Neural Regularized Domain Adaptation for Chinese Word Segmentation

Zuyi Bao and **Si Li** and **Weiran Xu** and **Sheng Gao**
Beijing University of Posts and Telecommunications, Beijing
baozuyi,lisi,xuweiran,gaosheng@bupt.edu.cn

Abstract

For Chinese word segmentation, the large-scale annotated corpora mainly focus on newswire and only a handful of annotated data is available in other domains such as patents and literature. Considering the limited amount of annotated *target* domain data, it is a challenge for segmenters to learn domain-specific information while avoid getting over-fitted at the same time. In this paper, we propose a neural regularized domain adaptation method for Chinese word segmentation. The teacher networks trained in *source* domain are employed to regularize the training process of the student network by preserving the general knowledge. In the experiments, our neural regularized domain adaptation method achieves a better performance comparing to previous methods.

1 Introduction

As the Chinese text comes without word delimiters, the Chinese word segmentation becomes a necessary step towards further syntactic analysis. With the evolving of statistical word segmentation techniques (Peng et al., 2004; Kiat Low et al., 2005; Zhang and Clark, 2008), some of the state-of-the-art systems (Sun, 2011; Hatori et al., 2012) reported high accuracy in large-scale annotated dataset (Xue et al., 2005; Emerson, 2005). However, as large-scale annotated corpora mainly focus on domains like newswire, it often brings a significant decrease in performance when we directly apply models trained on these corpora to other domains (Liu and Zhang, 2012; Li and Xue, 2014; Qiu and Zhang, 2015). Such a problem is mainly due to the differences in distributions between the training (source domain) and testing (target domain) data, and well-known as *domain adaptation*. In this paper, we focus on the fully-supervised domain adaptation (Daume, 2007) where large-scale annotated corpora of source domain and only a handful of annotated data of target domain are available. As the annotated data in target domain is often insufficient to train a effective model, the key problem is how to fully explore the information contained in the target domain data and avoid getting over-fitted at the same time.

Regularization is often employed in previous domain adaptation methods to escape the trap of over-fitting. Blitzer et al. (2007); Rozantsev et al. (2016) introduced loss functions that prevent corresponding weights from deviating significantly from the source model parameters. Kullback-Leibler divergence was added to force the feature distribution from adapted model to be close to that from the unadapted model (Yu et al., 2013). Ganin et al. (2016) adopted adversarial training to ensure that the feature distributions over the different domains are close to each other. In this paper, we employ a neural regularized domain adaption method based on *Knowledge Distillation* (Bucilu et al., 2006; Hinton et al., 2015) for Chinese word segmentation.

Knowledge distillation is first designed and proposed to do model compression (Bucilu et al., 2006; Hinton et al., 2015), where a teacher model and a student model is involved. The teacher model is a complex model and trained on large-scale annotated data. The student model is a small model and trained by mimicking the output of the teacher model. Because knowledge distillation is able to transfer knowledge between models, this method is extended and applied to other tasks. Li and Hoiem (2016) adopted this method to gradually add new capabilities to a multi-task system. Hu et al. (2016) transferred the knowledge of first-

Proceedings of the 9th SIGHAN Workshop on Chinese Language Processing, pages 11–20,
Taipei, Taiwan, December 1, 2017.

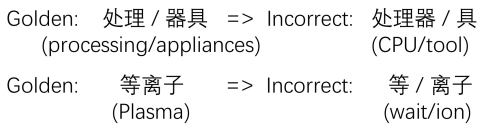

Figure 1: The model trained on newswire data makes mistakes on patent data.

order logic rules to enhance neural networks.

Domain adaptation is also explored by using knowledge distillation. Ao et al. (2017) utilized the unlabeled data to transfer the knowledge from the source models. Support Vector Machine is used as base classifier to efficiently solve the imitation parameter. Ruder et al. (2017) employed a measure for obtaining the trustworthiness of a teacher model. However, previous work mainly focus on semi-supervised domain adaptation of sentiment analysis, while we explore the fully-supervised domain adaptation of Chinese word segmentation.

In the domain adaptation for Chinese word segmentation, two kinds of domain adaptation tasks have been explored. One is annotation standard adaptation (Jiang et al., 2009; Chen et al., 2017), which explores the common underlying knowledge between the corpora with different annotation standards. The other is document type adaptation (Liu and Zhang, 2012; Liu et al., 2014; Zhang et al., 2014; Qiu and Zhang, 2015; Li and Xue, 2016), such as using newswire document to label novel (Liu et al., 2014).

In this paper, we focus on the document type adaptation which is a challenging problem in many real-world applications. As shown in Fig. 1, the model trained on publicly available newswire data outputs incorrect segmentation for patents.

In the previous work of this task, lexicons were proved effective for improving cross-domain performance (Zhang et al., 2014; Liu et al., 2014). Cross-domain features were explored to capture the characteristics of distributions utilizing unlabelled data in both source and target domain (Liu and Zhang, 2012; Li and Xue, 2016). However, previous methods mainly focus on feature-based methods utilizing unlabelled data or external resources such as lexicons. How to utilize a handful of annotated target domain data is still under exploration.

In this paper, we propose a neural regularized domain adaption method for Chinese word segmentation. A neural segmenter trained with source domain data is employed as the teacher model. A student model is then trained with target domain data under the regularization from the teacher model. The regularization retains the general information from source domain and prevents the student model from over-fitting during the target domain-specific training. Our contributions are as follows:

(1) we propose a neural method for fully-supervised domain adaptation of Chinese word segmentation and show its effectiveness in the experiments.

(2) we perform our neural domain adaptation method with different hyper-parameters and show it works as an neural regularization.

(3) we analyse the results showing that our method explores the domain-specific information and preserves the general knowledge at the same time.

(4) we propose a split of CTB9 data and perform domain adaptation experiments on the CTB9.

2 Method

2.1 Fully-supervised Domain Adaptaion

In the fully-supervised domain adaptation of Chinese word segmentation, one or multiple source domains $\{D_{s_1}, \cdots, D_{s_i}\}$ are provided with one target domain D_t. In each source domain, a trained model T_i or a large-scale of annotated sentences $\{(x_1, y_1), \cdots, (x_{n_i}, y_{n_i})\}$ are available. While only a handful of annotated target domain sentences $\{(x_1, y_1), \cdots, (x_{n_t}, y_{n_t})\}$ are provided, where we have $n_i >> n_t$. In the domain adaptation, we aim at training a model that works well on the target domain. As the amount of target domain annotated data is limited, we are forced to explore both the general information of the source domain and the domain-specific information of the target domain.

2.2 Baseline Segmenter

In this paper, we take the convolutional neural segmenter as our baseline model because that (1) same as previous baseline models, convolutional neural segmenters take Chinese word segmentation as sequence labelling task (Xue, 2003); and (2) previous baseline segmenters (Liu and Zhang, 2012; Zhang et al., 2014; Li and Xue, 2016) are limited with local features. Therefore, it may be unfair to take recurrent networks with long-range

dependence as rival; (3) the performance of convolutional neural segmenter is comparable with previous baseline segmenters.

The architecture of our baseline model is simplified from (Chen et al., 2016), we remove the highway, recurrent and k-max pooling layer. And it is equivalent to a feed-forward neural network (Collobert et al., 2011) with multiple window sizes. We take the convolutional neural segmenter as an example, but our method is not limited by the architecture of neural segmenters.

The basic unit of convolutional neural networks (CNN) is filters (Kim, 2014), a filter of window size w is represented as $\mathbf{m} \in R^{w \times d}$ where d is the size of embeddings. Let $\mathbf{x}$ refers to the concatenate of w character embeddings. Then features $\mathbf{c_i}$ from a filter i is generated by:

$$\mathbf{c_i} = f(\mathbf{m} \otimes \mathbf{x} + \mathbf{b}), \tag{1}$$

where $\otimes$ is convolution operator, $\mathbf{m}$ and $\mathbf{b}$ are the weight matrix of filter and bias, f is the non-linear function such as ReLU in our network. And for each window size, multiple filters are applied to generate multiple feature maps which are concatenated together. Then a softmax layer is appended for predicting the label of each character. Our neural word segmenter regards Chinese word segmentation as a sequence labelling task. The segmenter adopts BIES (Begin, Inside, End, Single) four labels scheme which represents the position of character inside a word. During the training phase, the cross-entropy cost function is used. And during the testing phase, the label sequences are constructed through beam search.

2.3 Neural Regularization

As shown in Fig. 2, the architecture of our neural regularization strategy consists of a teacher network and a student network. Both of them can be arbitrary neural network structures, and we take our baseline segmenter as an example. The teacher network can be obtained in two ways: (1) a provided source domain segmenter; or (2) a segmenter trained by provided annotated source domain data. And we aim at utilizing the teacher network $softmax(f_T(x))$ with a handful of target domain data $(x_1, y_1), \cdots, (x_{n_t}, y_{n_t})$ to train a student network $softmax(f_S(x))$ that works well in target domain.

The process of training is as following: (1) a sentence is feeded into the teacher network and the soft label distribution of each character s^T is predicted by the teacher network as:

$$s^T_{ij} = softmax(f_T(x_{ij})/T), \tag{2}$$

where x_{ij} is the j-th character of i-th sentence, T is a hyper-parameter named temperature to control the smoothness of the soft label distribution and smooth the regularization. (2) similar with step 1, the sentence is also feeded into the student network. The label distribution p^S and a smoothed version s^S are predicted for each character by the student network as:

$$p^S_{ij} = softmax(f_S(x_{ij})), \tag{3}$$

$$s^S_{ij} = softmax(f_S(x_{ij}/T)), \tag{4}$$

(3) train the student network with the annotated target domain data using the loss function as:

$$\ell_{seg} = \frac{1}{n}\sum_{i,j} -y_{ij} \log p^S_{ij}, \tag{5}$$

$$\ell_{re} = \frac{1}{n}\sum_{i,j} -s^T_{ij} \log s^S_{ij}, \tag{6}$$

$$\arg\min_{\theta} \ell = \alpha\ell_{seg} + (1-\alpha)\ell_{re}, \tag{7}$$

where ℓ_{seg} is the supervised loss, ℓ_{re} is the regularization loss from the teacher network, θ is the parameters in the student network, α is a hyper-parameter balancing the supervised loss and regularization. Our neural regularization for Chinese word segmentation can be easily applied to multiple source domain scenario as:

$$\ell_{seg} = \frac{1}{n}\sum_{i,j} -y_{ij} \log p^S_{ij}, \tag{8}$$

$$\ell_{re_m} = \frac{1}{n}\sum_{i,j} -s^{T_m}_{ij} \log s^S_{ij}, \tag{9}$$

$$\arg\min_{\theta} \ell = \alpha_1\ell_{seg} + \sum_m \alpha_m \ell_{re_m}, \tag{10}$$

$$s.t. \quad \alpha_1 + \sum_m \alpha_m = 1, \tag{11}$$

where ℓ_{re_m} is the regularization loss from the m-th teacher network. The amount of target domain data is insufficient to train a model that generalizes well directly. In our neural regularized method, the neural regularization loss from the teacher network prevents the student network from overfitting in the target domain and protects the general information from the domain-specific training.

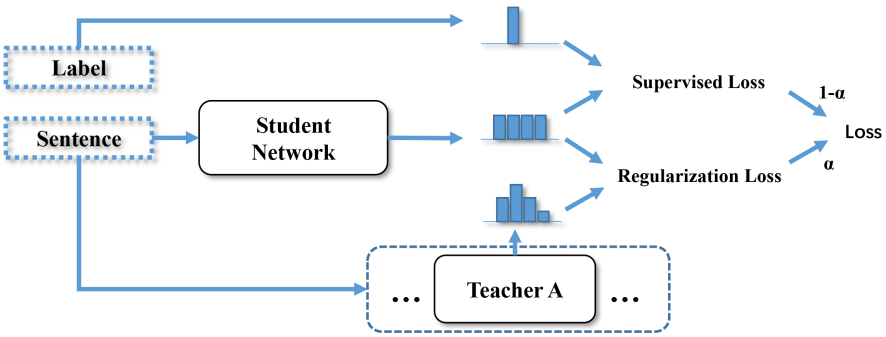

Figure 2: The architecture of our neural regularization strategy for the domain adaptation of Chinese word segmentation.

Our neural regularization is different from the traditional regularization used in the domain adaptation such as weights regularization (Blitzer et al., 2007; Rozantsev et al., 2016). The weights regularization works as a global setting that prevents any weights deviating from source domain models. Our neural regularization is more meticulous and tunes the loss of each sample respectively.

3 Experiments

3.1 Dataset

Following previous Chinese word segmentation domain adaptation methods, we employ the Chinese Treebank (CTB) (Xue et al., 2005) as the source domain data. The Patent (Li and Xue, 2014) and Zhuxian (Zhang et al., 2014) are used as the target domain data. The patent is often a description of a specifically designed system, which contains a high concentration of technical terms. Zhuxian is a Internet novel and has a different writing style comparing to CTB. Zhuxian also contains many novel specific named entity. The statistics of the data is shown in Table 1. It is obvious that the amount of source domain data is much larger than target domain data.

We also perform our method between different genres of CTB9. The *Newswire* (nw) in CTB9 is chosen as the source domain data. The *Weblogs* (wb), *SMS/Chat messages* (sc) and *conversational speech* (cs) are employed as the target domain data. We split each genre into train, development, test set, and the filelist is shown in Table 3. The statistics of the data is shown in Table 2. Note that in the CTB9, the source domain nw is not significantly larger than target domain such as wb, cs. The nw is even smaller comparing to sc.

Type	Sec.	Source		Target	
		CTB5	CTB7	Patent	Zhuxian
sent.	train	18k	36k	11k	2.4k
words.		641k	839k	345k	67.6k
sent.	dev.	0.35k	4.8k	1.5k	0.79k
words.		6.8k	120k	46.2k	20.4k
sent.	test	0.35k	11k	1.5k	1.4k
words.		8.0k	241k	48.4k	34.4k

Table 1: Statistics of source and target datasets

Type	Sec.	CTB9			
		nw	wb	sc	cs
sent.	train	8.1k	8.3k	35.2k	12.7k
words.		197k	167k	242k	124k
sent.	dev.	1.1k	0.80k	4.3k	1.9k
words.		26.5k	21.3k	30.6k	17.6k
sent.	test	1.1k	1.1k	4.5k	2.1k
words.		26.7k	21.7k	30.6k	18.9k

Table 2: Statistics of genres used in our experiments. nw refers to *Newswire*. wb, sc and cs refer to *Weblogs*, *SMS/Chat messages* and *conversational speech*.

3.2 Hyper-Parameter Settings

In the experiments, the hyper parameters are chosen through grid search. The filters are set to 300 feature maps for each window size ranging from 2 to 5 characters. A dropout of 50% is adopted. The size of unigram and bigram character embeddings is 200 with a 20% dropout.[1] The training is done through stochastic gradient descent with Adadelta (Zeiler, 2012). The hyper-parameter T is set to 2. The α is set to 0.4 for *Zhuxian 300s* and CTB9 *Weblogs*, 0.5 for *Patent 10*, 0.6 for CTB9 *conversational speech*, *Zhuxian 600s* and *Patent 20*, 0.7 for *Patent 100*, 0.8 for CTB9 *SMS/Chat messages*

[1]We use the bigram embedding following the implements of (Zhang et al., 2016).

Genres	Sec.	ID list
nw	dev.	4041-4045, 0924-0927, 0830-0857, 0531-0535, 0443-0448, 0254-0288.
	test	4046-4050, 0928-0931, 0858-0885, 0536-0540, 0449-0454, 0289-0325.
wb	dev.	4332-4336.
	test	4337-4411.
sc	dev.	6548-6623.
	test	6624-6700.
cs	dev.	7014-7015.
	test	7016-7017.

Table 3: The split filelist of each genre. We only list the filelist of development and test data. The rest of data in each genre is used as training data.

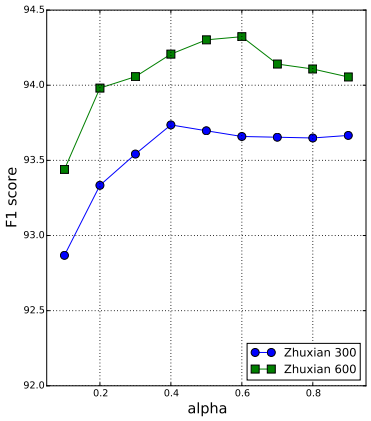

Figure 3: The results of our neural regularized method under different hyper-parameter α in Zhuxian development data.

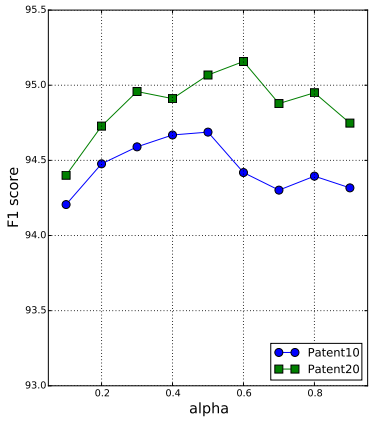

Figure 4: The results of our neural regularized method under different hyper-parameter α in Patent development data.

according to the performance on the development set. [2] The beam size of beam search is 10. We pre-train the embeddings using the publicly available Chinese Wikipedia corpus with *word2vec*. The teacher network and student network share the same architecture and hyper-parameter setting for simplicity.

3.3 Regularization Weights

For the traditional weight regularization, a hyper-parameter is often included to control the degree of regularization. When the network is regularized heavily, it often leads to under-fitting. While slight regularization may lead to over-fitting. In this section, we employ experiments to explore the effectiveness of the balancing hyper-parameter α used in our neural regularization. We want to know how the hyper-parameter α influences the performance of our method.

We perform our method in experiments between both CTB5 to Zhuxian and CTB7 to Patent. The hyper-parameter settings of the segmenter is same as mentioned in Sec. 3.2. The hyper-parameter α is searched ranging from 0.1 to 0.9 with a step size of 0.1. The results of CTB5 to Zhuxian and CTB7 to Patent are shown in Fig. 3 and Fig. 4.

Take CTB5 to Zhuxian as an example, we train our teacher network with training data from CTB5 and perform our neural domain adaptation method to a student network using *Zhuxian 300s* and *Zhuxian 600s*. For both *Zhuxian 300s* and *Zhuxian 600s* data, the performance of our student network first improves and then decreases with the increasing of hyper-parameter α. The decrease of performance is similar to traditional regularization with heavy or insufficient regularization.

And the best performance on the development is achieved in $\alpha = 0.6$ for *Zhuxian 600s*, $\alpha = 0.4$ for *Zhuxian 300s*. Note that a higher α makes the student network more focus on the target domain.

[2] *Patent 10*, *Patent 20* and *Patent 100* refer to the 10%, 20% and 100% of the Patent training data. *Zhuxian 300s* and *Zhuxian 600s* refer to the 300 and 600 sentences of the Zhuxian training data.

Methods	P	R	F1
(Li and Xue, 2016)			
Baseline	86.10	86.30	86.20
Patent 100	94.96	95.19	95.08
Ours			
Baseline	86.31	86.30	86.31
Mix *Patent 100*	94.56	94.39	94.47
Patent 100	95.13	95.26	95.20
+Patent 10	94.57	94.54	94.56
+Patent 20	94.95	95.09	95.02
+Patent 100	95.57	95.81	**95.69**

Table 4: The results between CTB7 and Patent. *Patent 10*, *Patent 20*, *Patent 100* refers to 10%, 20%, 100% of Patent train set. Mix refers to the method of training the model with mixed training data from Source and Target.

Methods	P	R	F1
(Zhang et al., 2014)			
Baseline	-	-	87.71
+Self-Training	-	-	88.62
+300	-	-	92.44
+300 +Self-Training	-	-	93.24
+3K +300	-	-	93.27
+3K +300 +Self-Training	-	-	93.98
+600	-	-	93.09
+600 +Self-Training	-	-	93.77
Ours			
Baseline	85.91	85.05	85.48
Mix 300	92.08	91.42	91.75
Mix 600	93.14	92.69	92.92
+300	93.61	93.30	93.45
+600	94.43	94.11	**94.27**

Table 5: The results between CTB5 and Zhuxian

The best development performance is achieved with different α is quite reasonable, because when the target domain data is becoming more and more sufficient, we can rely more on target domain data. And when the target domain data is sufficient to train a effective model by itself, we can use $\alpha = 1.0$ to turn off the regularization finally.

The similar results can also be found in the experiments between CTB7 to *Patent 10* and *Patent 20*. The best preformance of the student network is achieved when $\alpha = 0.5$ for *Patent 10*, $\alpha = 0.6$ for *Patent 20*.

3.4 Main Results

From CTB7 to Patent

We compare our neural regularized method with models from (Li and Xue, 2016) for the adaptation from CTB7 to Patent. The results are shown in Table 4. The performance of *Baseline* refers to the target domain performance of a baseline segmenter trained on source domain without any domain adaptation method. Li and Xue (2016) use a CRFs model as baseline model and improve the model from (Li and Xue, 2014). As shown in Table 4, the performance of our baseline model is comparable with their baseline model.

Li and Xue (Li and Xue, 2016) propose manually-crafted features to explore the domain-specific information in the patents and improve the accuracy of Chinese patent word segmentation. The manually-crafted features can be divided into *In-domain features* and *Out-of-domain features*. These features are used to model both the domain-specific characters combination and common cross-domain characteristics. They use the train set of Patent to train their model and the result is shown as *Patent 100*.

We employ the baseline model trained on the source domain as the teacher network and apply our neural regularized domain adaptation method to the student network with target domain data. Our method achieves a comparable performance with their model using only 20% of the Patent train set. We also list the performance of our method with 10%, 100% Patent train set as *+Patent 10* and *+Patent 100*. As the target domain data is often considered much 'expensive' comparing to publicly available source domain data, it is better to use as less target domain data as possible.

From CTB5 to Zhuxian

We also compare our methods with methods from (Zhang et al., 2014) for the adaptation from CTB5 to Zhuxian. The results are shown in Table 5. For (Zhang et al., 2014), manual annotated lexicon *3K*, self-training and two train set with 300/600 sentences are adopted. The annotated lexicon is used as plugins to the model for different domains through feature templates. The self-training method uses the model with lexicon features to label target domain sentences. Then the automatically labelled sentences are combined with source domain data to extend the training data. The annotated target domain sentences are directly mixed with source domain data as training data.

We train our teacher network with CTB5 training data and apply the teacher network to regularize the target domain specific training of the student network with our neural regularized domain adaptation method. Although Zhang et al. (2014) employ a joint model of word segmentation and POS tagging as baseline model, which

is stronger than our single-task baseline model. Our neural regularized domain adaptation method still achieves a better result under the same target domain resources. It shows the effectiveness of our neural regularization method on exploring target domain information and preserving general knowledge.

3.5 Result Analysis

In this section, we show and analyse the results of different model on the target domain test data. We take the Patent as an example and pick three sentences from the test set of Patent as shown in Fig. 5. The *Baseline* in the figure refers to a baseline segmenter trained on source domain without any domain adaptation method. The *Patent20* in the figure refers to a baseline segmenter trained on target domain data *Patent 20* without any regularization from source domain. *Our method* refers to the model trained with our neural regularized domain adaptation method utilizing both the source domain teacher network and target domain data.

Take the third sentence as an example, the meaning of this sentence is "after the blank rod is sent,". This sentence contains both domain-specific words like "blank", "rod" and general words such as "after", "is sent". The *Baseline* is trained on source domain lacking the target domain-specific information, and therefore, makes mistakes when handling the domain-specific words. For example, the *Baseline* did not segment the "blank" and "rod" correctly in the third sentence.

The *Patent20* is trained on target domain data, but the training data is insufficient and leads to the lack of general knowledge. As shown in the figure, the *Patent20* segments the domain-specific words correctly while makes mistakes when facing the general words. The *Patent20* did not segment the general word "is sent" correctly.

Finally, with our neural regularized domain adaptation method, the neural model segments both domain-specific and general words correctly. It shows that our method explores the domain-specific information and preserves the general knowledge at the same time. The similar results can also be observed in other two sentences.

3.6 Experiments on the CTB9

We also perform our method between different genres of CTB9 as shown in Table 6. As mentioned in Sec. 3.1, in the CTB9, the source domain data is not significantly larger than target domain data. The nw, wb, sc, cs refer to *Newswire*, *Weblogs*, *SMS/Chat messages*, *conversational speech* respectively. The nw is chosen as the source domain data and the others are employed as the target domain data.

nw $->$ wb			
Methods	P	R	F1
Baseline	86.45	88.04	87.24
Target only	90.90	89.67	90.28
Mix	92.64	92.41	92.52
Our method	92.91	92.40	**92.65**
nw $->$ sc			
Methods	P	R	F1
Baseline	80.49	80.98	80.74
Target only	94.93	94.21	94.57
Mix	94.93	94.66	94.80
Our method	94.92	94.91	**94.92**
nw $->$ cs			
Methods	P	R	F1
Baseline	82.86	82.21	82.53
Target only	95.94	95.64	95.79
Mix	96.10	96.02	96.06
Our method	96.32	96.68	**96.50**

Table 6: The experiment results of CTB9 between nw and wb, sc, cs genres.

The *Baseline* refers to the target domain performance of a baseline segmenter trained with the *Newswire* data. The *Target only* refers to the target domain performance of a baseline segmenter trained with the target domain data only. The *Our method* refers to the performance of our neural regularized domain adaptation method.

Because few previous methods are adopted in CTB9, we only compare our method with a baseline model trained on source domain and a baseline model trained on target domain providing the performance of our method for further comparision of domain adaptation methods in the future. Our method achieves improvement over both *Baseline* and *Target only*.

4 Related Work

Domain adaptation can be roughly divided into the fully-supervised and the semi-supervised domain adaptation (Daume III, 2007). Much work has been done in this area. For example, in the fully-supervised scenario, the well-known method *Easy Adaptation* is proposed to augment the feature space of both source and target data and then the combined feature space is used to train cross-domain model(Daume III, 2007). Daumé III et al. (2010) then proposed a semi-supervised extension

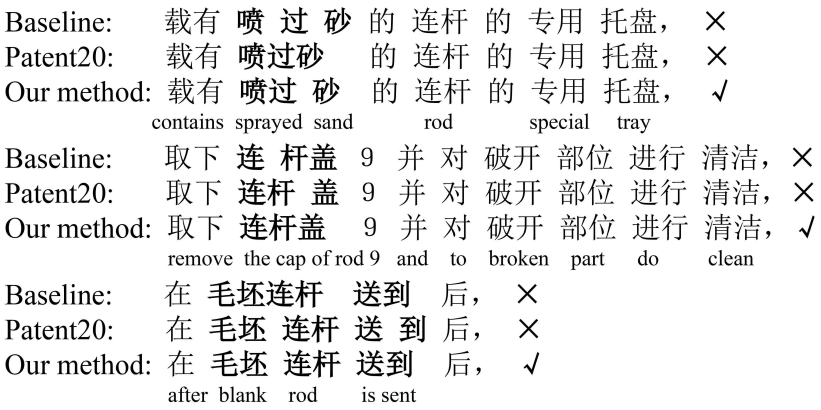

Figure 5: The results of different model on the same three test sentences of the Patent.

of the *Easy Adaptation*, which harnesses unlabeled target domain data to ameliorate the transfer of information from source to target.

Knowledge Distillation is first proposed to compress the knowledge of a source model (Bucilu et al., 2006) into a smaller target model. Hinton et al. (2015) developed this approach using a different compression technique. (Lopez-Paz et al., 2015) proposed a framework unifying *Knowledge Distillation* (Hinton et al., 2015) and *privileged information* (Vapnik and Izmailov, 2015). As *Knowledge Distillation* is able to transfer knowledge, it has been extended to other tasks. Li and Hoiem (2016) adopted a method to gradually add new capabilities to a multi-task system while preserve the original capabilities. Hu et al. (2016) employed *Knowledge Distillation* to enhance various types of neural networks with declarative first-order logic rules. Ao et al. (2017) utilized the unlabeled data to transfer the knowledge from the source models and SVM was used as base classifier to efficiently solve the imitation parameter.

For Chinese word segmentation, previous works mainly focused on semi-supervised domain adaptation methods. Unsupervised character clustering feature and self-training method were explored (Liu and Zhang, 2012). The partially-annotated data was found to be more effective than lexicons based features (Liu et al., 2014). The effectiveness of manually annotated lexicons and sentences were explored and compared (Zhang et al., 2014). Li and Xue (2014) designed *In-domain* and *Out-of-domain* features to capture the distributional characteristics in patents and annotated a significant amount of Chinese patent data (Li and Xue, 2016). Qiu and Zhang (2015) reduced the burden of the manually annotated lexicons by mining entities in Chinese novel with information extraction techniques.

5 Conclusion

In this paper, we focus on the fully-supervised domain adaptation for Chinese word segmentation and propose a neural regularized domain adaptation method. As the amount of annotated data in target domain is limited, it is insufficient to directly train a effective model and avoid overfitting. In our method, teacher networks trained in source domain are employed as general background knowledge to regularize the training process of the student network.

We investigate that the effect of hyper-parameter α is similar to the hyper-parameter of traditional weights regularization. Then we evaluate our method in the adaptation of two target domain datasets, from CTB5 to Zhuxian and from CTB7 to Patent. Experiments show that our neural regularized domain adaptation method can achieve improved performance with previous domain adaptation methods. We also analyse the results and display some examples, which shows that our method explores the domain-specific information and preserves the general knowledge at the same time. Finally, we apply our method to different genres of CTB9 and provide the results for further comparision in the future.

Acknowledge

This work was supported by Beijing Natural Science Foundation (4174098), National Natural Science Foundation of China (61702047) and the Fundamental Research Funds for the Central Universities (2017RC02).

References

Shuang Ao, Xiang Li, and Charles X Ling. 2017. Fast generalized distillation for semi-supervised domain adaptation. In *AAAI*. pages 1719–1725.

John Blitzer, Mark Dredze, Fernando Pereira, et al. 2007. Biographies, bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification. In *ACL*. volume 7, pages 440–447.

Cristian Bucilu, Rich Caruana, and Alexandru Niculescu-Mizil. 2006. Model compression. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, pages 535–541.

Xinchi Chen, Xipeng Qiu, and Xuanjing Huang. 2016. A long dependency aware deep architecture for joint chinese word segmentation and pos tagging. *arXiv preprint arXiv:1611.05384* .

Xinchi Chen, Zhan Shi, Xipeng Qiu, and Xuanjing Huang. 2017. Adversarial multi-criteria learning for chinese word segmentation. *arXiv preprint arXiv:1704.07556* http://arxiv.org/abs/1704.07556.

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.

Hal Daume III. 2007. Frustratingly easy domain adaptation. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*. Association for Computational Linguistics, pages 256–263. http://aclweb.org/anthology/P07-1033.

Hal Daumé III, Abhishek Kumar, and Avishek Saha. 2010. Frustratingly easy semi-supervised domain adaptation. In *Proceedings of the 2010 Workshop on Domain Adaptation for Natural Language Processing*. Association for Computational Linguistics, pages 53–59.

Thomas Emerson. 2005. The second international chinese word segmentation bakeoff. In *Proceedings of the Fourth SIGHAN Workshop on Chinese Language Processing*. http://aclweb.org/anthology/I05-3017.

Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. 2016. Domain-adversarial training of neural networks. *Journal of Machine Learning Research* 17(59):1–35.

Jun Hatori, Takuya Matsuzaki, Yusuke Miyao, and Jun'ichi Tsujii. 2012. Incremental joint approach to word segmentation, pos tagging, and dependency parsing in chinese. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics, pages 1045–1053. http://aclweb.org/anthology/P12-1110.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. 2015. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531* .

Zhiting Hu, Xuezhe Ma, Zhengzhong Liu, Eduard Hovy, and Eric Xing. 2016. Harnessing deep neural networks with logic rules. *arXiv preprint arXiv:1603.06318* .

Wenbin Jiang, Liang Huang, and Qun Liu. 2009. Automatic adaptation of annotation standards: Chinese word segmentation and pos tagging – a case study. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*. Association for Computational Linguistics, pages 522–530. http://aclweb.org/anthology/P09-1059.

Jin Kiat Low, Hwee Tou Ng, and Wenyuan Guo. 2005. A maximum entropy approach to chinese word segmentation. In *Proceedings of the Fourth SIGHAN Workshop on Chinese Language Processing*. http://aclweb.org/anthology/I05-3025.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics, pages 1746–1751.

Si Li and Nianwen Xue. 2014. Effective document-level features for chinese patent word segmentation. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*. Association for Computational Linguistics, pages 199–205. https://doi.org/10.3115/v1/P14-2033.

Si Li and Nianwen Xue. 2016. Towards accurate word segmentation for chinese patents. *arXiv preprint arXiv:1611.10038* .

Zhizhong Li and Derek Hoiem. 2016. Learning without forgetting. In *European Conference on Computer Vision*. Springer, pages 614–629.

Yang Liu and Yue Zhang. 2012. Unsupervised domain adaptation for joint segmentation and pos-tagging. In *Proceedings of COLING 2012: Posters*. The COLING 2012 Organizing Committee, pages 745–754. http://aclweb.org/anthology/C12-2073.

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 the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics, pages 864–874. https://doi.org/10.3115/v1/D14-1093.

David Lopez-Paz, Léon Bottou, Bernhard Schölkopf, and Vladimir Vapnik. 2015. Unifying distillation and privileged information. *arXiv preprint arXiv:1511.03643* .

Fuchun Peng, Fangfang Feng, and Andrew McCallum. 2004. Chinese segmentation and new word detection using conditional random fields. In *COLING 2004: Proceedings of the 20th International Conference on Computational Linguistics*. http://aclweb.org/anthology/C04-1081.

Likun Qiu and Yue Zhang. 2015. Word segmentation for chinese novels. In *AAAI*. pages 2440–2446.

Artem Rozantsev, Mathieu Salzmann, and Pascal Fua. 2016. Beyond sharing weights for deep domain adaptation. *arXiv preprint arXiv:1603.06432* .

Sebastian Ruder, Parsa Ghaffari, and John G Breslin. 2017. Knowledge adaptation: Teaching to adapt. *arXiv preprint arXiv:1702.02052* .

Weiwei Sun. 2011. A stacked sub-word model for joint chinese word segmentation and part-of-speech tagging. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*. Association for Computational Linguistics, pages 1385–1394. http://aclweb.org/anthology/P11-1139.

Vladimir Vapnik and Rauf Izmailov. 2015. Learning using privileged information: similarity control and knowledge transfer. *Journal of Machine Learning Research* 16:2023–2049.

Naiwen Xue, Fei Xia, Fu-Dong Chiou, and Marta Palmer. 2005. The penn chinese treebank: Phrase structure annotation of a large corpus. *Natural language engineering* 11(02):207–238.

Nianwen Xue. 2003. Chinese word segmentation as character tagging. In *International Journal of Computational Linguistics & Chinese Language Processing, Volume 8, Number 1, February 2003: Special Issue on Word Formation and Chinese Language Processing*. pages 29–48. http://aclweb.org/anthology/O03-4002.

Dong Yu, Kaisheng Yao, Hang Su, Gang Li, and Frank Seide. 2013. Kl-divergence regularized deep neural network adaptation for improved large vocabulary speech recognition. In *IEEE International Conference on Acoustics, Speech and Signal Processing*. pages 7893–7897.

Matthew D Zeiler. 2012. Adadelta: an adaptive learning rate method. *arXiv preprint arXiv:1212.5701* .

Meishan Zhang, Yue Zhang, Wanxiang Che, and Ting Liu. 2014. Type-supervised domain adaptation for joint segmentation and pos-tagging. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics*. Association for Computational Linguistics, pages 588–597. https://doi.org/10.3115/v1/E14-1062.

Meishan Zhang, Yue Zhang, and Guohong Fu. 2016. Transition-based neural word segmentation. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics, pages 421–431. https://doi.org/10.18653/v1/P16-1040.

Yue Zhang and Stephen Clark. 2008. Joint word segmentation and pos tagging using a single perceptron. In *Proceedings of ACL-08: HLT*. Association for Computational Linguistics, pages 888–896. http://aclweb.org/anthology/P08-1101.