of incomplete annotation and noisy annotation brought by distant supervision.
- We create two datasets for Chinese named entity recognition with the help of distant supervision in e-commerce and news domains. The experimental results on the newly created datasets show that the proposed approach performs better than the comparison systems.

## 2 Basic Settings

### 2.1 Distantly Supervised NER Data

Here we focus mainly on the Chinese NER, which is more difficult than NER for other languages such as English for the lack of morphological variations such as capitalization and in particular the uncertainty in word segmentation. To get a good tagger for new entity types in new domains, we perform distant supervision to acquire labeled data for Chinese NER.

Initially, we have a small set of labeled seed data  $\mathcal{H}$  for new entity types, and large-scale unlabeled data pool  $\mathcal{U}$ . We collect named entities to construct dictionary  $\mathcal{D}$ , and use the entries of  $\mathcal{D}$  to match the strings of the sentences in  $\mathcal{U}$  by the method of distant supervision. Then we obtain a set of sentences containing at least one matched strings, and the set is denoted as  $\mathcal{A}$ . The purpose in this paper is that we make full use of  $\mathcal{H}$  and  $\mathcal{A}$  to build a NER system.

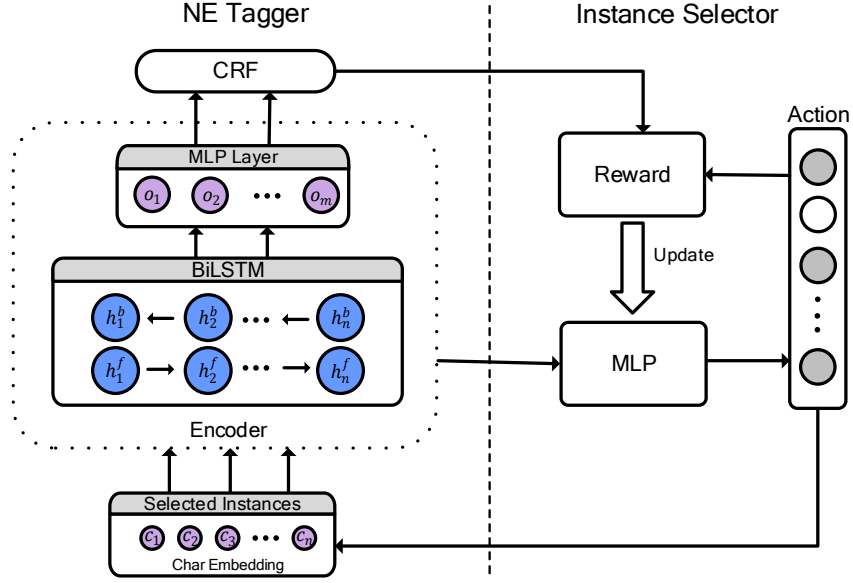


Figure 2: The framework of the proposed model, which consists of two parts. The right instance selector is a policy network, which chooses sentences from candidate dataset to expand training data to improve the left NE Tagger. The instance selector is trained based on the reward provided by NE Tagger.

In this paper, we treat Chinese NER task as a sequence labeling problem. We exploit the traditional BIO schema to represent the tags of sentences. Concretely, we tag the beginning character of an entity by “B-XX”, the other characters of this entity by “I-XX”, and the character as “O” if it is not inside an entity, where “XX” is the type of entities.

## 2.2 The Baseline LSTM-CRF

Given a sentence  $x = c_1 c_2 \cdots c_n$ , the goal is to assign a unique tag  $y_i$  for Chinese character  $c_i$  in the sentence. In general, the model predicts the entities in the sentence  $x$  by estimating the probability  $p(y|x)$ , where  $y$  is a possible label sequence for sentence  $x$ . The final output  $y_{max}$  of the system for one sentence is the label sequence with the maximum probability.

Here, we present a new NE tagger based on the LSTM-CRF model of Lample et al. (2016), which achieves the state-of-the-art performance in the NER task. Following Peng and Dredze (2015a), we represent Chinese characters as vectors and feed them into BiLSTM layer in the Chinese NER task. The left part of Figure 2 shows the framework of our baseline LSTM-CRF based NE tagger.

**The input layer.** For each input sentence  $x = c_1 c_2 \cdots c_n$ , we map serialized characters into a list of vectors  $\mathbf{x}_1 \mathbf{x}_2 \cdots \mathbf{x}_n$  with an embedding layer including a lookup table as its key parameter. Following Lample et al. (2016), the lookup table is initialized with embeddings pre-trained on a large-scale raw corpus and is further fine-tuned during our training process.

**The BiLSTM layer.** With vector sequence  $\mathbf{x}_1 \mathbf{x}_2 \cdots \mathbf{x}_n$  from input layer, we apply a bidirectional LSTM layer to encode the semantic dependency which provides high level features. The LSTM incorporates a memory-cell to solve the exploding and diminishing gradients of basic RNNs (Graves and Schmidhuber, 2005). For each character  $c_t$  in the sentence, we can obtain the output features  $\mathbf{h}_t$  by concatenating  $\vec{\mathbf{h}}_t$  and  $\overleftarrow{\mathbf{h}}_t$ , which computed in turn and in reverse to represent its left and right context. What’s more, dropout (Srivastava et al., 2014) is applied to the outputs of the BiLSTM layer to avoid overfitting.

**The MLP layer.** Then, we employ a multi-layer perceptron (MLP), also known as feed-forward neural networks, to compute the scores of all labels at each position  $t$ , denoted as  $\mathbf{o}_t$ ,

$$\begin{aligned} \mathbf{h}_t^{\text{mlp1}} &= \mathbf{W}^{\text{mlp1}} \mathbf{h}_t + \mathbf{b}^{\text{mlp1}} \\ \mathbf{o}_t &= \mathbf{W}^{\text{mlp2}} \mathbf{h}_t^{\text{mlp1}} + \mathbf{b}^{\text{mlp2}} \quad t \in [1, n] \end{aligned} \quad (1)$$

where  $\mathbf{W}^{\text{mlp1/mlp2}}$  and  $\mathbf{b}^{\text{mlp1/mlp2}}$  are the parameters.

**The CRF layer.** Finally, we adopt a CRF layer to derive the conditional probability of a label sequence  $y$  as follows:

$$p(y|x) = \frac{e^{\text{score}(x,y)}}{\sum_{\tilde{y} \in \mathbf{Y}_x} e^{\text{score}(x,\tilde{y})}} \quad (2)$$

$$\text{score}(x, y) = \sum_{t=1}^n (o_{t,y_t} + T_{y_{t-1},y_t})$$

where  $T_{y_{t-1},y_t}$  represents the transmission score from  $y_{t-1}$  to  $y_t$ , and  $\mathbf{Y}_x$  are all candidate label sequences of input sentence  $x$ .

During evaluation, we adopt the minimum Bayes risk (MBR) decoding to find the optimal label sequence as follows:

$$y^* = \arg \max_y \left( \sum_{t=1}^n p(y_t|x, t) \right) \quad (3)$$

$$p(y_t|x, t) = \sum_{\tilde{y} \in \mathbf{Y}_x, \tilde{y}_t = y_t} p(\tilde{y}|x)$$

where  $p(y_t|x, t)$  is the marginal probability of tagging the  $t$ -th position in  $x$  as  $y_t$ .

**Training objective.** In order to maximize the probability of the correct label sequence, we exploit a negative log-likelihood objective as the loss function:

$$\text{loss}(\Theta, x, y) = -\log p(y|x) \quad (4)$$

where  $\Theta$  is the set of baseline model parameters.

### 3 Our Approach

This section describes our approach for new-type NER via distant supervision. To handle the problem of incomplete and noisy annotations, we propose a novel model for NER task. As shown in Figure 2, the framework of our model consists of two modules: the NE Tagger built on the idea of partial annotation learning to reduce the effect of unknown-type characters, the instance selector which chooses positive sentences from a candidate set and provides them to the NE Tagger.

#### 3.1 LSTM-CRF-PA for Incomplete Annotation

It is inappropriate to regard those characters as non-entities, although they can not be matched according to the dictionary. It is a common problem known as false negative instance which may misguide model if we arbitrarily label them as ‘‘O’’. Therefore, we consider that each non-matched character can be annotated as any proper label. For example in Figure 3, except that ‘‘皮鞋 (leather shoes)’’ have definite labels, all of the remaining characters can be labeled as ‘‘B-PDT’’, ‘‘I-PDT’’ and so on. In other words, we denote a set of label sequences  $\mathbf{z}$  for every distantly supervised sentence, whose probability is naturally the sum of probability of each possible label sequence  $\tilde{y}$  in  $\mathbf{z}$ . We expand the original model for this situation and apply softmax over all candidate output label sequences, thus the probability of a distantly supervised instance is computed as follows:

$$p(\mathbf{z}|x) = \sum_{\tilde{y} \in \mathbf{z}} p(\tilde{y}|x) = \frac{\sum_{\tilde{y} \in \mathbf{z}} e^{\text{score}(x,\tilde{y})}}{\sum_{\tilde{y} \in \mathbf{Y}_x} e^{\text{score}(x,\tilde{y})}} \quad (5)$$

We exploit a negative log-likelihood objective as the loss function. Therefore, the loss function of our model with CRF-PA can be computed as follows:

$$\text{loss}(\Theta, x, \mathbf{z}) = -\log p(\mathbf{z}|x) \quad (6)$$

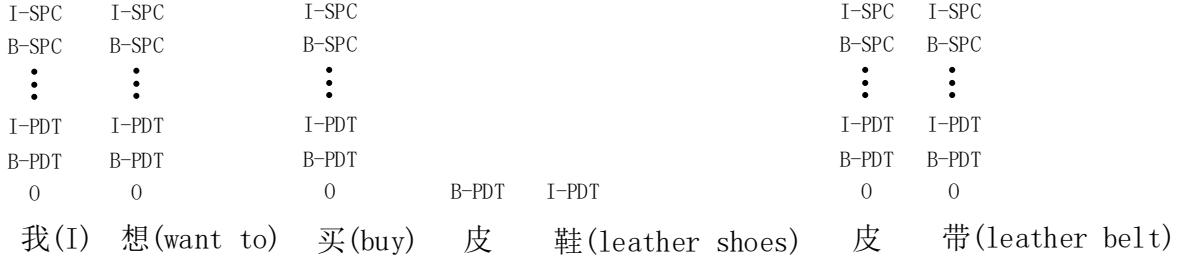


Figure 3: An example instance which is partially annotated. “PDT” means product name and “SPC” means specification name.

where  $\Theta$  is the set of all NE tagger parameters.

In particular, if the sentence is annotated by hand and each character has definite label, the set  $\mathbf{z}$  includes only one label sequence. Thus the above objective function is also suitable for supervised instances. We use standard back-propagation method to minimize the loss function of the NE tagger.

### 3.2 Instance Selector for Noisy Annotation

Our goal is to train an agent as an instance selector with reinforcement learning (RL) technology. Following Feng et al. (2018), the agent interacts with the environment and makes a decision at the sentence-level. We merge the initial hand-tagged seed set  $\mathcal{H}$  and the distantly supervised set  $\mathcal{A}$  into a candidate dataset  $\mathcal{C}$ . At each episode, we collect a random-size bag of instances  $\mathcal{B}$  from  $\mathcal{C}$ . All the supervised instances in the current bag are default to be selected without decisions of agent. For each distantly supervised instance in the current bag, the agent makes an action from set of  $\{1, 0\}$  to decide whether to select this instance or not. The agent receives the reward when all the actions have been completed. The reward represents the feedback of actions on this bag and will be used to update agent. The goal of agent is to decide actions that enable to maximize the reward.

**State representation.** In our view, the state  $s_t$  represents the current instance along with its label sequences. We represent the state as a vector  $\mathbf{S}_t$ , which consists of the following information: (1) The serialized vector representations of current instance, which are observed from the BiLSTM layer of baseline model. (2) The label scores calculated with output of the MLP layer from the shared encoder (denoted as  $\mathbf{o}_t$  in Formula 1) and annotation of this instance, which means the tag-conditional noise of the distantly supervised annotation. More specifically, if a character is a part of an entity and annotated as a definite label (such as “皮” and “鞋” in Figure 3), the score of this position is the corresponding value in  $\mathbf{o}_t$ . Otherwise, we compute it by averaging the scores of all labels in  $\mathbf{o}_t$ . In this way, the dimension of the label scores vector is equal to the uniform length of sentences and will be concatenated with the first part.

**Policy network.** The agent decides an action  $a_t \in \{0, 1\}$  to indicate whether the selector will select the  $t$ -th distantly supervised instance. The action value is sampled by the selector as  $A_\Theta(s_t, a_t)$  where  $\Theta$  is a multi-layer perceptron (MLP) with the parameter  $\{\mathbf{W}, \mathbf{b}\}$ . We adopt a logistic function as the policy function:

$$A_\Theta(s_t, a_t) = a_t \sigma(\mathbf{W} * \mathbf{S}_t + \mathbf{b}) + (1 - a_t)(1 - \sigma(\mathbf{W} * \mathbf{S}_t + \mathbf{b})) \quad (7)$$

where  $\mathbf{S}_t$  is the state vector, and  $\sigma(\cdot)$  is the sigmoid function.

**Reward.** The reward is used to evaluate the ability of current NE tagger to predict labels of each character. The model receives a delayed average reward when it finishes all the selections in current bag, and before that the reward of each action is zero. The current bag  $\mathcal{B}$  consists of two subsets: hand-tagged sentences  $\mathcal{H}$  and distantly supervised instances  $\hat{\mathcal{A}}$ . Now that the NE tagger calculates conditional probability for every sentence of bag  $\mathcal{B}$ , the reward can be computed on the set of selected distantly

supervised instances  $\tilde{\mathcal{A}}_s$  and all the hand-tagged sentences:

$$r = \frac{1}{|\tilde{\mathcal{A}}_s| + |\tilde{\mathcal{H}}|} \left( \sum_{x_j, \mathbf{z} \in \tilde{\mathcal{A}}_s} \log p(\mathbf{z}|x_j) + \sum_{x_k, y \in \tilde{\mathcal{H}}} \log p(y|x_k) \right) \quad (8)$$

Different from the work of Feng et al. (2018), we have a set of supervised data. Our selector can be trained along the guidance of these prior knowledge about which sentences are labeled correctly. Therefore, the reward will become dependable and oriented, and it can guide the selector to maximize likelihood of all the instances in training dataset.

**Selector Training.** We use policy gradient method (Sutton et al., 2000) to optimize the policy network to maximize the reward of selections. For each random-sized bag  $\mathcal{B}$ , the feedback of every action  $r(a_t)$  is same as average reward  $r$ . We compute the gradient and update the selector as follows:

$$\Theta = \Theta + \alpha \sum_{t=1}^{|\tilde{\mathcal{A}}|} r(a_t) \nabla_{\Theta} \log A_{\Theta}(s_t, a_t) \quad (9)$$

### 3.3 Joint Training

The parameters of the NE tagger and instance selector are learned iteratively. In each round, the selector first selects  $\mathcal{A}_s$  from  $\mathcal{A}$  and merges them with supervised sentences for the tagger. Meanwhile, the parameters of the NE tagger are learned from the newly training data and the tagger provides feedback reward to the selector to optimize its policy function.

## 4 Experiment

### 4.1 Datasets

We use two datasets in the experiments: one is from e-commerce domain and another is from news domain.

**EC:** In e-commerce domain (EC), we have five types of entities: Brand, Product, Model, Material, and Specification on user queries. This data contains 2,400 sentences tagged by annotators. We split the data into three sets: 1,200 sentences for training, 400 for dev, and 800 for testing. We collect a list of entities to construct dictionary from the training data. To reduce the effect of ambiguities, we remove the entry that belongs to more than one type, or it is a number or single character. Finally, the dictionary has 927 entries (included as EC.dic in supplementary materials). We perform distant supervision on raw data to obtain 2,500 sentences.

**NEWS:** For news domain, we use a NER data from MSRA, which was used in Sighan-bakeoff (Levow, 2006). We only test our systems on the type of PERSON. We randomly select 3,000 sentences as training dataset, 3,328 as dev data, and 3,186 as testing data. The rest set is used as raw data, having 36,602 sentences. We collect a list of person names from the training data. To increase the coverage, we add an additional names to the list. Finally, the list has 71,664 entries (included as NEWS.dic in supplementary materials). We perform distant supervision on raw data to obtain 3,722 sentences.

**Embedding:** In our approach, we need to map Chinese characters into vector representations by the lookup table, which can be initialized either by random or pre-trained. Many previous works (Lample et al., 2016; Peng and Dredze, 2015b) have shown that pre-trained embeddings on large-scale unlabeled corpus enable to initialize the table and observe efficiently improvements. Therefore, we collect one million sentences from the user-generated text in Internet, and pre-trained embeddings with the tool `word2vec`<sup>1</sup>. We set the embedding dimension as 100, the minimum frequency of occurrence as 5, and the window size of 5.

<sup>1</sup><https://code.google.com/archive/p/word2vec>

Model	Training Data	Dev			Test		
		P	R	F1	P	R	F1
Dict-based	/	<b>74.21</b>	29.68	42.41	<b>75.60</b>	31.05	44.02
LSTM-CRF	$\mathcal{H}$	63.78	61.26	62.49	59.93	58.46	59.19
LSTM-CRF	$\mathcal{H} + \mathcal{A}$	67.75	52.91	59.42	62.36	48.54	54.59
LSTM-CRF+SL	$\mathcal{H} + \mathcal{A}$	67.56	55.04	60.66	62.86	50.87	56.23
LSTM-CRF-PA	$\mathcal{H} + \mathcal{A}$	60.34	<b>64.49</b>	62.35	59.36	60.82	60.08
LSTM-CRF-PA+SL	$\mathcal{H} + \mathcal{A}$	62.31	63.79	<b>63.04</b>	61.57	<b>61.33</b>	<b>61.45</b>

Table 1: Main results on the EC dataset.

Model	Training Data	Dev			Test		
		P	R	F1	P	R	F1
Dict-based	/	<b>95.40</b>	35.87	52.14	<b>96.08</b>	31.77	47.75
LSTM-CRF	$\mathcal{H}$	85.21	78.91	81.94	78.50	74.50	76.45
LSTM-CRF	$\mathcal{H} + \mathcal{A}$	87.00	65.20	74.54	83.41	58.96	69.09
LSTM-CRF+SL	$\mathcal{H} + \mathcal{A}$	86.33	72.34	78.72	81.99	66.10	73.19
LSTM-CRF-PA	$\mathcal{H} + \mathcal{A}$	83.78	<b>81.79</b>	82.77	79.19	<b>77.59</b>	78.38
LSTM-CRF-PA+SL	$\mathcal{H} + \mathcal{A}$	86.94	80.12	<b>83.40</b>	81.63	76.95	<b>79.22</b>

Table 2: Main results on the NEWS dataset.

## 4.2 Settings

For evaluation, we use the entity-level metrics of Precision (P), Recall (R), and their F1 values in our experiments, treating one tagged entity as correct only when it matches the gold entity exactly.

There are several hyper-parameters in our models. We set them empirically by the development performances. The instance selector is a multi-layer perceptron with 100 units in each hidden layer. We use Adam (Kingma and Ba, 2014) to train the instance selector with the learning rate 0.001. For the parameters of the tagger, we set the character embedding dimension as 100, the dimension sizes of hidden features as 200. We exploit online training with a mini-batch size 128 to learn model parameters. The max-epoch iteration is set by 800, and the best-epoch model is chosen according to the development performances. We use RMSprop (Tieleman and Hinton, 2012) with a learning rate 0.001 to update model parameters. We adopt the dropout technique to avoid overfitting by a drop value of 0.2 at the training stage.

## 4.3 Baselines

We build four systems based on our proposed approach and a dictionary-based baseline system, listed as follows:

- Dict-based: The collected entity dictionary is directly used to match the strings in the testing data.
- LSTM-CRF: the baseline model described in Section 3.2.
- LSTM-CRF-PA: the system trained on  $\mathcal{H}$  and  $\mathcal{A}$  with CRF-PA learning, but without instance selector.
- LSTM-CRF+SL: the system trained on  $\mathcal{H}$  and  $\mathcal{A}$  with instance selector, but without CRF-PA learning.
- LSTM-CRF-PA+SL: our final system trained on  $\mathcal{H}$  and  $\mathcal{A}$  with CRF-PA learning and instance selector.

Model	Training Data	Dev			Test		
		P	R	F1	P	R	F1
LSTM-CRF	25% $\mathcal{H}$	43.35	46.18	44.72	40.36	40.78	40.57
LSTM-CRF-PA+SL	25% $\mathcal{H}$ + 25% $\mathcal{A}$	45.63	50.95	48.14	44.08	46.57	45.29
LSTM-CRF	50% $\mathcal{H}$	54.73	52.77	53.73	49.88	47.64	48.73
LSTM-CRF-PA+SL	50% $\mathcal{H}$ + 50% $\mathcal{A}$	53.25	56.84	54.99	50.48	51.96	51.21
LSTM-CRF	100% $\mathcal{H}$	63.78	61.26	62.49	59.93	58.46	59.19
LSTM-CRF-PA+SL	100% $\mathcal{H}$ + 100% $\mathcal{A}$	62.31	63.79	63.04	61.57	61.33	61.45

Table 3: Results for varying percent of training data on the EC dataset.

#### 4.4 Results

In this section, we show the model performances of our proposed systems and the other systems mentioned above. Table 1 shows the experimental results on the EC data and Table 2 shows the results on the NEWS data, respectively.

The low recall scores of Dict-based systems show that the coverage of the dictionaries is low even we have more than 70K person-names for the NEWS data. Compared with LSTM-CRF trained on  $\mathcal{H}$ , LSTM-CRF system trained on  $\mathcal{H}$  and  $\mathcal{A}$  provides much lower performance on two datasets. These facts indicate that the data generated by distant supervision contains many noises that affect the performance of the models. LSTM-CRF-PA yields better performance than LSTM-CRF trained on  $\mathcal{H}$ , showing +0.89 F1 improvement on EC and +1.93 F1 on NEWS. This indicates that CRF-PA learning can reduce the effect of incomplete annotation.

From the tables, we find that compared with LSTM-CRF-PA, LSTM-CRF-PA+SL obtains absolute improvements of +1.37 and +0.84 F1 points on EC and NEWS, respectively. Overall, our final system (LSTM-CRF-PA+SL) achieves better improvement with +2.26 and +2.77 F1 on EC and NEWS respectively over our baseline system LSTM-CRF. These facts show that the RL-based instance selector can provide additional help to CRF-PA learning.

We further investigate the effect of different sizes of human-annotated data. We randomly select 25% and 50% sentences from human-annotated data  $\mathcal{H}$  as training data and build new dictionaries of entities based on them respectively. The new dictionaries are used to generate distantly supervised annotated data. Table 3 shows the results on the EC dataset, where the first two lines are for 25%, the third and fourth lines are for 50%, and the last two are for 100%. From the table, LSTM-CRF-PA+SL performs better performance than the baseline system, showing +4.72 F1 improvement on 25% and +2.48 improvement on 50%, respectively. This indicates that with smaller human-annotated data, our proposed approach can provide relatively larger improvement.

## 5 Related Work

Our approach is to utilize partial annotation learning and reinforcement learning to perform new-type named entity recognition in new domains. Several previous studies relevant to our approach have been conducted.

**NER.** Most early studies treated NER task as the sequence labeling problem based on a large annotated corpus with supervised methods, such as HMM, MEMM (Hai and Ng, 2002) and CRF (Lafferty et al., 2001). Recently, neural networks have been explored by researchers (Collobert et al., 2011; Lample et al., 2016), and applied to reduce the weakness of feature sparsity problem and heavy feature engineering (Cai and Zhao, 2016). Those models have the similar architecture for decoding and feature extraction, which is chosen as our baseline model. In order to overcome the challenge of data deficient, some approaches based on weakly supervised learning (Nadeau et al., 2006; Riloff and Jones, 1999) have been proposed and successfully expand training data and feature space. However, it is difficult to implement these methods on Chinese tasks because of the lack of morphological variations such as capitalization and in particular the uncertainty in word segmentation, and it may cause large number of matching errors.



Under reasonable assumptions, OOV features should not be forced into certain tag. What's more, joint models have also obtained great performance (Qian and Liu, 2013; Finkel and Manning, 2009).

**Learning from PAs.** Learning from PAs has always been an attractive idea, since it usually requires much less or even none human annotation effort to obtain partially annotated data than fully annotated data, especially for complex tasks like sequence labeling. Li et al. (2012) propose to only manually annotate the most uncertain word boundaries in a sentence for Chinese word segmentation in order to reduce annotation cost. Tsuboi et al. (2008) extend the standard CRF to directly learn from incomplete annotations for sequence labeling tasks. This work refers to their model as *CRF-PA*. Jiang et al. (2013) propose to derive segmentation boundaries from implicit information encoded in web texts, such as anchor texts and punctuation marks, and use them as partially labeled training data in Chinese word segmentation. Liu et al. (2014) and Yang and Vozila (2014) further improve the work of Jiang et al. (2013) by employing the more sophisticated CRF-PA model. Marcheggiani and Artières (2014) systematically compare a dozen uncertainty metrics in token-wise active learning with CRF-PA for several sequence labeling tasks. Li et al. (2016b) propose a coupled sequence labeling approach for exploiting heterogeneous data by treating the single-sided annotations as PAs for the task of joint word segmentation and POS tagging. In this work, we for the first time apply the CRF-PA model to NER, and employ distance supervision to produce partially annotated NE data.

**Reinforcement Learning.** In recent years, reinforcement learning has become an issue in research, and applied successfully to many tasks. In text generation community, a deep Q-learning is served by Guo (2015) as generative model to improve the seq2seq model, which completes the process of decoding by Iteration. Li et al. (2016a) show how to apply deep reinforcement learning to model future reward in chatbot dialogue and capture the impact of this conversation in the future. Dethlefs (2010) aim to optimize the integration of NLG tasks that are inherently different in nature by learning a generation policy with reinforcement learning. For computer vision, Yeung et al. (2017) propose a reinforcement learning-based formulation select the right examples for training a classifier from noisy web search results. To more fine-grainedly select high-quality training sentences from noisy data, Feng et al. (2018) train an instance selector based on a policy function with reinforcement learning, which is inspirational to our model.

## 6 Conclusion

This paper presents a new approach to utilize the data generated by distant supervision to perform new-type named entity recognition in new domains. We adopt partial annotation learning to address the problem of incomplete annotation and design the instance selector to choose positive sentences to reduce the effect of noisy annotation. The instance selector is based on reinforcement learning and obtains the feedback reward from the NE tagger. When tested on two newly created datasets, our systems provide better performance than the comparison systems. The data and code is available at <https://github.com/rainarch/DSNER>.

## 7 Acknowledgments

This work is supported by the National Natural Science Foundation of China (Grant No. 61572338 and 61525205). Wenliang is also partially supported by the Natural Science Foundation of the Jiangsu Higher Education Institutions(Grant No. 16KJA520001). We would also thank the anonymous reviewers for their detailed comments, which have helped us to improve the quality of this work.

## References

- Deng Cai and Hai Zhao. 2016. Neural word segmentation learning for chinese. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 409–420.
- Jason PC Chiu and Eric Nichols. 2015. Named entity recognition with bidirectional lstm-cnns. *arXiv preprint arXiv:1511.08308*.

- Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. 2011. Natural language processing (almost) from scratch. *The Journal of Machine Learning Research*, 12:2493–2537.
- Nina Dethlefs. 2010. Hierarchical reinforcement learning for adaptive text generation. In *International Natural Language Generation Conference*, pages 37–45.
- Jun Feng, Minlie Huang, Li Zhao, Yang Yang, and Xiaoyan Zhu. 2018. Reinforcement learning for relation classification from noisy data. pages 5779–5786.
- Jenny Rose Finkel and Christopher D Manning. 2009. Joint parsing and named entity recognition. In *NAACL*, pages 326–334.
- Alex Graves and Jürgen Schmidhuber. 2005. Framewise phoneme classification with bidirectional lstm and other neural network architectures. *Neural Networks*, 18(5):602–610.
- Hongyu Guo. 2015. Generating text with deep reinforcement learning. *Computer Science*, 40(4):1–5.
- Leong Chieu Hai and Hwee Tou Ng. 2002. Named entity recognition: a maximum entropy approach using global information. In *International Conference on Computational Linguistics*, pages 1–7.
- Zhiheng Huang, Wei Xu, and Kai Yu. 2015. Bidirectional lstm-crf models for sequence tagging. *CoRR*, abs/1508.01991.
- Wenbin Jiang, Meng Sun, Yajuan Lü, Yating Yang, and Qun Liu. 2013. Discriminative learning with natural annotations: Word segmentation as a case study. In *Proceedings of ACL*, pages 761–769.
- Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *Computer Science*.
- John Lafferty, Andrew McCallum, Fernando Pereira, et al. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *ICML*, volume 1, pages 282–289.
- Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. 2016. Neural architectures for named entity recognition. In *NAACL*, pages 260–270, June.
- Gina-Anne Levow. 2006. The third international chinese language processing bakeoff: Word segmentation and named entity recognition. In *Proceedings of the Fifth SIGHAN Workshop on Chinese Language Processing*, pages 108–117, Sydney, Australia, July. Association for Computational Linguistics.
- Shoushan Li, Guodong Zhou, and Chu-Ren Huang. 2012. Active learning for Chinese word segmentation. In *Proceedings of COLING 2012: Posters*, pages 683–692.
- Jiwei Li, Will Monroe, Alan Ritter, Dan Jurafsky, Michel Galley, and Jianfeng Gao. 2016a. Deep reinforcement learning for dialogue generation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1192–1202, Austin, Texas, November. Association for Computational Linguistics.
- Zhenghua Li, Jiayuan Chao, Min Zhang, and Jiwen Yang. 2016b. Fast coupled sequence labeling on heterogeneous annotations via context-aware pruning. In *Proceedings of EMNLP*, pages 753–762.
- Yijia Liu, Yue Zhang, Wanxiang Che, Ting Liu, and Fan Wu. 2014. Domain adaptation for CRF-based Chinese word segmentation using free annotations. In *Proceedings of EMNLP*, pages 864–874.
- Xuezhe Ma and Eduard Hovy. 2016. End-to-end sequence labeling via bi-directional lstm-cnns-crf. In *Meeting of the Association for Computational Linguistics*, pages 1064–1074.
- Diego Marcheggiani and Thierry Artières. 2014. An experimental comparison of active learning strategies for partially labeled sequences. In *Proceedings of EMNLP*, pages 898–906.
- Mintz, Mike, Steven, Jurafsky, and Dan. 2009. Distant supervision for relation extraction without labeled data. In *Joint Conference of the Meeting of the ACL and the International Joint Conference on Natural Language Processing of the Afnlp: Volume*, pages 1003–1011.
- David Nadeau, Peter D. Turney, and Stan Matwin. 2006. Unsupervised named-entity recognition: Generating gazetteers and resolving ambiguity. In *Conference of the Canadian Society for Computational Studies of Intelligence*, pages 266–277.
- Nanyun Peng and Mark Dredze. 2015a. Named entity recognition for chinese social media with jointly trained embeddings. In *Proceedings of the EMNLP*, pages 548–554, Lisbon, Portugal.

- Nanyun Peng and Mark Dredze. 2015b. Named entity recognition for chinese social media with jointly trained embeddings. In *Conference on Empirical Methods in Natural Language Processing*, pages 548–554.
- Xian Qian and Yang Liu. 2013. Joint chinese word segmentation, pos tagging and parsing. In *Proceedings of EMNLP*, pages 501–511.
- Sebastian Riedel, Limin Yao, and Andrew McCallum. 2010. Modeling relations and their mentions without labeled text. In *European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 148–163.
- Ellen Riloff and Rosie Jones. 1999. Learning dictionaries for information extraction by multi-level bootstrapping. In *Sixteenth National Conference on Artificial Intelligence and the Eleventh Innovative Applications of Artificial Intelligence Conference Innovative Applications of Artificial Intelligence*, pages 474–479.
- Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: A simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, 15(1):1929–1958.
- Richard S Sutton, David A McAllester, Satinder P Singh, and Yishay Mansour. 2000. Policy gradient methods for reinforcement learning with function approximation. In *Advances in neural information processing systems*, pages 1057–1063.
- Tijmen Tieleman and Geoffrey Hinton. 2012. Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude. *COURSERA: Neural networks for machine learning*, 4(2):26–31.
- Yuta Tsuboi, Hisashi Kashima, Hiroki Oda, Shinsuke Mori, and Yuji Matsumoto. 2008. Training conditional random fields using incomplete annotations. In *Proceedings of COLING*, pages 897–904.
- Fan Yang and Paul Vozila. 2014. Semi-supervised Chinese word segmentation using partial-label learning with conditional random fields. In *Proceedings of EMNLP*, pages 90–98.
- Serena Yeung, Vignesh Ramanathan, Olga Russakovsky, Liyue Shen, Greg Mori, and Li Fei-Fei. 2017. Learning to learn from noisy web videos. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5154–5162, July.
- Daojian Zeng, Kang Liu, Yubo Chen, and Jun Zhao. 2015. Distant supervision for relation extraction via piecewise convolutional neural networks. In *Conference on Empirical Methods in Natural Language Processing*, pages 1753–1762.