Discriminative Learning with Natural Annotations: Word Segmentation as a Case Study

Wenbin Jiang [1] **Meng Sun** [1] **Yajuan Lü** [1] **Yating Yang** [2] **Qun Liu** [3, 1]

[1]Key Laboratory of Intelligent Information Processing
Institute of Computing Technology, Chinese Academy of Sciences
{jiangwenbin, sunmeng, lvyajuan}@ict.ac.cn
[2]Multilingual Information Technology Research Center
The Xinjiang Technical Institute of Physics & Chemistry, Chinese Academy of Sciences
yangyt@ms.xjb.ac.cn
[3]Centre for Next Generation Localisation
Faculty of Engineering and Computing, Dublin City University
qliu@computing.dcu.ie

Abstract

Structural information in web text provides natural annotations for NLP problems such as word segmentation and parsing. In this paper we propose a discriminative learning algorithm to take advantage of the linguistic knowledge in large amounts of natural annotations on the Internet. It utilizes the Internet as an external corpus with massive (although slight and sparse) natural annotations, and enables a classifier to evolve on the large-scaled and real-time updated web text. With Chinese word segmentation as a case study, experiments show that the segmenter enhanced with the Chinese wikipedia achieves significant improvement on a series of testing sets from different domains, even with a single classifier and local features.

1 Introduction

Problems related to information retrieval, machine translation and social computing need fast and accurate text processing, for example, word segmentation and parsing. Taking Chinese word segmentation for example, the state-of-the-art models (Xue and Shen, 2003; Ng and Low, 2004; Gao et al., 2005; Nakagawa and Uchimoto, 2007; Zhao and Kit, 2008; Jiang et al., 2009; Zhang and Clark, 2010; Sun, 2011b; Li, 2011) are usually trained on human-annotated corpora such as the Penn Chinese Treebank (CTB) (Xue et al., 2005), and perform quite well on corresponding test sets. Since the text used for corpus annotating are usually drawn from specific fields (e.g. newswire or finance), and the annotated corpora are limited in size (e.g. tens of thousands), the performance of word segmentation tends to degrade sharply when applied to new domains.

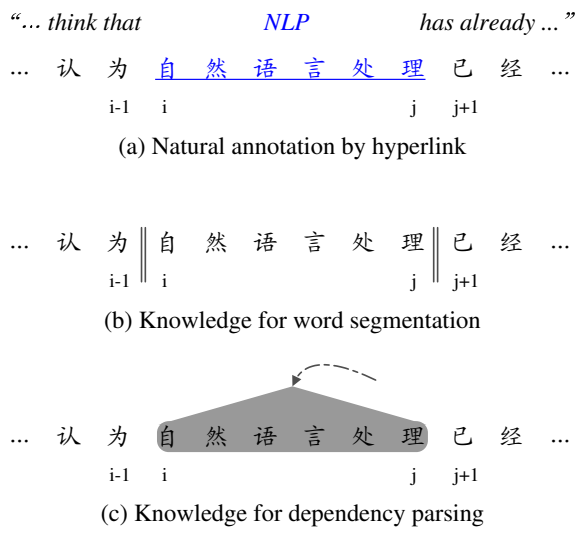


Figure 1: Natural annotations for word segmentation and dependency parsing.

Internet provides large amounts of raw text, and statistics collected from it have been used to improve parsing performance (Nakov and Hearst, 2005; Pitler et al., 2010; Bansal and Klein, 2011; Zhou et al., 2011). The Internet also gives massive (although slight and sparse) natural annotations in the forms of structural information including hyperlinks, fonts, colors and layouts (Sun, 2011a). These annotations usually imply valuable knowledge for problems such as word segmentation and parsing, based on the hypothesis that the subsequences marked by structural information are meaningful fragments in sentences. Figure 1 shows an example. The hyperlink indicates

Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics, pages 761–769,
Sofia, Bulgaria, August 4-9 2013.

a Chinese phrase (meaning NLP), and it probably corresponds to a connected sub-graph for dependency parsing. Creators of web text give valuable annotations during editing, the whole Internet can be treated as a wide-coveraged and real-time updated corpus.

Different from the dense and accurate annotations in human-annotated corpora, natural annotations in web text are sparse and slight, it makes direct training of NLP models impracticable. In this work we take for example a most important problem, word segmentation, and propose a novel discriminative learning algorithm to leverage the knowledge in massive natural annotations of web text. Character classification models for word segmentation usually factorize the whole prediction into atomic predictions on characters (Xue and Shen, 2003; Ng and Low, 2004). Natural annotations in web text can be used to get rid of implausible predication candidates for related characters, knowledge in the natural annotations is therefore introduced in the manner of searching space pruning. Since constraint decoding in the pruned searching space integrates the knowledge of the baseline model and natural annotations, it gives predictions not worse than the normal decoding does. Annotation differences between the outputs of constraint decoding and normal decoding are used to train the enhanced classifier. This strategy makes the usage of natural annotations simple and universal, which facilitates the utilization of massive web text and the extension to other NLP problems.

Although there are lots of choices, we choose the Chinese wikipedia as the knowledge source due to its high quality. Structural information, including hyperlinks, fonts and colors are used to determine the boundaries of meaningful fragments. Experimental results show that, the knowledge implied in the natural annotations can significantly improve the performance of a baseline segmenter trained on CTB 5.0, an F-measure increment of 0.93 points on CTB test set, and an average increment of 1.53 points on 7 other domains. It is an effective and inexpensive strategy to build word segmenters adaptive to different domains. We hope to extend this strategy to other NLP problems such as named entity recognition and parsing.

In the rest of the paper, we first briefly introduce the problems of Chinese word segmentation and the character classification model in section 2, then describe the representation of the knowledge in natural annotations of web text in section 3, and finally detail the strategy of discriminative learning on natural annotations in section 4. After giving the experimental results and analysis in section 5, we briefly introduce the previous related work and then give the conclusion and the expectation of future research.

Type	Templates	Instances
n-gram	C_{-2}	C_{-2}=认
	C_{-1}	C_{-1}=为
	C_0	C_0=自
	C_1	C_1=然
	C_2	C_2=语
	$C_{-2}C_{-1}$	$C_{-2}C_{-1}$=认为
	$C_{-1}C_0$	$C_{-1}C_0$=为自
	C_0C_1	C_0C_1=自然
	C_1C_2	C_1C_2=然语
	$C_{-1}C_1$	$C_{-1}C_1$=为然
function	$Pu(C_0)$	$Pu(C_0)$=false
	$T(C_{-2:2})$	$T(C_{-2:2})$= 44444

Table 1: Feature templates and instances for character classification-based word segmentation model. Suppose we are considering the i-th character "自" in "...认为 自 然语言处理已经...".

2 Character Classification Model

Character classification models for word segmentation factorize the whole prediction into atomic predictions on single characters (Xue and Shen, 2003; Ng and Low, 2004). Although natural annotations in web text do not directly support the discriminative training of segmentation models, they do get rid of the implausible candidates for predictions of related characters.

Given a sentence as a sequence of n characters, word segmentation splits the sequence into $m(\leq n)$ subsequences, each of which indicates a meaningful word. Word segmentation can be formalized as a character classification problem (Xue and Shen, 2003), where each character in the sentence is given a boundary tag representing its position in a word. We adopt the boundary tags of Ng and Low (2004), b, m, e and s, where b, m and e mean the beginning, the middle and the end of a word, and s indicates a single-character word. the decoding procedure searches for the labeled character sequence y that maximizes the score func-

Algorithm 1 Perceptron training algorithm.

Input: Training corpus $\mathcal{C}$
$\vec{\alpha} \leftarrow \mathbf{0}$
for $t \leftarrow 1$ .. T **do** ▷ T iterations
  for $(x, \tilde{y}) \in \mathcal{C}$ **do**
    $y \leftarrow \arg\max_y \Phi(x, y) \cdot \vec{\alpha}$
    if $y \neq \tilde{y}$ **then**
      $\vec{\alpha} \leftarrow \vec{\alpha} + \mathbf{\Phi}(x, \tilde{y}) - \Phi(x, y)$
Output: Parameters $\vec{\alpha}$

tion:

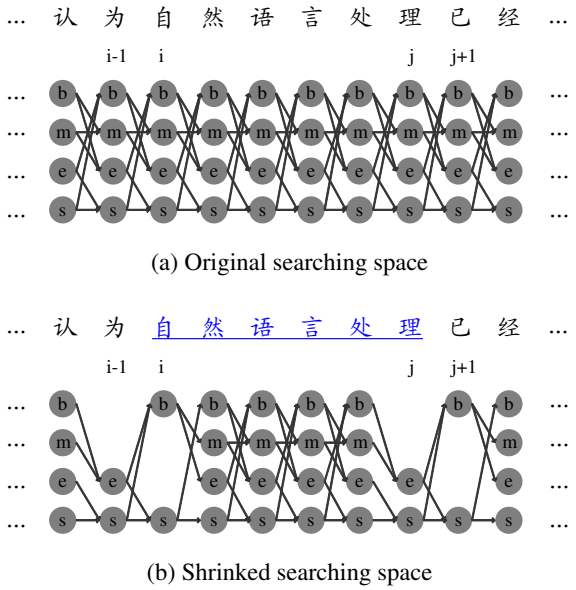
$$\begin{aligned} f(x) &= \arg\max_y S(y|\vec{\alpha}, \Phi, x) \\ &= \arg\max_y \Phi(x, y) \cdot \vec{\alpha} \\ &= \arg\max_y \sum_{(i,t)\in y} \Phi(i, t, x, y) \cdot \vec{\alpha} \end{aligned} \quad (1)$$

The score of the whole sequence y is accumulated across all its character-label pairs, $(i, t) \in y$ (s.t. $1 \leq i \leq n$ and $t \in \{b, m, e, s\}$). The feature function Φ maps a labeled sequence or a character-label pair into a feature vector, $\vec{\alpha}$ is the parameter vector and $\Phi(x, y) \cdot \vec{\alpha}$ is the inner product of $\Phi(x, y)$ and $\vec{\alpha}$.

Analogous to other sequence labeling problems, word segmentation can be solved through a viterbi-style decoding procedure. We omit the decoding algorithm in this paper due to its simplicity and popularity.

The feature templates for the classifier is shown in Table 1. C_0 denotes the current character, while C_{-k}/C_k denote the kth character to the left/right of C_0. The function $Pu(\cdot)$ returns *true* for a punctuation character and *false* for others, the function $T(\cdot)$ classifies a character into four types, 1, 2, 3 and 4, representing *number*, *date*, *English letter* and *others*, respectively.

The classifier can be trained with online learning algorithms such as perceptron, or offline learning models such as support vector machines. We choose the perceptron algorithm (Collins, 2002) to train the classifier for the character classification-based word segmentation model. It learns a discriminative model mapping from the inputs $x \in X$ to the outputs $\tilde{y} \in Y$, where X is the set of sentences in the training corpus and Y is the set of corresponding labeled results. Algorithm 1 shows the perceptron algorithm for tuning the parameter $\vec{\alpha}$. The "averaged parameters" technology (Collins, 2002) is used for better performance.



(a) Original searching space

(b) Shrinked searching space

Figure 2: Shrink of searching space for the character classification-based word segmentation model.

3 Knowledge in Natural Annotations

Web text gives massive natural annotations in the form of structural informations, including hyperlinks, fonts, colors and layouts (Sun, 2011a). Although slight and sparse, these annotations imply valuable knowledge for problems such as word segmentation and parsing.

As shown in Figure 1, the subsequence $P = i..j$ of sentence S is composed of bolded characters determined by a hyperlink. Such natural annotations do not clearly give each character a boundary tag, or define the head-modifier relationship between two words. However, they do help to shrink the set of plausible predication candidates for each character or word. For word segmentation, it implies that characters $i-1$ and j are the rightmost characters of words, while characters i and $j+1$ are the leftmost characters of words. For $i-1$ or j, the plausible predication set Ψ becomes $\{e, s\}$; For i and $j+1$, it becomes $\{b, s\}$; For other characters c except the two at sentence boundaries, $\Psi(c)$ is still $\{b, m, e, s\}$. For dependency parsing, the subsequence P tends to form a connected dependency graph if it contains more than one word. Here we use Ψ to denote the set of plausible head of a word (modifier). There must be a single word $w \in P$ as the root of subsequence P, whose plausible heads fall out of P, that is, $\Psi(w) = \{x|x \in S - P\}$. For the words in P except the root, the plausible heads for each

Algorithm 2 Perceptron learning with natural annotations.

$\vec{\alpha} \leftarrow \text{TRAIN}(\mathcal{C})$
for $x \in \mathcal{F}$ **do**
  $y \leftarrow \text{DECODE}(x, \vec{\alpha})$
  $\tilde{y} \leftarrow \text{CONSTRAINTDECODE}(x, \vec{\alpha}, \Psi)$
  if $y \neq \tilde{y}$ **then**
    $\mathcal{C}' \leftarrow \mathcal{C}' \cup \{\tilde{y}\}$
$\vec{\alpha} \leftarrow \text{TRAIN}(\mathcal{C} \cup \mathcal{C}')$

Algorithm 3 Online version of perceptron learning with natural annotations.

$\vec{\alpha} \leftarrow \text{TRAIN}(\mathcal{C})$
for x with natural annotations **do**
  $y \leftarrow \text{DECODE}(x, \vec{\alpha})$
  $\tilde{y} \leftarrow \text{CONSTRAINTDECODE}(x, \vec{\alpha}, \Psi)$
  if $y \neq \tilde{y}$ **then**
    $\vec{\alpha} \leftarrow \vec{\alpha} + \mathbf{\Phi}(x, \tilde{y}) - \mathbf{\Phi}(x, y)$
  output $\vec{\alpha}$ at regular time

word w are the words in P except w itself, that is, $\Psi(w) = \{x | x \in P - \{w\}\}$.

Creators of web text give valuable structural annotations during editing, these annotations reduce the predication uncertainty for atomic characters or words, although not exactly defining which predication is. Figure 2 shows an example for word segmentation, depicting the shrink of searching space for the character classification-based model. Since the decrement of uncertainty indicates the increment of knowledge, the whole Internet can be treated as a wide-coveraged and real-time updated corpus. We choose the Chinese wikipedia as the external knowledge source, and structural information including hyperlinks, fonts and colors are used in the current work due to their explicitness of representation.

4 Learning with Natural Annotations

Different from the dense and accurate annotations in human-annotated corpora, natural annotations are sparse and slight, which makes direct training of NLP models impracticable. Annotations implied by structural information do not give an exact predication to a character, however, they help to get rid of the implausible predication candidates for related characters, as described in the previous section.

Previous work on constituency parsing or machine translation usually resort to some kinds of heuristic tricks, such as punctuation restrictions, to eliminate some implausible candidates during decoding. Here the natural annotations also bring knowledge in the manner of searching space pruning. Conditioned on the completeness of the decoding algorithm, a model trained on an existing corpus probably gives better or at least not worse predications, by constraint decoding in the pruned searching space. The constraint decoding procedure integrates the knowledge of the baseline model and natural annotations, the predication differences between the outputs of constraint decoding and normal decoding can be used to train the enhanced classifier.

Restrictions of the searching space according to natural annotations can be easily incorporated into the decoder. If the completeness of the searching algorithm can be guaranteed, the constraint decoding in the pruned searching space will give predications not worse than those given by the normal decoding. If a predication of constraint decoding differs from that of normal decoding, it indicates that the annotation precision is higher than the latter. Furthermore, the degree of difference between the two predications represents the amount of new knowledge introduced by the natural annotations over the baseline.

The baseline model $\vec{\alpha}$ is trained on an existing human-annotated corpus. A set of sentences $\mathcal{F}$ with natural annotations are extracted from the Chinese wikipedia, and we reserve the ones for which constraint decoding and normal decoding give different predications. The predictions of reserved sentences by constraint decoding are used as additional training data for the enhanced classifier. The overall training pipeline is analogous to self-training (McClosky et al., 2006), Algorithm 2 shows the pseudo-codes. Considering the *online* characteristic of the perceptron algorithm, if we are able to leverage much more (than the Chinese wikipedia) data with natural annotations, an online version of learning procedure shown in Algorithm 3 would be a better choice. The technology of "averaged parameters" (Collins, 2002) is easily to be adapted here for better performance.

When constraint decoding and normal decoding give different predications, we only know that the former is probably better than the latter. Although there is no explicit evidence for us to measure how much difference in accuracy between the

Partition	Sections	# of word
CTB		
Training	$1-270$	0.47M
	$400-931$	
	$1001-1151$	
Developing	$301-325$	6.66K
Testing	$271-300$	7.82K

Table 2: Data partitioning for CTB 5.0.

two predications, we can approximate how much new knowledge that a naturally annotated sentence brings. For a sentence x, given the predications of constraint decoding and normal decoding, $\tilde{y}$ and y, the difference of their scores $\delta = S(y) - S(\tilde{y})$ indicates the degree to which the current model mistakes. This indicator helps us to select more valuable training examples.

The strategy of learning with natural annotations can be adapted to other situations. For example, if we have a list of words or phrases (especially in a specific domain such as medicine and chemical), we can generate annotated sentences automatically by string matching in a large amount of raw text. It probably provides a simple and effective domain adaptation strategy for already trained models.

5 Experiments

We use the Penn Chinese Treebank 5.0 (CTB) (Xue et al., 2005) as the existing annotated corpus for Chinese word segmentation. For convenient of comparison with other work in word segmentation, the whole corpus is split into three partitions as follows: chapters 271-300 for testing, chapters 301-325 for developing, and others for training. We choose the Chinese wikipedia [1] (version 20120812) as the external knowledge source, because it has high quality in contents and it is much better than usual web text. Structural informations, including hyperlinks, fonts and colors are used to derive the annotation information.

To further evaluate the improvement brought by the fuzzy knowledge in Chinese wikipedia, a series of testing sets from different domains are adopted. The four testing sets from SIGHAN Bakeoff 2010 (Zhao and Liu, 2010) are used, they are drawn from the domains of literature, finance, computer science and medicine. Although the reference sets are annotated according to a different

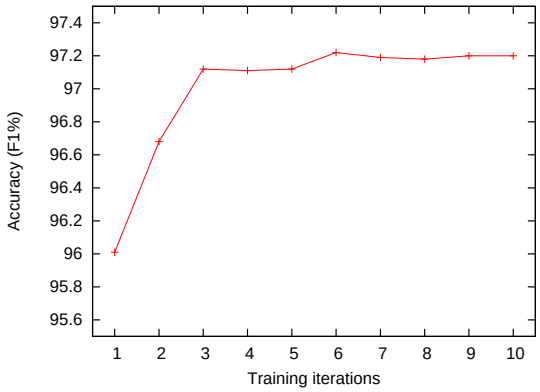


Figure 3: Learning curve of the averaged perceptron classifier on the CTB developing set.

word segmentation standard (Yu et al., 2001), the quantity of accuracy improvement is still illustrative since there are no vast diversities between the two segmentation standards. We also annotated another three testing sets [2], their texts are drawn from the domains of chemistry, physics and machinery, and each contains 500 sentences.

5.1 Baseline Classifier for Word Segmentation

We train the baseline perceptron classifier for word segmentation on the training set of CTB 5.0, using the developing set to determine the best training iterations. The performance measurement for word segmentation is balanced F-measure, $F = 2PR/(P+R)$, a function of precision P and recall R, where P is the percentage of words in segmentation results that are segmented correctly, and R is the percentage of correctly segmented words in the gold standard words.

Figure 3 shows the learning curve of the averaged perceptron on the developing set. The second column of Table 3 lists the performance of the baseline classifier on eight testing sets, where newswire denotes the testing set of the CTB itself. The classifier performs much worse on the domains of chemistry, physics and machinery, it indicates the importance of domain adaptation for word segmentation (Gao et al., 2004; Ma and Way, 2009; Gao et al., 2010). The accuracy on the testing sets from SIGHAN Bakeoff 2010 is even lower due to the difference in both domains and word segmentation standards.

[1] http://download.wikimedia.org/backup-index.html.

[2] They are available at http://nlp.ict.ac.cn/ jiangwenbin/.

Dataset	Baseline (F%)	Enhanced (F%)	
Newswire	97.35	98.28	+0.93
Out-of-Domain			
Chemistry	93.61	95.68	+2.07
Physics	95.10	97.24	+2.14
Machinery	96.08	97.66	+1.58
Literature	92.42	93.53	+1.11
Finance	92.50	93.16	+0.66
Computer	89.46	91.19	+1.73
Medicine	91.88	93.34	+1.46
Average	93.01	94.54	+1.53

Table 3: Performance of the baseline classifier and the classifier enhanced with natural annotations in Chinese wikipedia.

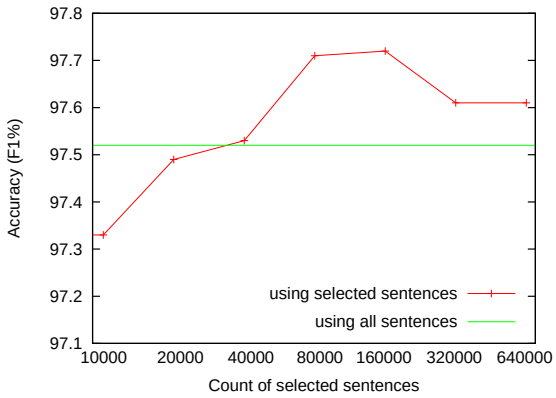


Figure 4: Performance curve of the classifier enhanced with selected sentences of different scales.

Model	Accuracy ($F\%$)
(Jiang et al., 2008)	97.85
(Kruengkrai et al., 2009)	97.87
(Zhang and Clark, 2010)	97.79
(Wang et al., 2011)	98.11
(Sun, 2011b)	98.17
Our Work	98.28

Table 4: Comparison with state-of-the-art work in Chinese word segmentation.

5.2 Classifier Enhanced with Natural Annotations

The Chinese wikipedia contains about 0.5 million items. From their description text, about 3.9 millions of sentences with natural annotations are extracted. With the CTB training set as the existing corpus $\mathcal{C}$, about 0.8 million sentences are reserved according to Algorithm 2, the segmentations given by constraint decoding are used as additional training data for the enhanced classifier.

According to the previous description, the difference of the scores of constraint decoding and normal decoding, $\delta = S(y) - S(\tilde{y})$, indicates the importance of a constraint segmentation to the improvement of the baseline classifier. The constraint segmentations of the reserved sentences are sorted in descending order according to the difference of the scores of constraint decoding and normal decoding, as described previously. From the beginning of the sorted list, different amounts of segmented sentences are used as the additional training data for the enhanced character classifier. Figure 4 shows the performance curve of the enhanced classifiers on the developing set of CTB. We found that the highest accuracy was achieved when $160,000$ sentences were used, while more additional training data did not give continuous improvement. A recent related work about self-training for segmentation (Liu and Zhang, 2012) also reported a very similar trend, that only a moderate amount of raw data gave the most obvious improvements.

The performance of the enhanced classifier is listed in the third column of Table 3. On the CTB testing set, training data from the Chinese wikipedia brings an F-measure increment of 0.93 points. On out-of-domain testing sets, the improvements are much larger, an average increment of 1.53 points is achieved on seven domains. It is probably because the distribution of the knowledge in the CTB training data is concentrated in the domain of newswire, while the contents of the Chinese wikipedia cover a broad range of domains, it provides knowledge complementary to that of CTB.

Table 4 shows the comparison with other work in Chinese word segmentation. Our model achieves an accuracy higher than that of the state-of-the-art models trained on CTB only, although using a single classifier with only local features. From the viewpoint of resource utilization, the comparison between our system and previous work without using additional training data is unfair. However, we believe this work shows another interesting way to improve Chinese word segmentation, it focuses on the utilization of fuzzy and sparse knowledge on the Internet rather than making full use of a specific human-annotated corpus. On the other hand, since only a single classifier and local features are used in our method, better performance could be achieved

resorting to complicated features, system combination and other semi-supervised technologies. What is more, since the text on Internet is wide-coveraged and real-time updated, our strategy also helps a word segmenter be more domain adaptive and up to date.

6 Related Work

Li and Sun (2009) extracted character classification instances from raw text for Chinese word segmentation, resorting to the indication of punctuation marks between characters. Sun and Xu (Sun and Xu, 2011) utilized the features derived from large-scaled unlabeled text to improve Chinese word segmentation. Although the two work also made use of large-scaled raw text, our method is essentially different from theirs in the aspects of both the source of knowledge and the learning strategy.

Lots of efforts have been devoted to semi-supervised methods in sequence labeling and word segmentation (Xu et al., 2008; Suzuki and Isozaki, 2008; Haffari and Sarkar, 2008; Tomanek and Hahn, 2009; Wang et al., 2011). A semi-supervised method tries to find an optimal hyperplane of both annotated data and raw data, thus to result in a model with better coverage and higher accuracy. Researchers have also investigated unsupervised methods in word segmentation (Zhao and Kit, 2008; Johnson and Goldwater, 2009; Mochihashi et al., 2009; Hewlett and Cohen, 2011). An unsupervised method mines the latent distribution regularity in the raw text, and automatically induces word segmentation knowledge from it. Our method also needs large amounts of external data, but it aims to leverage the knowledge in the fuzzy and sparse annotations. It is fundamentally different from semi-supervised and unsupervised methods in that we aimed to excavate a totally different kind of knowledge, the natural annotations implied by the structural information in web text.

In recent years, much work has been devoted to the improvement of word segmentation in a variety of ways. Typical approaches include the introduction of global training or complicated features (Zhang and Clark, 2007; Zhang and Clark, 2010), the investigation of word internal structures (Zhao, 2009; Li, 2011), the adjustment or adaptation of word segmentation standards (Wu, 2003; Gao et al., 2004; Jiang et al., 2009), the integrated solution of segmentation and related tasks such as part-of-speech tagging and parsing (Zhou and Su, 2003; Zhang et al., 2003; Fung et al., 2004; Goldberg and Tsarfaty, 2008), and the strategies of hybrid or stacked modeling (Nakagawa and Uchimoto, 2007; Kruengkrai et al., 2009; Wang et al., 2010; Sun, 2011b).

In parsing, Pereira and Schabes (1992) proposed an extended inside-outside algorithm that infers the parameters of a stochastic CFG from a partially parsed treebank. It uses partial bracketing information to improve parsing performance, but it is specific to constituency parsing, and its computational complexity makes it impractical for massive natural annotations in web text. There are also work making use of word co-occurrence statistics collected in raw text or Internet n-grams to improve parsing performance (Nakov and Hearst, 2005; Pitler et al., 2010; Zhou et al., 2011; Bansal and Klein, 2011). When enriching the related work during writing, we found a work on dependency parsing (Spitkovsky et al., 2010) who utilized parsing constraints derived from hypertext annotations to improve the unsupervised dependency grammar induction. Compared with their method, the strategy we proposed is formal and universal, the discriminative learning strategy and the quantitative measurement of fuzzy knowledge enable more effective utilization of the natural annotation on the Internet when adapted to parsing.

7 Conclusion and Future Work

This work presents a novel discriminative learning algorithm to utilize the knowledge in the massive natural annotations on the Internet. Natural annotations implied by structural information are used to decrease the searching space of the classifier, then the constraint decoding in the pruned searching space gives predictions not worse than the normal decoding does. Annotation differences between the outputs of constraint decoding and normal decoding are used to train the enhanced classifier, linguistic knowledge in the human-annotated corpus and the natural annotations of web text are thus integrated together. Experiments on Chinese word segmentation show that, the enhanced word segmenter achieves significant improvement on testing sets of different domains, although using a single classifier with only local features.

Since the contents of web text cover a broad range of domains, it provides knowledge comple-

mentary to that of human-annotated corpora with concentrated distribution of domains. The content on the Internet is large-scaled and real-time updated, it compensates for the drawback of expensive building and updating of corpora. Our strategy, therefore, enables us to build a classifier more domain adaptive and up to date. In the future, we will compare this method with self-training to better illustrate the importance of boundary information, and give error analysis on what types of errors are reduced by the method to make this investigation more complete. We will also investigate more efficient algorithms to leverage more massive web text with natural annotations, and further extend the strategy to other NLP problems such as named entity recognition and parsing.

Acknowledgments

The authors were supported by National Natural Science Foundation of China (Contracts 61202216), 863 State Key Project (No. 2011AA01A207), and National Key Technology R&D Program (No. 2012BAH39B03). Qun Liu's work was partially supported by Science Foundation Ireland (Grant No.07/CE/I1142) as part of the CNGL at Dublin City University. Sincere thanks to the three anonymous reviewers for their thorough reviewing and valuable suggestions!

References

Mohit Bansal and Dan Klein. 2011. Web-scale features for full-scale parsing. In *Proceedings of ACL*.

Michael Collins. 2002. Discriminative training methods for hidden markov models: Theory and experiments with perceptron algorithms. In *Proceedings of EMNLP*, pages 1–8, Philadelphia, USA.

Pascale Fung, Grace Ngai, Yongsheng Yang, and Benfeng Chen. 2004. A maximum-entropy chinese parser augmented by transformation-based learning. In *Proceedings of TALIP*.

Jianfeng Gao, Andi Wu, Mu Li, Chang-Ning Huang, Hongqiao Li, Xinsong Xia, and Haowei Qin. 2004. Adaptive chinese word segmentation. In *Proceedings of ACL*.

Jianfeng Gao, Mu Li, Andi Wu, and Chang-Ning Huang. 2005. Chinese word segmentation and named entity recognition: A pragmatic approach. *Computational Linguistics*.

Wenjun Gao, Xipeng Qiu, and Xuanjing Huang. 2010. Adaptive chinese word segmentation with online passive-aggressive algorithm. In *Proceedings of CIPS-SIGHAN Workshop*.

Yoav Goldberg and Reut Tsarfaty. 2008. A single generative model for joint morphological segmentation and syntactic parsing. In *Proceedings of ACL-HLT*.

Gholamreza Haffari and Anoop Sarkar. 2008. Homotopy-based semi-supervised hidden markov models for sequence labeling. In *Proceedings of COLING*.

Daniel Hewlett and Paul Cohen. 2011. Fully unsupervised word segmentation with bve and mdl. In *Proceedings of ACL*.

Wenbin Jiang, Liang Huang, Yajuan Lv, and Qun Liu. 2008. A cascaded linear model for joint chinese word segmentation and part-of-speech tagging. In *Proceedings of ACL*.

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 47th ACL*.

Mark Johnson and Sharon Goldwater. 2009. Improving nonparameteric bayesian inference: experiments on unsupervised word segmentation with adaptor grammars. In *Proceedings of NAACL*.

Canasai Kruengkrai, Kiyotaka Uchimoto, Jun' ichi Kazama, Yiou Wang, Kentaro Torisawa, and Hitoshi Isahara. 2009. An error-driven word-character hybrid model for joint chinese word segmentation and pos tagging. In *Proceedings of ACL-IJCNLP*.

Zhongguo Li and Maosong Sun. 2009. Punctuation as implicit annotations for chinese word segmentation. *Computational Linguistics*.

Zhongguo Li. 2011. Parsing the internal structure of words: A new paradigm for chinese word segmentation. In *Proceedings of ACL*.

Yang Liu and Yue Zhang. 2012. Unsupervised domain adaptation for joint segmentation and pos-tagging. In *Proceedings of COLING*.

Yanjun Ma and Andy Way. 2009. Bilingually motivated domain-adapted word segmentation for statistical machine translation. In *Proceedings of EACL*.

David McClosky, Eugene Charniak, and Mark Johnson. 2006. Effective self-training for parsing. In *Proceedings of the HLT-NAACL*.

Daichi Mochihashi, Takeshi Yamada, and Naonori Ueda. 2009. Bayesian unsupervised word segmentation with nested pitman-yor language modeling. In *Proceedings of ACL-IJCNLP*.

Tetsuji Nakagawa and Kiyotaka Uchimoto. 2007. A hybrid approach to word segmentation and pos tagging. In *Proceedings of ACL*.

Preslav Nakov and Marti Hearst. 2005. Using the web as an implicit training set: Application to structural ambiguity resolution. In *Proceedings of HLT-EMNLP*.

Hwee Tou Ng and Jin Kiat Low. 2004. Chinese part-of-speech tagging: One-at-a-time or all-at-once? word-based or character-based? In *Proceedings of EMNLP*.

Fernando Pereira and Yves Schabes. 1992. Inside-outside reestimation from partially bracketed corpora. In *Proceedings of ACL*.

Emily Pitler, Shane Bergsma, Dekang Lin, and Kenneth Church. 2010. Using web-scale n-grams to improve base np parsing performance. In *Proceedings of COLING*.

Valentin I. Spitkovsky, Daniel Jurafsky, and Hiyan Alshawi. 2010. Profiting from mark-up: Hyper-text annotations for guided parsing. In *Proceedings of ACL*.

Weiwei Sun and Jia Xu. 2011. Enhancing chinese word segmentation using unlabeled data. In *Proceedings of EMNLP*.

Maosong Sun. 2011a. Natural language processing based on naturally annotated web resources. *CHINESE INFORMATION PROCESSING*.

Weiwei Sun. 2011b. A stacked sub-word model for joint chinese word segmentation and part-of-speech tagging. In *Proceedings of ACL*.

Jun Suzuki and Hideki Isozaki. 2008. Semi-supervised sequential labeling and segmentation using gigaword scale unlabeled data. In *Proceedings of ACL*.

Katrin Tomanek and Udo Hahn. 2009. Semi-supervised active learning for sequence labeling. In *Proceedings of ACL*.

Kun Wang, Chengqing Zong, and Keh-Yih Su. 2010. A character-based joint model for chinese word segmentation. In *Proceedings of COLING*.

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 IJCNLP*.

Andi Wu. 2003. Customizable segmentation of morphologically derived words in chinese. *Computational Linguistics and Chinese Language Processing*.

Jia Xu, Jianfeng Gao, Kristina Toutanova, and Hermann Ney. 2008. Bayesian semi-supervised chinese word segmentation for statistical machine translation. In *Proceedings of COLING*.

Nianwen Xue and Libin Shen. 2003. Chinese word segmentation as lmr tagging. In *Proceedings of SIGHAN Workshop*.

Nianwen Xue, Fei Xia, Fu-Dong Chiou, and Martha Palmer. 2005. The penn chinese treebank: Phrase structure annotation of a large corpus. In *Natural Language Engineering*.

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 ACL 2007*.

Yue Zhang and Stephen Clark. 2010. A fast decoder for joint word segmentation and pos-tagging using a single discriminative model. In *Proceedings of EMNLP*.

Huaping Zhang, Hongkui Yu, Deyi Xiong, and Qun Liu. 2003. Hhmm-based chinese lexical analyzer ictclas. In *Proceedings of SIGHAN Workshop*.

Hai Zhao and Chunyu Kit. 2008. Unsupervised segmentation helps supervised learning of character tagging for word segmentation and named entity recognition. In *Proceedings of SIGHAN Workshop*.

Hongmei Zhao and Qun Liu. 2010. The cips-sighan clp 2010 chinese word segmentation bakeoff. In *Proceedings of CIPS-SIGHAN Workshop*.

Hai Zhao. 2009. Character-level dependencies in chinese: Usefulness and learning. In *Proceedings of EACL*.

Guodong Zhou and Jian Su. 2003. A chinese efficient analyser integrating word segmentation, part-ofspeech tagging, partial parsing and full parsing. In *Proceedings of SIGHAN Workshop*.

Guangyou Zhou, Jun Zhao, Kang Liu, and Li Cai. 2011. Exploiting web-derived selectional preference to improve statistical dependency parsing. In *Proceedings of ACL*.