Coupling Distant Annotation and Adversarial Training for Cross-Domain Chinese Word Segmentation

Ning Ding[1,2] , **Dingkun Long**[2], **Guangwei Xu**[2],
Muhua Zhu[2], **Pengjun Xie**[2], **Xiaobin Wang**[2], **Hai-Tao Zheng**[1*]
[1]Tsinghua University, China [2]Alibaba Group
{dingn18}@mails.tsinghua.edu.cn, {zhumuhua}@gmail.com,
{dingkun.ldk,kunka.xgw}@alibaba-inc.com,
{chengchen.xpj,xuanjie.wxb}@alibaba-inc.com,
{zheng.haitao}@sz.tsinghua.edu.cn

Cross-domain CWS

- Gap of domain distributions
- Out of vocabulary (OOV) problem

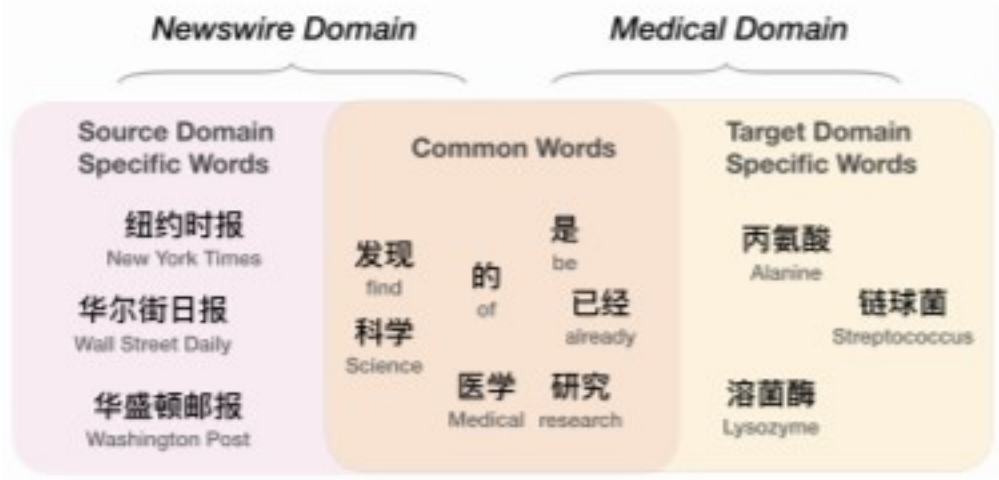

Figure 1: Different word distributions for the newswire domain and the medical domain.

Model

- **distant annotation**： automatically construct labeled target domain data with no requirement for human-curated domain-specific dictionaries.
- **adversarial training**： training jointly on the source domain dataset and the distantly constructed target domain dataset

Model

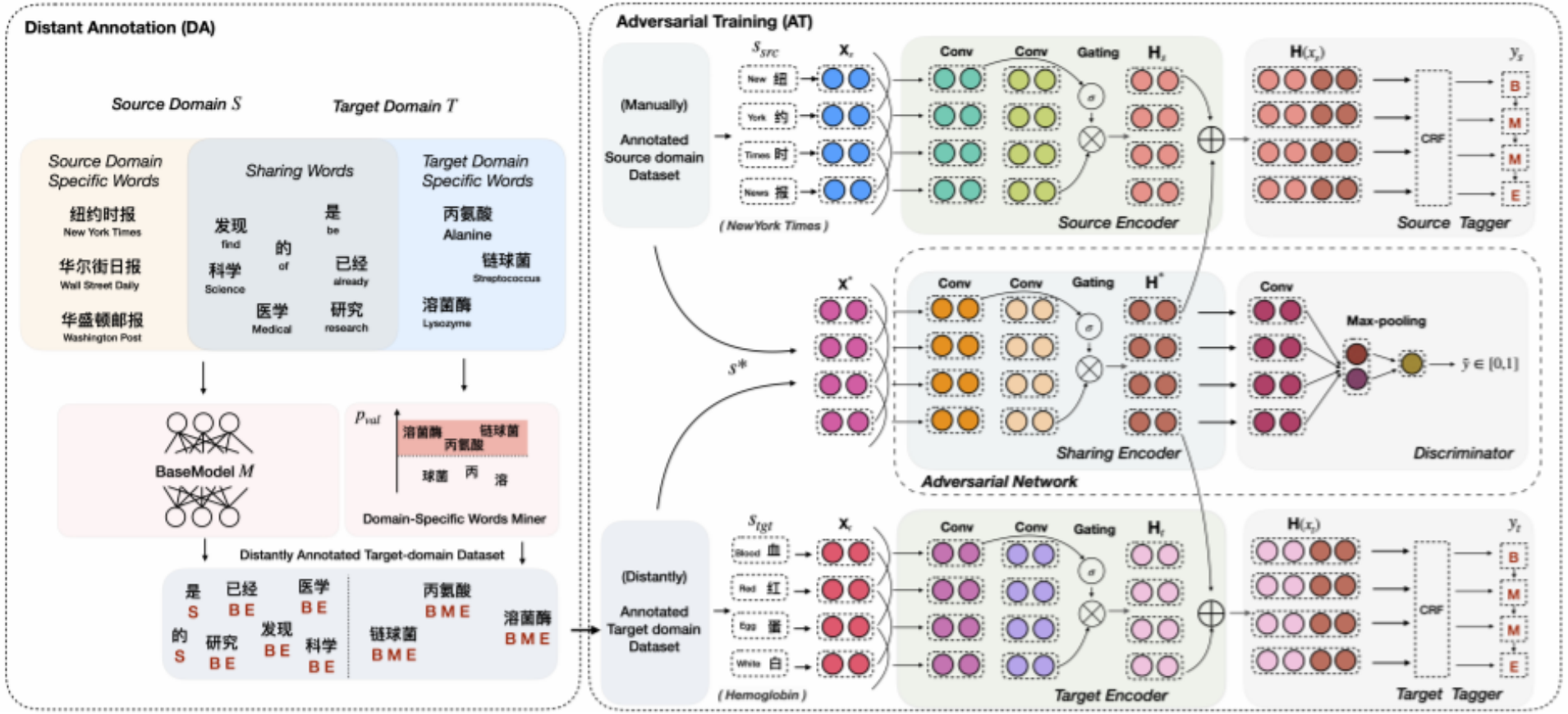

Figure 2: Detailed architecture of DAAT, the left part is the structure of the Distant Annotation (DA) module. The annotated dataset on target domain will be sent to the Adversarial Training (AT) module on the right part.

Distant Annotation

- DA has two main modules, including **a base segmenter** and **a Domainspecific Words Miner**

- Base Segmenter：GCNN-CRF
- Words Miner：

Given large raw text on target domain and a base segmenter, we can obtain a set of segmented texts $\Gamma = \{T_1, T_2, ..., T_N\}$, where stop-words are removed. Then let $\gamma = \{t_1, t_2, ..., t_m\}$ denote all the n-gram sequences extracted from Γ. For each sequence t_i, we need to calculate the possibility that it is a valid word. In this procedure, four factors are mainly considered.

Words Miner

1) *Mutual Information* (MI). MI (Kraskov et al., 2004) is widely used to estimate the correlation of two random variables. Here, we use mutual information between different sub-strings to measure the internal tightness for a text segment, as shown in Figure 3(a). Further, in order to exclude extreme cases, it is necessary to enumerate all the sub-string candidates. The final MI score for one sequence t_i consists of n characters $t_i = \{c_1 ... c_n\}$ is defined as:

$$\mathrm{MIS}(t_i) = \min_{j \in [1:n]} \{ \frac{p(t_i)}{p(c_1 ... c_j) \cdot p(c_{j+1} ... c_n)} \}, \quad (5)$$

where $p(\cdot)$ denotes the probability given the whole corpus Γ.

2) *Entropy Score* (ES). Entropy is a crucial concept aiming at measuring the uncertainty of random variables in information theory (Jaynes, 1957).

Thus, we can use ES to measure the uncertainty of candidate text fragment, since higher uncertainty means a richer neighboring context. Let $N_l(t_i) = \{l_1, ..., l_k\}$ and $N_r(t_i) = \{r_1, ..., r_{k'}\}$ be the set of left and right adjacent characters for t_i. The left entropy score ES_l and right entropy ES_r of t_i can be formulated as $\mathrm{ES}_l(t_i) = \sum_j^k -p(l_j) \log p(l_j)$ and $\mathrm{ES}_r(t_i) = \sum_j^{k'} -p(r_j) \log p(r_j)$ respectively. We choose $\min(\mathrm{ES}_l(t_i), \mathrm{ES}_r(t_i))$ as the final score for t_i. Hence, $\mathrm{ES}(t_i)$ could explicitly represent the external flexibility for a text segment (as shown in Figure 3(b)), and further serve as an important indicator to judge whether the segment is an independent word.

Words Miner

3) *tf-idf*. tf-idf is a widely used numerical statistic that can reflect how important a word is to a document in a collection or corpus. As illustrated in Figure 1, most of the domain-specific words are noun entities, which share a large weighting factor in general.

In this work, we define a word probability score $p_{val}(t_i)$ to indicate how likely t_i can be defined as a valid word.

$$p_{val}(t_i) = \sigma(\mathrm{N}[\mathrm{MIS}(t_i)] + \mathrm{N}[\mathrm{ES}(t_i)] + \mathrm{N}[\mathrm{tfidf}(t_i)]), \tag{6}$$

where σ denotes the sigmoid function and N denotes normalization operation with the max-min method.

4) *Word frequency*. If t_i is a valid word, it should appear repeatedly in Γ.

Finally, by setting an appropriate threshold for $p_{val}(t_i)$ and word frequence, the *Domain-Specific Words Miner* could effectively explore domain-specific words, then construct the domain-specific word collection $\mathcal{C}$ for the target domain. In this work, we only consider words t_i with $p_{val}(t_i) \geq 0.95$ and frequency larger than 10.

Distant Annotation

The left part of Figure 2 illustrates the data construction process of *DA*. First, we utilize the *Domain-specific Words Miner* to build the collection $\mathcal{C}$ for the target domain. Take sentence "溶酶菌的科学研究 (Scientific research on lysozyme)" as an example, we use the forward maximizing match algorithm based on $\mathcal{C}$, which shows that "溶酶菌 (lysozyme)" is a valid word. Hence, the labels of characters "溶", "酶", "菌" are "B", "M", "E". For the left part of the sentence, we adopt the baseline segmenter to perform the labelling process. "的科学研究" will be assigned with {"S", "B"."E", "B", "E"}. To this end, we are able to automatically build annotated dataset on the target domain.

Adversarial Training

domain $\mathcal{S}$ and target domain $\mathcal{T}$. There are three encoders to extract features with different emphases, and all the encoders are based on GCNN as introduced in section 3.1. For domain-specific features, we adopt two independent encoders E_{src} and E_{tgt} for source domain and target domain. For domain-agnostic features, we adopt a sharing encoder E_{shr} and a discriminator G_d, which will be both trained as adversarial players.

The discriminator G_d of the network aims to distinguish the domain of each sentence. Specifically, we will feed the final representation $\boldsymbol{H}^*$ of every sentence s to a binary classifier G_y where we adopt the text CNN network (Kim, 2014). G_y will produce a probability that the input sentence s is from the source domain or target domain. Thus, the loss function of the discriminator is:

$$\begin{aligned}\mathcal{L}_d = &- \mathbb{E}_{s\sim p_{\mathcal{S}}(s)}[\log G_y(E_{shr}(s)] \\ &- \mathbb{E}_{s\sim p_{\mathcal{T}}(s)}[\log (1 - G_y(E_{shr}(s))],\end{aligned} \tag{7}$$

Features generated by the sharing encoder E_{shr} should be able to fool the discriminator to correctly predict the domain of s. Thus, the loss function for the sharing encoder $\mathcal{L}_c$ is a flipped version of $\mathcal{L}_d$:

$$\begin{aligned}\mathcal{L}_c = &- \mathbb{E}_{s\sim p_{\mathcal{S}}(s)}[\log (1 - G_y(E_{shr}(s)])) \\ &- \mathbb{E}_{s\sim p_{\mathcal{T}}(s)}[\log G_y(E_{shr}(s)],\end{aligned} \tag{8}$$

Adversarial Training

Finally, we concatenate $\boldsymbol{H}$ and $\boldsymbol{H}^*$ as the final sequence representation of the input sentence. For s_{src} from source domain, $\boldsymbol{H}(s_{src}) = [\boldsymbol{H}_s \oplus \boldsymbol{H}_s^*]$, while for s_{tgt} from the target domain, $\boldsymbol{H}(s_{tgt}) = [\boldsymbol{H}_t \oplus \boldsymbol{H}_t^*]$. The final representation will be fed into the CRF tagger.

So far, our model can be jointly trained in an end-to-end manner with the standard back-propagation algorithm. More details about the adversarial training process are described in Algorithm 1. When there is no annotated dataset on the target domain, we could remove $\mathcal{L}_{tgt}$ during the adversarial training process and use the segmenter on source domain for evaluation.

Algorithm 1 Adversarial training algorithm.

Input: Manually annotated dataset $\mathcal{D}_s$ for source domain $\mathcal{S}$, and distantly annotated dataset $\mathcal{D}_t$ for target domain $\mathcal{T}$

for $i \leftarrow 1$ **to** *epochs* **do**
  for $j \leftarrow 1$ **to** *num_of_steps per epoch* **do**
    Sample mini-batches $\mathcal{X}_s \sim \mathcal{D}_s$, $\mathcal{X}_t \sim \mathcal{D}_t$
    if $j\%2 = 1$ **then**
      $loss = \mathcal{L}_{src} + \mathcal{L}_{tgt} + \mathcal{L}_d$
      Update θ w.r.t $loss$
    else
      $loss = \mathcal{L}_{src} + \mathcal{L}_{tgt} + \mathcal{L}_c$
      Update θ w.r.t $loss$
    end
  end
end

Dataset

Dataset			Sents	Words	Chars	Domain
SRC	PKU	Train	47.3K	1.1M	1.8M	News
		Test	6.4K	0.2M	0.3M	
TGT	DL	Full	40.0K	2.0M	2.9M	Novel
		Test	1.0K	32.0K	47.0K	
	FR	Full	148K	5.0M	7.1M	Novel
		Test	1.0K	17.0K	25.0K	
	ZX	Full	59.0K	2.1M	3.0M	Novel
		Test	1.0K	21K	31.0K	
	DM	Full	32.0K	0.7M	1.2M	Medical
		Test	1.0K	17K	30K	
	PT	Full	17.0K	0.6M	0.9M	Patent
		Test	1.0K	34.0K	57.0K	

Table 1: Statistics of datasets. The datasets of the target domain (TGT) are originally raw texts without golden segmentation, and the statistics are obtained by the baseline segmenter. The *DA* module will distantly annotate the datasets as mentioned in 3.1.

Baseline

We make comprehensive experiments with selective previous proposed methods, which are: **Partial CRF** (Liu et al., 2014) builds partially annotated data using raw text and lexicons via handcrafted rules, then trains the CWS model based on both labeled dataset (PKU) and partially annotated data using CRF. **CWS-DICT** (Zhang et al., 2018) trains the CWS model with a BiLSTM-CRF architecture, which incorporates lexicon into a neural network by designing handcrafted feature templates. For fair comparison, we use the same domain dictionaries produced by the *Domain-specific Words Miner* for **Partial CRF** and **CWS-DICT** methods. **WEB-CWS** (Ye et al., 2019) is a semi-supervised word-based approach using word embeddings trained with segmented text on target domain to improve cross-domain CWS.

Besides, we implement strong baselines to perform a comprehensive evaluation, which are: **GCNN (PKU)** uses the PKU dataset only, and we adopt the GCNN-CRF sequence tagging architecture (Wang and Xu, 2017). **GCNN (Target)** uses the distantly annotated dataset built on the target domain only. **GCNN (Mix)** uses the mixture dataset with both the PKU dataset and the distantly annotated target domain dataset. **DA** is a combination of GCNN (PKU) and domain-specific words. Details are introduced in 3.1. **AT** denotes the setting that we adopt adversarial training when no distantly annotated dataset on the target domain is provided, but the raw text is available.

Experiment

Dataset	Previous Methods (F1-score)			Ours (F1-score)					
	Partial CRF	CWS-DICT	WEB-CWS	AT	GCNN (PKU)	DA	GCNN(Mix)	GCNN (Target)	**DAAT**
DL	92.5	92.0	**93.5**	90.7	90.0	93.6	93.9	93.9	**94.1 (+0.6)**
FR	**90.2**	89.1	89.6	86.8	86.0	92.4	92.6	92.6	**93.1 (+2.9)**
ZX	83.9	88.8	**89.6**	85.0	85.4	90.4	90.6	90.7	**90.9 (+1.3)**
DM	**82.8**	81.2	82.2	81.0	82.4	83.8	83.9	84.3	**85.0 (+2.2)**
PT	85.0	**85.9**	85.1	85.1	87.6	89.1	89.3	89.3	**89.6 (+3.7)**

Table 3: The overall results on five datasets. The first block contains the latest cross-domain methods. And the second block reports the results for our implemented methods and DAAT. Numbers in the parentheses indicate absolute improvement than previous SOTA results.

Experiment

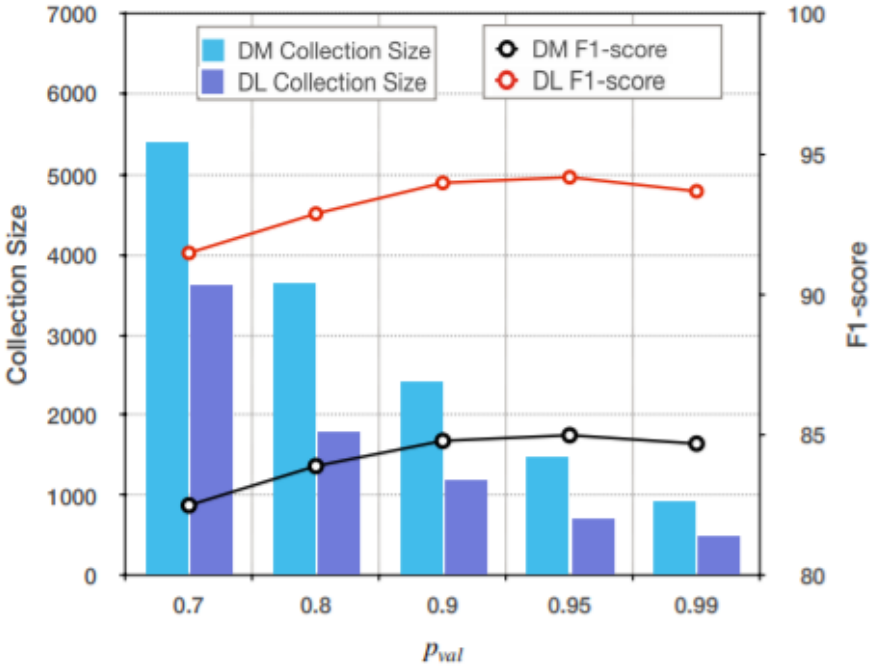

Figure 4: The impact of different p_{val} on mined collection size and model performance.

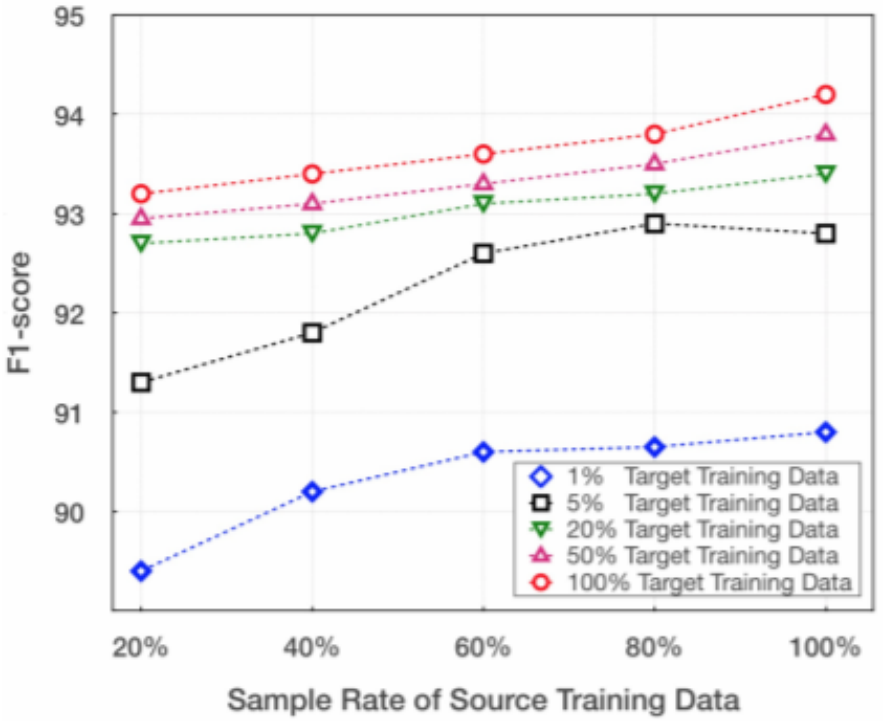

Figure 6: The impact of data amount for the source and target data on PKU (source, 47.3k sentences) and DL (target, 40.0k sentences).