Domain Adaptation for CRF-based Chinese Word Segmentation using Free Annotations

Yijia Liu †‡, Yue Zhang †, Wanxiang Che ‡, Ting Liu ‡, Fan Wu †
†Singapore University of Technology and Design
‡Research Center for Social Computing and Information Retrieval
Harbin Institute of Technology, China
{yjliu,car,tliu}@ir.hit.edu.cn {yue_zhang,fan_wu}@sutd.edu.sg

Abstract

Supervised methods have been the dominant approach for Chinese word segmentation. The performance can drop significantly when the test domain is different from the training domain. In this paper, we study the problem of obtaining partial annotation from freely available data to help Chinese word segmentation on different domains. Different sources of free annotations are transformed into a unified form of partial annotation and a variant CRF model is used to leverage both fully and partially annotated data consistently. Experimental results show that the Chinese word segmentation model benefits from free partially annotated data. On the SIGHAN Bakeoff 2010 data, we achieve results that are competitive to the best reported in the literature.

1 Introduction

Statistical Chinese word segmentation gains high accuracies on newswire (Xue and Shen, 2003; Zhang and Clark, 2007; Jiang et al., 2009; Zhao et al., 2010; Sun and Xu, 2011). However, manually annotated training data mostly come from the news domain, and the performance can drop severely when the test data shift from newswire to blogs, computer forums and Internet literature (Liu and Zhang, 2012).

Several methods have been proposed for solving the domain adaptation problem for segmentation, which include the traditional token- and type-supervised methods (Song et al., 2012; Zhang et al., 2014). While token-supervised methods rely on manually annotated target-domain sentences, type-supervised methods leverage manually assembled domain-specific lexicons to improve target-domain segmentation accuracies. Both methods are competitive given the same amount of annotation effects (Garrette and Baldridge, 2012; Zhang et al., 2014). However, obtaining manually annotated data can be expensive.

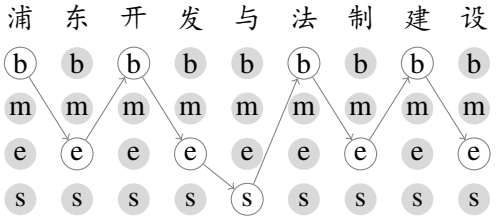


Figure 1: The segmentation problem, illustrated using the sentence "浦东 (Pudong) 开发 (development) 与 (and) 法制 (legal) 建设 (construction)". Possible segmentation labels are drawn under each character, where *b, m, e, s* stand for the beginning, middle, end of a multi-character word, and a single character word, respectively. The path shows the correct segmentation by choosing one label for each character.

On the other hand, there are *free* data which contain limited but useful segmentation information over the Internet, including large-scale unlabeled data, domain-specific lexicons and semi-annotated web pages such as Wikipedia. In the last case, word-boundary information is contained in hyperlinks and other markup annotations. Such free data offer a useful alternative for improving the segmentation performance, especially on domains that are not identical to newswire, and for which little annotation is available.

In this paper, we investigate techniques for adopting freely available data to help improve the performance on Chinese word segmentation. We propose a simple but robust method for constructing partial segmentation from different sources of free data, including unlabeled data and the Wikipedia. There has been work on making use of both unlabeled data (Sun and Xu, 2011; Wang et al., 2011) and Wikipedia (Jiang et al., 2013)

Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 864–874,
October 25-29, 2014, Doha, Qatar.

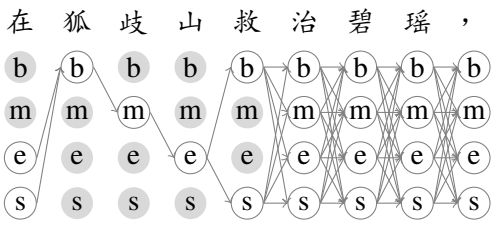
to improve segmentation. However, no empirical results have been reported on a unified approach to deal with different types of free data. We use a conditional random fields (Lafferty et al., 2001; Tsuboi et al., 2008) variant that can leverage the partial annotations obtained from different sources of free annotation. Training is achieved by a modification to the learning objective, incorporating partial annotation likelihood, so that a single model can be trained consistently with a mixture of full and partial annotation.

Experimental results show that our method of using partially annotated data can consistently improves cross-domain segmentation performance. We obtain results which are competitive to the best reported in the literature. Our segmentor is freely released at `https://github.com/ExpResults/partial-crfsuite`.

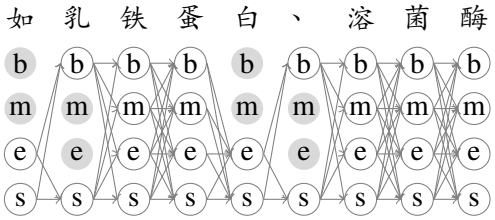
2 Obtaining Partially Annotated Data

We model the Chinese word segmentation task as a character sequence tagging problem, which is to give each character in a sentence a word-boundary tag (Xue and Shen, 2003). We adopt four tags, *b, m, e* and *s*, which represent the *b*eginning, *m*iddle, *e*nd of a multi-character word, and a *s*ingle character word, respectively. A manually segmented sentence can be represented as a tag sequence, as shown in Figure 1.

We investigate two major sources of freely-available annotations: lexicons and natural annotation, both with the help of unannotated data. To make use of the first source of information, we incorporate words from a lexicon into unannotated sentences by matching of character sequences, resulting in partially annotated sentences, as shown in Figure 2a. In this example, the word "狐岐山 (the Huqi Mountain)" in the unannotated sentence matches an item in the lexicon. As a result, we obtain a partially-annotated sentence, in which the segmentation ambiguity of the characters "狐 (fox)", "岐 (brandy road)" and "山 (mountain)" are resolved ("狐" being the beginning, "岐" being the middle and "山" being the end of the same word). At the same time, the segmentation ambiguity of the surrounding characters "在 (at)" and "救 (save)" are reduced ("在" being either a single-character word or the end of a multi-character word, and "救" being either a single-character word or the beginning of a multi-character word).



(a) "在 (at) 狐岐山 (Huqi Mountain) 救 治 (save) 碧 瑶 (Biyao)", where "狐岐山" matches a lexicon word.



(b) "如 (e.g.) 乳铁蛋白 (lysozyme) 、 溶菌酶 (lactoferrin)", where "乳铁蛋白" is a hyperlink.

Figure 2: Examples of partially annotated data. The paths show possible correct segmentations.

Natural annotation, which refers to word boundaries that can be inferred from URLs, fonts or colors on web pages, also result in partially-annotated sentences. Taking a web page shown in Figure 2b for example. It can be inferred from the URL tags on "乳铁蛋白" that "乳" should be either the beginning of a multi-character word or a single-character word, and "白" should be either the end a multi-character word or single-character word. Similarly, possible tags of the surrounding character "如" and "、" can also be inferred.

We turn both lexicons and natural annotation into the same form of *partial annotation* with same unresolved ambiguities, as shown in Figure 2, and use them together with available *full annotation* (Figure 1) as the training data for the segmentor. In this section, we describe in detail how to obtain partially annotated sentences from each resource, respectively.

2.1 Lexicons

In this scenario, we assume that there are unlabeled sentences along with a lexicon for the target domain. We obtain partially segmented sentences by extracting word boundaries from the unlabeled sentences with the help of the lexicon. Previous matching methods (Wu and Tseng, 1993; Wong and Chan, 1996) for Chinese word segmentation largely rely on the lexicons, and are generally considered being weak in ambiguity resolution (Gao

People's Daily	看到 (saw) 海南 (Hainan) 旅游业 (tourist industry) 充满 (full) 希望 (hope) saw tourist industry in Hainan is full of hope
Wikipedia	主要(mainly) 是(is) 旅游 (tourist) 业 (industry) 和(and) 软件 (software) 产业(industry) mainly is tourist industry and software industry

(a) Case of incompatible annotation on "旅游业(tourist industry)" between People's Daily and Wikipedia.

Literature	《说文解字 (Shuo Wen Jie Zi, a book) 段(segmented) 注(annotated) 》 the segmented and annotated version of Shuo Wen Jie Zi
Computer	每条(each) 记录(record) 被(is) 分隔(splitted) 为(into) 字段 (fields) each record is splitted into several fields

(b) Similar subsequence "字段(field)" is segmented differently under different domains in Wikipedia.

Table 1: Examples natural annotation from Wikipedia. Underline marks annotated words.

et al., 2005). But for obtaining the partial labeled data with lexicon, the matching method can still be a solution. Since we do not aim to recognize every word from sentence, we can select a lexicon with smaller coverage but less ambiguity to achieve relatively precise matching result.

In this paper, we apply two matching schemes to the same raw sentences to obtain partially annotated sentences. The first is a simple forward-maximum matching (FMM) scheme, which is very close to the forward maximum matching algorithm of Wu and Tseng (1993) for Chinese word segmentation. This scheme scans the input sentence from left to right. At each position, it attempts to find the longest subsequence of Chinese characters that matches a lexicon entry. If such an entry is found, the subsequence is tagged with the corresponding tags, and its surrounding characters are also constrained to a smaller set of tags. If no subsequence is found in the lexicon, the character is left with all the possible tags. Taking the sentence in Figure 2a for example. When the algorithm scans the second character, "狐", and finds the entry "狐岐山" in the lexicon, the subsequence of characters is recognized as a word, and tagged with *b, m* and *e*, respectively. At the same time, the previous character "在" can be inferred as only end of a multi-character word (*e*) or a single-character word (*s*). The second matching scheme is backward maximum matching, which can be treated as the application of FMM on the reverse of unlabeled sentences using a lexicon of reversed words.

To mitigate the errors resulting from one single matching scheme, we combine the two matching results by agreement. The basic idea is that if a subsequence of sentence is recognized as word by multiple matching results, it can be considered as a more precise annotation. Our algorithm reads partial segmentation by different methods and selects the subsequences that are identified as word by all methods as annotated words.

2.2 Natural Annotation

We use the Chinese Wikipedia for natural annotation. Partially annotated sentences are readily formed in Wikipedia by markup syntax, such as URLs. However, some subtle issues exist if the sentences are used directly. One problem is incompatibility of segmentation standards between the annotated training data and Wikipedia. Jiang et al. (2009) discuss this incompatibility problem between two corpora — the CTB and the People's Daily; the problem is even more severe on Wikipedia because it can be edited by any user. Table 1a shows a case of incompatible annotation between the People's Daily data and natural annotation in Wikipedia, where the three characters "旅游业" are segmented differently. Both can be treated as correct, although they have different segmentation granularities.

Another problem is the intrinsic ambiguity of segmentation. The same character sequence can be segmented into different words under different contexts. If the training and test data contain different contexts, the learned model can give incorrect results on the test data. This is particularly true across different domains. Table 1b gives such an example, where the character sequence "字段" is segmented differently in two of our test domains, but both cases exist in Wikipedia.

In summary, Wikipedia introduces both useful information for domain adaptation and harmful noise with negative effects on the model. To

achieve better performance of domain adaptation using Wikipedia, one intuitive approach is to select more domain-related data and less irrelevant data to minimize the risks that result from incompatible annotation and domain difference.

To this end, we assume that there are some raw sentences on the target domain, which can be used to evaluate the relevance between Wikipedia and target domain test data. We assume that URL-tagged entries reflect the segmentation standards of Wikipedia sentence, and use them to match Wikipedia sentences with the raw target domain data. If the character sequence of any URL-tagged entry in a Wikipedia sentence matches the target domain data, the Wikipedia sentence is selected for training. Another advantage of such data selection is that the training time consumption can be reduced by reducing the size of training data.

3 CRF for Word Segmentation

We follow the work of Zhao et al. (2010) and Sun and Xu (2011), and adopt the Conditional Random Fields (CRF) model (Lafferty et al., 2001) for the sequence labeling problem of word segmentation. Given an input characters sequence, the task is to assign one segmentation label from $\{b, m, e, s\}$ on each character. Let $\mathbf{x} = (x_1, x_2, ..., x_T)$ be the sequence of characters in sentence whose length is T, and $\mathbf{y} = (y_1, y_2, ..., y_T)$ be the corresponding label sequence, where $y_i \in Y$. The linear-chain conditional random field for Chinese word segmentation can be formalized as

$$p(\mathbf{y}|\mathbf{x}) = \frac{1}{Z} \exp \sum_{t=1}^{T} \sum_{k} \lambda_k f_k(y_t, y_{t-1}, \mathbf{x}) \quad (1)$$

where λ_k are the model parameters, f_k are the feature functions and Z is the probability normalizer.

$$Z = \sum_{\mathbf{y}} \exp \sum_{t=1}^{T} \sum_{k} \lambda_k f_k(y_t, y_{t-1}, \mathbf{x}) \quad (2)$$

We follow Sun and Xu (2011) and use the feature templates shown in Table 2 to model the segmented task. For ith character in the sentence, the n-gram features represent the surrounding characters of this character; $Type$ categorizes the character it into $digit$, $punctuation$, $english$ and $other$; $Identical$ indicates whether the input character is the same with its surrounding characters. This feature captures repetition patterns such as "试试 (try)" or "走走 (stroll)".

Type	Template
unigram	C_s $(i-3 < s < i+3)$
bigram	C_sC_{s+1} $(i-3 < s < i+2)$
	C_sC_{s+2} $(i-3 < s < i+1)$
type	$Type(C_i)$
	$Type(C_s)Type(C_{s+1})$ $(i-1 < s < i+2)$
identical	$Identical(C_s, C_{s+1})$ $(i-3 < s < i+1)$
	$Identical(C_s, C_{s+2})$ $(i-3 < s < i)$

Table 2: Feature templates for the ith character.

For fully-annotated training data, the learning problem of conditional random fields is to maximize the log likelihood over all the training data (Lafferty et al., 2001)

$$\mathcal{L} = \sum_{n=1}^{N} \log p(\mathbf{y}^{(n)}|\mathbf{x}^{(n)})$$

Here N is the number of training sentences. Both the likelihood $p(\mathbf{y}^{(n)}|\mathbf{x}^{(n)})$ and its gradient can be calculated by performing the forward-backward algorithm (Baum and Petrie, 1966) on the sequence, and several optimization algorithm can be adopted to learn parameters from data, including L-BFGS (Liu and Nocedal, 1989) and SGD (Bottou, 1991).

4 Training a CRF with partially annotated data

For word segmentation with partially annotated data, some characters in a sentence can have a definite segmentation label, while some can have multiple labels with ambiguities remaining. Taking the partially annotated sentence in Figure 2a for example, the corresponding potential label sequence for "在狐岐山救" is $\{(e, s), (b), (m), (e), (b, s)\}$, where the characters "狐", "岐" and "山" have fixed labels but for "在" and "救", some ambiguities exist. Note that the full annotation in Figure 1 can be regarded as a special case of partial annotation, where the number of potential labels for each character is one.

We follow Tsuboi et al. (2008) and model marginal probabilities over partially annotated data. Define the possible labels that correspond to the partial annotation as $\mathbf{L} = (L_1, L_2, ..., L_T)$, where each L_i is a non-empty subset of Y that corresponds to the set of possible labels for x_i. Let

$\mathbf{Y_L}$ be the set of all possible label sequences where $\forall \mathbf{y} \in \mathbf{Y_L}$, $y_i \in L_i$. The marginal probability of $\mathbf{Y_L}$ can be modeled as

$$p(\mathbf{Y_L}|\mathbf{x}) = \frac{1}{Z} \sum_{\mathbf{y} \in \mathbf{Y_L}} \exp \sum_{t=1}^{T} \sum_{k} \lambda_k f_k(y_t, y_{t-1}, \mathbf{x}) \quad (3)$$

Defining the unnormalized marginal probability as

$$Z_{\mathbf{Y_L}} = \sum_{\mathbf{y} \in \mathbf{Y_L}} \exp \sum_{t=1}^{T} \sum_{k} \lambda_k f_k(y_t, y_{t-1}, \mathbf{x}),$$

and the normalizer Z being the same as Equation 2, the log marginal probability of $\mathbf{Y_L}$ over N partially annotated training examples can be formalized as

$$\mathcal{L}_{\mathbf{Y_L}} = \sum_{n=1}^{N} \log p(\mathbf{Y_L}|\mathbf{x}) = \sum_{n=1}^{N} (\log Z_{\mathbf{Y_L}} - \log Z)$$

The gradient of the likelihood can be written as

$$\frac{\partial \mathcal{L}_{\mathbf{Y_L}}}{\partial \lambda_k} = \sum_{n=1}^{N} \sum_{t=1}^{T} \sum_{\substack{y_{\mathbf{Y_L}} \in L_t, \\ y'_{\mathbf{Y_L}} \in L_{t-1}}} f_k(y_{\mathbf{Y_L}}, y'_{\mathbf{Y_L}}, \mathbf{x}) p_{\mathbf{Y_L}}(y_{\mathbf{Y_L}}, y'_{\mathbf{Y_L}}|\mathbf{x}) - \sum_{n=1}^{N} \sum_{t=1}^{T} \sum_{y,y'} f_k(y, y', \mathbf{x}) p(y, y'|\mathbf{x})$$

Both $Z_{\mathbf{Y_L}}$ and its gradient are similar in form to Z. By introducing a modification to the forward-backward algorithm, $Z_{\mathbf{Y_L}}$ and $\mathcal{L}_{\mathbf{Y_L}}$ can be calculated. Define the *forward variable* for partially annotated data $\alpha_{\mathbf{Y_L},t}(j) = p_{\mathbf{Y_L}}(x_{\langle 1,\ldots,t \rangle}, y_t = j)$. A modification on the forward algorithm can be formalized as

$$\alpha_{\mathbf{Y_L},t}(j) = \begin{cases} 0 & j \notin L_t \\ \sum_{i \in L_{t-1}} \Psi_t(j, i, x_t) \alpha_{\mathbf{Y_L},t-1}(i) & j \in L_t \end{cases}$$

where $\Psi_t(j, i, x)$ is a potential function that equals $\sum_k \lambda_k f_k(y_t = j, y_{t-1} = i, x_t)$. Similarly, for the *backward variable* $\beta_{\mathbf{Y_L},t}$,

$$\beta_{\mathbf{Y_L},t}(i) = \begin{cases} 0 & i \notin L_t \\ \sum_{j \in L_{t+1}} \Psi_t(j, i, x_{t+1}) \beta_{\mathbf{Y_L},t+1}(j) & i \in L_t \end{cases}$$

$Z_{\mathbf{Y_L}}$ can be calculated by $\alpha_{\mathbf{Y_L}}(T)$, and $p_{\mathbf{Y_L}}(y, y'|x)$ can be calculated by $\alpha_{\mathbf{Y_L},t-1}(y') \Psi_t(y, y', x_t) \beta_{\mathbf{Y_L},t}(y)$.

Note that if each element in $\mathbf{Y_L}$ is constrained to one single label, the CRF model in Equation 3 degrades into Equation 1. So we can train a unified model with both fully and partially annotated data. We implement this CRF model based on a open source toolkit CRFSuite.[1] In our experiments, we use the L-BFGS (Liu and Nocedal, 1989) algorithm to learn parameters from both fully and partially annotated data.

5 Experiments

We perform our experiments on the domain adaptation test data from SIGHAN Bakeoff 2010 (Zhao et al., 2010), adapting annotated training sentences from People's Daily (PD) (Yu et al., 2001) to different test domains. The fully annotated data is selected from the People's Daily newspaper in January of 1998, and the four test domains from the SIGHAN Bakeoff 2010 include finance, medicine, literature and computer. Sample segmented data in the computer domain from this bakeoff is used as development set. Statistics of the data are shown in first half of Table 3. We use wikidump20140419[2] for the Wikipedia data. All the traditional Chinese pages in Wikipedia are converted to simplified Chinese. After filtering functional pages like *redirection* and removing duplication, 5.45 million sentences are reserved.

For comparison with related work on using a lexicon to improve segmentation, another set of test data is chosen for this setting. We use the Chinese Treebank (CTB) as the source domain data, and Zhuxian (a free Internet novel, also named as "Jade dynasty", referred to as ZX henceforth) as the target domain data.[3] The ZX data are written in a different style from newswire, and contains many out-of-vocabulary words. This setting has been used by Liu and Zhang (2012) and Zhang et al. (2014) for domain adaptation of segmentation and POS-tagging. We use the standard training, development and test split. Statistics of the test data annotated by Zhang et al. (2014) are shown in the second half of Table 3.

The data preparation method in Section 2 and the CRF method in Section 4 are used for all the experiments. Both recall of out-of-vocabulary words (R_{oov}) and F-score are used to evaluate the

[1] http://www.chokkan.org/software/crfsuite/

[2] http://dumps.wikimedia.org/zhwiki/20140419/

[3] Annotated target domain test data and lexicon are available from http://ir.hit.edu.cn/~mszhang/eacl14mszhang.zip.

	Data set	Train	Development	Test			
PD → SIGHAN		PD	Computer	Finance	Medicine	Literature	Computer
	# sent.	19,056	1,000	560	1,308	670	1,329
	# words	1,109,734	21,398	33,035	31,499	35,735	35,319
	OOV		0.1766	0.0874	0.1102	0.0619	0.1522

	Data set	Train	Development	Test	Unlabeled		Unlabeled
CTB5 → ZX		CTB5	ZX			Wikipedia	
	# sent.	18,086	788	1,394	32,023		5,456,151
	# words	493,934	20,393	34,355			
	OOV		0.1377	0.1550			

Table 3: Statistics of data used in this paper.

segmentation performance. There is a mixture of Chinese characters, English words and numeric expression in the test data from SIGHAN Bakeoff 2010. To test the influence of Wikipedia data on Chinese word segmentation alone, we apply regular expressions to detect English words and numeric expressions, so that they are marked as *not segmented*. After performing this preprocessing step, cleaned test input data are fed to the CRF model to give a relatively strong baseline.

5.1 Free Lexicons

5.1.1 Obtaining lexicons

For domain adaption from CTB to ZX, we use a lexicon released by Zhang et al. (2014). The lexicon is crawled from a online encyclopedia[4], and contains the names of 159 characters and artifacts in the Zhuxian novel. We follow Zhang et al. (2014) and name it **NR** for convenience of further discussion. The NR lexicon can be treated as a strongly domain-related, high quality but relatively small lexicon. It's a typical example of freely available lexicon over the Internet.

For domain adaptation from PD to medicine and computer, we collect a list of page titles under the corresponding categories in Wikipedia. For medicine, entries under *essential medicines*, *biological system* and *diseases* are collected. For computer, entries under *computer network*, *Microsoft Windows* and *software widgets* are selected. These lexicons are typical freely available lexicons that we can access to.

5.1.2 Obtaining Unlabeled Sentences

For ZX, partially annotated sentences are obtained using the *NR* lexicon and unlabeled ZX sentences by applying the matching scheme described in Section 2. The CTB5 training data and the partially annotated data are mixed as the final training data. Different amounts of unlabeled data are applied to the development test set, and results are shown in Figure 3. From this figure we can see that incorporating 16K sentences gives the highest accuracy, and adding more partial labeled data does not change the accuracy significantly. So for the ZX experiments, we choose the 16K sentences as the unlabeled data.

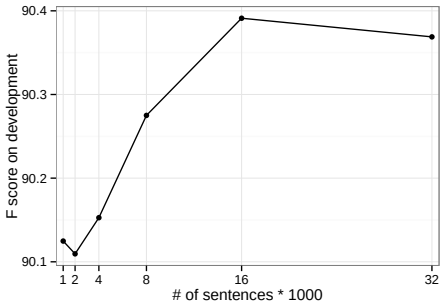


Figure 3: F-score on the development data when using different numbers of unlabeled data.

For the medicine and computer experiments, we selected domain-specific sentences by matching with the domain-specific lexicons. About 46K out of the 5.45 million wiki sentences contain subsequences in the medicine lexicon and 22K in the case of the computer domain. We randomly select 16K sentences as the unlabeled data for each domain, respectively.

5.1.3 Final results

We incorporate the partially annotated data obtained with the help of lexicon for each of the test domain. For adaptation from CTB to ZX, we trained our baseline model on the CTB5 training data with the feature templates in Table 2. For adaptation from PD to medicine and computer, we

[4] http://baike.baidu.com/view/18277.htm

Domain	ZX		Medicine		Computer	
	F	Roov	F	Roov	F	Roov
Baseline	87.50	73.65	91.36	72.95	93.16	84.02
Baseline+Lexicon Feature	90.36	80.69	91.60	74.39	93.14	84.27
Baseline+PA (Lex)	**90.63**	**84.88**	**91.68**	**74.99**	**93.47**	**85.63**
Zhang et al. (2014)	88.34	-	-	-	-	-

Table 4: Final result for adapting CTB to Zhuxian and adapting PD to the medicine and computer domains, using partially annotated data (referred to as *PA*) obtained from unlabeled data and lexicons.

trained our baseline model on the PD training data with the same feature template setting.

Previous research makes use of a lexicon by adding lexicon features directly into a model (Sun and Xu, 2011; Zhang et al., 2014), rather than transforming them into partially annotated sentences. To make a comparison, we follow Sun and Xu (2011) and add three lexicon features to represent whether c_i is located at the beginning, middle or the end of a word in the lexicon, respectively. For each test domain, the lexicon for the lexicon feature model consists of the most frequent words in the source domain training data (about 6.7K for CTB5 and 8K for PD, respectively) and the domain-specific lexicon we obtained in Section 5.1.1.

The results are shown in Table 4, where the first row shows the performance of the baseline models and the second row shows the performance of the model incorporating lexicon feature. The third row shows our method using partial annotation. On the ZX test set, our method outperforms the baseline by more than 3 absolute percentage. The model with partially annotated data performs better than the one with additional lexicon features. Similar conclusion is obtained when adapting from PD to medicine and computer. By incorporating the partially annotated data, the segmentation of lexicon words, along with the context, is learned.

We also compare our method with the work of Zhang et al. (2014), who reported results only on the ZX test data. We use the same lexicon settings. Our method gives better result than Zhang et al. (2014), showing that the combination of a lexicon and unannotated sentence into partially annotated data can lead to better performance than using a dictionary alone in type-supervision. Given that we only explore the use of free resource, combining a lexicon with *unannotated* sentences is a better option than using the lexicon directly. Zhang et al.'s concern, on the other hand, is to compare type- and token-annotation. Our partial annotation can thus be treated as a compromise to obtain some *pseudo* partial token-annotations when *full* token annotations are unavailable. Another thing to note is that the model of Zhang et al. (2014) is a joint model for segmentation and POS-tagging, which is generally considered stronger than a single segmentation model.

Method	Com. Dev	
	F	Roov
Baseline	93.56	83.75
Baseline+PA (Random 160K)	94.29	86.58
Baseline+PA (Selected)	**95.00**	**88.28**

Table 5: The performance of data selection on the development set of the computer domain.

5.2 Free Natural Annotation

When extracting word boundaries from Wikipedia sentences, we ignore natural annotations on English words and digits because these words are recognized by the preprocessor. Following Jiang et al. (2013), we also recognize a naturally annotated two-character subsequence as a word.

5.2.1 Effect of data selection

To make better use of more domain-specific data, and to alleviate noise in partial annotation, we apply the selection method proposed in Section 2 to the Wikipedia data. On the computer domain development test data, this selection method results in 9.4K computer-related sentences with partial annotation. A model is trained with both the PD training data and the partially annotated computer domain Wikipedia data. For comparison, we also trained a model with 160K randomly selected Wikipedia sentences. The experimental result is shown in Table 5. The model incorporating selected data achieves better performance compared to the model with randomly sampled data, demonstrating that data selection is helpful to improving

Method	Finance		Medicine		Literature		Computer		
	F	Roov	F	Roov	F	Roov	F	Roov	Avg-F
Baseline	95.20	86.90	91.36	72.90	92.27	73.61	93.16	83.48	93.00
Baseline+PA (Random 160K)	95.16	87.60	92.41	78.13	92.17	75.30	93.91	83.48	93.41
Baseline+PA (Selected)	**95.54** +0.34	**88.53**	92.47 +1.11	**78.28**	92.49 +0.22	**76.84**	**93.93** +0.77	**87.53**	**93.61**
Jiang et al. (2013)	93.16		**93.34**		**93.53**		91.19		92.80

Table 6: Experimental results on the SIGHAN Bakeoff 2010 data.

the domain adaption accuracy.

5.2.2 Final Result

The final results on the four test domains are shown in Table 6. From this table, we can see that significant improvements are achieved with the help of the partially annotated Wikipedia data, when compared to the baseline. The models trained with selected partial annotation perform better than those trained with random partial annotation. Our F-scores are competitive to those reported by Jiang et al. (2013). However, since their model is trained on a different source domain, the results are not directly comparable.

5.2.3 Analysis

In this section, we study the effect of Wikipedia on domain adaptation when no data selection is performed, in order to analyze the effect of partially annotated data. We randomly sample 10K, 20K, 40K, 80K and 160K sentences from the 5.45 million Wikipedia sentences, and incorporate them into the training process, respectively. Five models are obtained adding the baseline, and we test their performances on the four test domains. Figure 4 shows the results.

From the figure we can see that for the medicine and computer domains, where the OOV rate is relatively high, the F-score generally increases when more data from Wikipedia are used. The trends of F-score and OOV recall against the volume of Wikipedia data are almost identical. However, for the finance and literature domains, which have low OOV rates, such a relation between data size and accuracy is not witnessed. For the literature domain, even an opposite trends is shown.

We can draw the following conclusions: (1) Natural annotation on Wikipedia data contributes to the recognition of OOV words on domain adaptation; (2) target domains with more OOV words benefit more from Wikipedia data. (3) along with the positive effect on OOV recognition, Wikipedia data can also introduce noise, and hence data selection can be useful.

Method	Med.	Com.
	F	F
Baseline	91.36	93.16
Baseline+PA (Lex)	91.68	93.47
Baseline+PA (Natural)	92.47	93.93
Baseline+PA (Lex+Natural)	**92.63**	**94.07**

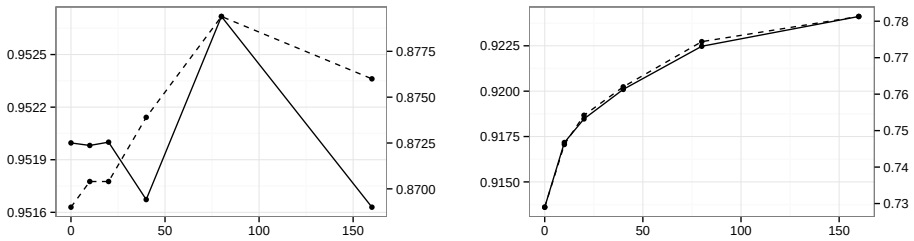
Table 7: Results by combining different sources of free annotation.

5.3 Combining Lexicon and Natural Annotation

To make the most use of free annotation, we combine available free lexicon and natural annotation resources by joining the partially annotated sentences derived using each resource, training our CRF model with these partially annotated sentences and the fully annotated PD sentences. The tests are performed on medicine and computer domains. Table 7 shows the results, where further improvements are made on both domains when the two types of resources are combined.

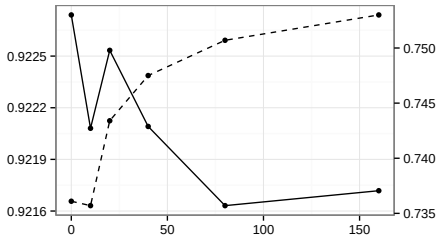
6 Related Work

There has been a line of research on making use of unlabeled data for word segmentation. Zhao and Kit (2008) improve segmentation performance by mutual information between characters, collected from large unlabeled data; Li and Sun (2009) use punctuation information in a large raw corpus to learn a segmentation model, and achieve better recognition of OOV words; Sun and Xu (2011) explore several statistical features derived from unlabeled data to help improve character-based word segmentation. These investigations mainly focus on in-domain accuracies. Liu and Zhang (2012)

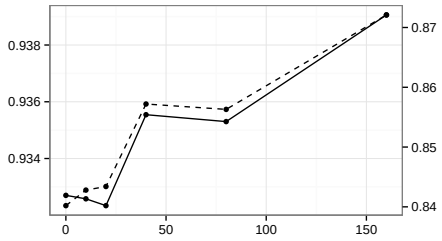


(a) Finance

(b) Medicine



(c) Literature



(d) Computer

Figure 4: Performance of the model incorporating difference sizes of Wikipedia data. The solid line represents the F-score and dashed line represents the recall of OOV words.

study domain adaptation using an unsupervised self-training method. In contrast to their work, we make use of not only unlabeled data, but also leverage any free annotation to achieve better results for domain adaptation.

There has also been work on making use of a dictionary and natural annotation for segmentation. Zhang et al. (2014) study type-supervised domain adaptation for Chinese segmentation. They categorize domain difference into two types: different vocabulary and different POS distributions. While the first type of difference can be effectively resolved by using lexicon for each domain, the second type of difference needs to be resolved by using annotated sentences. They found that given the same manual annotation time, a combination of the lexicon and sentence is the most effective. Jiang et al. (2013) use 160K Wikipedia sentences to improves segmentation accuracies on several domains. Both Zhang et al. (2014) and Jiang et al. (2013) work on discriminative models using the structure perceptron (Collins, 2002), although they study two different sources of information. In contrast to their work, we unify both types of information under the CRF framework.

CRF has been used for Chinese word segmentation (Tseng, 2005; Shi and Wang, 2007; Zhao and Kit, 2008; Wang et al., 2011). However, most previous work train a CRF by using full annotation only. In contrast, we study CRF based segmentation by using both full and partial annotation.

Several other variants of CRF model has been proposed in the machine learning literature, such as the generalized expectation method (Mann and McCallum, 2008), which introduce knowledge by incorporating a manually annotated feature distribution into the regularizer, and the JESS-CM (Suzuki and Isozaki, 2008), which use a EM-like method to iteratively optimize the parameter on both the annotated data and unlabeled data. In contrast, we directly incorporate the likelihood of partial annotation into the objective function. The work that is the most similar to ours is Tsuboi et al. (2008), who modify the CRF learning objective for partial data. They focus on Japanese lexical analysis using manually collected partial data, while we investigate the effect of partial annotation from freely available sources for Chinese segmentation.

7 Conclusion

In this paper, we investigated the problem of domain adaptation for word segmentation, by transferring various sources of free annotations into a consistent form of partially annotated data and applying a variant of CRF that can be trained using fully- and partially-annotated data simultaneously. We performed a large set of experiments to study the effectness of free data, finding that they are useful for improving segmentation accuracy. Experiments also show that proper data selection can further benefit the model's performance.

Acknowledgments

We thank the anonymous reviewers for their constructive comments. This work was supported by the National Key Basic Research Program of China via grant 2014CB340503 and the National Natural Science Foundation of China (NSFC) via grant 61133012 and 61370164, the Singapore Ministry of Education (MOE) AcRF Tier 2 grant T2MOE201301 and SRG ISTD 2012 038 from Singapore University of Technology and Design.

References

Leonard E Baum and Ted Petrie. 1966. Statistical inference for probabilistic functions of finite state markov chains. *The annals of mathematical statistics*, pages 1554–1563.

Léon Bottou. 1991. Stochastic gradient learning in neural networks. In *Proceedings of Neuro-Nîmes 91*, Nimes, France. EC2.

Michael Collins. 2002. Discriminative training methods for hidden markov models: Theory and experiments with perceptron algorithms. In *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing*, pages 1–8. Association for Computational Linguistics, July.

Jianfeng Gao, Mu Li, Andi Wu, and Chang-Ning Huang. 2005. Chinese word segmentation and named entity recognition: A pragmatic approach. *Comput. Linguist.*, 31(4):531–574, December.

Dan Garrette and Jason Baldridge. 2012. Type-supervised hidden markov models for part-of-speech tagging with incomplete tag dictionaries. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 821–831, Jeju Island, Korea, July. Association for Computational Linguistics.

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*, pages 522–530, Suntec, Singapore, August. Association for Computational Linguistics.

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 the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 761–769, Sofia, Bulgaria, August. Association for Computational Linguistics.

John D. Lafferty, Andrew McCallum, and Fernando C. N. Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the Eighteenth International Conference on Machine Learning*, ICML '01, pages 282–289, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Zhongguo Li and Maosong Sun. 2009. Punctuation as implicit annotations for chinese word segmentation. *Comput. Linguist.*, 35(4):505–512, December.

D. C. Liu and J. Nocedal. 1989. On the limited memory bfgs method for large scale optimization. *Math. Program.*, 45(3):503–528, December.

Yang Liu and Yue Zhang. 2012. Unsupervised domain adaptation for joint segmentation and POS-tagging. In *Proceedings of COLING 2012: Posters*, pages 745–754, Mumbai, India, December. The COLING 2012 Organizing Committee.

Gideon S. Mann and Andrew McCallum. 2008. Generalized expectation criteria for semi-supervised learning of conditional random fields. In *Proceedings of ACL-08: HLT*, pages 870–878, Columbus, Ohio, June. Association for Computational Linguistics.

Yanxin Shi and Mengqiu Wang. 2007. A dual-layer crfs based joint decoding method for cascaded segmentation and labeling tasks. In *Proceedings of the 20th International Joint Conference on Artifical Intelligence*, IJCAI'07, pages 1707–1712, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Yan Song, Prescott Klassen, Fei Xia, and Chunyu Kit. 2012. Entropy-based training data selection for domain adaptation. In *Proceedings of COLING 2012: Posters*, pages 1191–1200, Mumbai, India, December. The COLING 2012 Organizing Committee.

Weiwei Sun and Jia Xu. 2011. Enhancing chinese word segmentation using unlabeled data. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 970–979, Edinburgh, Scotland, UK., July. Association for Computational Linguistics.

Jun Suzuki and Hideki Isozaki. 2008. Semi-supervised sequential labeling and segmentation using gigaword scale unlabeled data. In *Proceedings of ACL-08: HLT*, pages 665–673, Columbus, Ohio, June. Association for Computational Linguistics.

Huihsin Tseng. 2005. A conditional random field word segmenter. In *In Fourth SIGHAN Workshop on Chinese Language Processing*.

Yuta Tsuboi, Hisashi Kashima, Shinsuke Mori, Hiroki Oda, and Yuji Matsumoto. 2008. Training conditional random fields using incomplete annotations. In *Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008)*, pages 897–904, Manchester, UK, August. Coling 2008 Organizing Committee.

Yiou Wang, Jun'ichi Kazama, Yoshimasa Tsuruoka, Wenliang Chen, Yujie Zhang, and Kentaro Torisawa. 2011. Improving chinese word segmentation and pos tagging with semi-supervised methods using large auto-analyzed data. In *Proceedings of 5th International Joint Conference on Natural Language Processing*, pages 309–317, Chiang Mai, Thailand, November. Asian Federation of Natural Language Processing.

Pak-kwong Wong and Chorkin Chan. 1996. Chinese word segmentation based on maximum matching and word binding force. In *Proceedings of the 16th Conference on Computational Linguistics - Volume 1*, COLING '96, pages 200–203, Stroudsburg, PA, USA. Association for Computational Linguistics.

Zimin Wu and Gwyneth Tseng. 1993. Chinese text segmentation for text retrieval: Achievements and problems. *J. Am. Soc. Inf. Sci.*, 44(9):532–542, October.

Nianwen Xue and Libin Shen. 2003. Chinese word segmentation as lmr tagging. In *Proceedings of the Second SIGHAN Workshop on Chinese Language Processing - Volume 17*, SIGHAN '03, pages 176–179, Stroudsburg, PA, USA. Association for Computational Linguistics.

Shiwen Yu, Jianming Lu, Xuefeng Zhu, Huiming Duan, Shiyong Kang, Honglin Sun, Hui Wang, Qiang Zhao, and Weidong Zhan. 2001. Processing norms of modern chinese corpus. Technical report.

Yue Zhang and Stephen Clark. 2007. Chinese segmentation with a word-based perceptron algorithm. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 840–847, Prague, Czech Republic, June. Association for Computational Linguistics.

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*, pages 588–597, Gothenburg, Sweden, April. Association for Computational Linguistics.

Hai Zhao and Chunyu Kit. 2008. An empirical comparison of goodness measures for unsupervised chinese word segmentation with a unified framework. In *In: The Third International Joint Conference on Natural Language Processing (IJCNLP-2008*.

Hai Zhao, Chang-Ning Huang, Mu Li, and Bao-Liang Lu. 2010. A unified character-based tagging framework for chinese word segmentation. 9(2):5:1–5:32, June.