Punctuation as Implicit Annotations for Chinese Word Segmentation

Zhongguo Li*
Tsinghua University

Maosong Sun**
Tsinghua University

We present a Chinese word segmentation model learned from punctuation marks which are perfect word delimiters. The learning is aided by a manually segmented corpus. Our method is considerably more effective than previous methods in unknown word recognition. This is a step toward addressing one of the toughest problems in Chinese word segmentation.

1. Introduction

Paragraphs are composed of sentences. Hence when a paragraph begins, a sentence must begin, and as a paragraph closes, some sentence must finish. This observation is the basis of the sentence boundary detection method proposed by Riley (1989). Similarly, sentences consist of words. As a sentence begins or ends there must be word boundaries.

Inspired by this notion, we invent a method to learn a Chinese word segmentation model with punctuation marks in a large raw corpus. The learning is guided by a segmented corpus (Section 3.2). Section 4 demonstrates that our method improves notably the recognition of out-of-vocabulary (OOV) words with respect to approaches which use only annotated data (Xue 2003; Low, Ng, and Guo 2005). This work has practical implications in that the OOV problem has long been a big challenge for the research community.

2. Segmentation as Tagging

We call the first character of a Chinese word its **left boundary** L, and the last character its **right boundary** R. If we regard L and R as random events, then we can derive four events (or tags) from them:

$$\mathrm{b} = \mathrm{L} \cdot \overline{\mathrm{R}}, \quad \mathrm{m} = \overline{\mathrm{L}} \cdot \overline{\mathrm{R}}, \quad \mathrm{s} = \mathrm{L} \cdot \mathrm{R}, \quad \mathrm{e} = \overline{\mathrm{L}} \cdot \mathrm{R}$$

* Department of Computer Science and Technology, Tsinghua University, Beijing 100084, China. E-mail: eemath@gmail.com.
** Department of Computer Science and Technology, Tsinghua University, Beijing 100084, China. E-mail: sms@mail.thu.edu.cn.

Submission received: 16 July 2008; revised submission received: 26 March 2009; accepted for publication: 4 May 2009.

Here $\overline{\mathrm{R}}$ means not R, and thus tag b represents the left but not the right boundary of a word. The other tags can be interpreted similarly. This coding scheme was used by Borthwick (1999) and Xue (2003), where b, m, s, and e stand for *begin*, *middle*, *only member*, and *end* of a word, respectively. We reformulate them in terms of L and R to facilitate the presentation of our method.

For a sentence $S = c_1 c_2 \cdots c_n$ and a sequence $T = t_1 t_2 \cdots t_n$ of $\mathrm{b}, \mathrm{m}, \mathrm{s}, \mathrm{e}$ tags, we define

$$\mathcal{P}(T|S) = \prod_{i=1}^{n} \Pr(t_i | context_i) \tag{1}$$

where $context_i$ is c_i with up to four surrounding characters. The legal tag sequence (e.g., tag b followed by s is illegal) with highest $\mathcal{P}$ gives the segmentation result of S. Then from Equation (1) it is obvious that knowing the probability distribution of $\mathrm{b}, \mathrm{m}, \mathrm{s}$, and e given *context* is adequate for carrying out Chinese word segmentation. The purpose of this article is to show that punctuation can play a major role in estimating this distribution.

We use the maximum entropy approach to model the conditional probability $\Pr(y|x)$, which has the following parametric form according to Berger, Della Pietra, and Della Pietra (1996):

$$\Pr(y|x) = \frac{1}{Z(x)} \exp\left(\sum_i \lambda_i f_i(x, y)\right) \tag{2}$$

$$Z(x) = \sum_y \exp\left(\sum_i \lambda_i f_i(x, y)\right) \tag{3}$$

For Chinese word segmentation, the binary valued functions f_i are defined through the 10 features shown in Table 2. Xue (2003) explains how these features map to the feature functions in Equations (2) and (3).

3. Method

Our key idea is to approximate probabilities of $\mathrm{b}, \mathrm{m}, \mathrm{s}$, and e with those of L and R. To do this, we assume L and R are conditionally independent given *context*. Then we have

$$\begin{aligned}
\Pr(\mathrm{b}\,|\mathrm{context}) &= \Pr(\mathrm{L} \cdot \overline{\mathrm{R}}|context) && \text{(definition of b)} \\
&= \Pr(\mathrm{L}\,|context) \cdot \Pr(\overline{\mathrm{R}}|context) && \text{(independence)} \\
&= \Pr(\mathrm{L}\,|context) \cdot (1 - \Pr(\mathrm{R}\,|context)) && (4)
\end{aligned}$$

Probabilities for m, s, and e can be derived in the same way and so their derivations are not provided here. As mentioned earlier, these probabilities are sufficient for Chinese word segmentation. Now to model $\Pr(\mathrm{L}\,|context)$ and $\Pr(\mathrm{R}\,|context)$ with the maximum

entropy technique, we must have positive and negative examples of L and R. It is here that punctuation comes into play.

3.1 Positive Examples

Punctuation offers directly positive examples of L and R. For instance, we can extract four training examples from the sentence in Table 1, as listed in Table 2.

3.2 Negative Examples

Suppose for the moment we know the real probability distribution of tags b, m, s, and e given *context*. Then a character in *context* is itself a word and should be tagged s if

$$\Pr(\mathrm{s}\,|context) > \max_{y\in\{\mathrm{b,m,e}\}} \Pr(y|context) \tag{5}$$

Each positive example given by punctuation is subjected to the test in (5). If an example labeled L passes this test, then it is also a positive example of R because $\mathrm{s} = \mathrm{L}\cdot\mathrm{R}$, and failing this test gives a negative R. In a similar way we obtain negative examples of L. This process is summarized in Figure 1.

A segmented corpus is needed to estimate the probabilities in test (5) with maximum entropy modeling. Here we use the data provided by Microsoft Research in the SIGHAN 2005 Bakeoff. The trained model (the MSR model) was used in earlier work (Low, Ng, and Guo 2005) and is one of the state-of-the-art models for Chinese word segmentation.

With the MSR model, only the last example in Table 2 passes test (5). Hence we get the three negative examples shown in Table 3. Examples like 1, 3, 6, and 8 are used to estimate $\Pr(\mathrm{L}\,|context)$ and those like 2, 4, 5, and 7 are used to estimate $\Pr(\mathrm{R}\,|context)$. Appendix A provides more details on this issue.

Table 1
Illustration of word boundaries near punctuation in a simple sentence.
– means the label is *unknown* with only the help of punctuation.

sentence	阳	光	明	媚	，	椰	树	摇	风	。
word boundary	L	–	–	R		L	–	–	R	

Table 2
Positive training examples extracted from the sentence in Table 1.

		features of context									
No.	label	c_{-2}	c_{-1}	c_0	c_1	c_2	$c_{-1}c_1$	$c_{-2}c_{-1}$	$c_{-1}c_0$	c_0c_1	c_1c_2
1	L			阳	光	明				阳光	光明
2	R	光	明	媚	椰	树	明椰	光明	明媚	媚椰	椰树
3	L	明	媚	椰	树	摇	媚树	明媚	媚椰	椰树	树摇
4	R	树	摇	风				树摇	摇风		

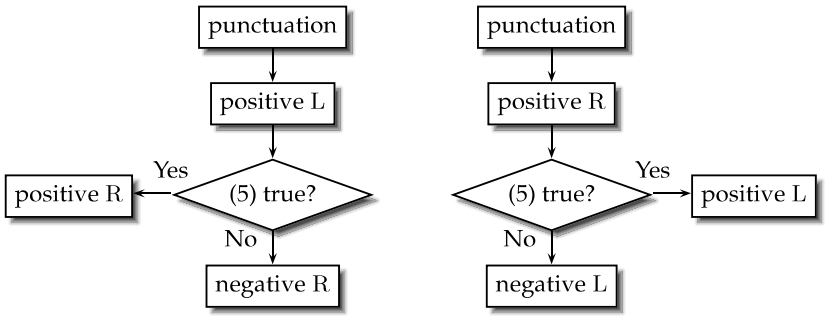

Figure 1
How to get negative examples of L and R. Test (5) is applied to all positive examples given by punctuation. Those failing this test are negative training examples. It is test (5) that invokes the need of a manually segmented corpus.

Table 3
Training examples derived from those in Table 2. We have 1→5, 2→6, 3→7, and 4→8.

		features of context									
No.	label	c_{-2}	c_{-1}	c_0	c_1	c_2	$c_{-1}c_1$	$c_{-2}c_{-1}$	$c_{-1}c_0$	c_0c_1	c_1c_2
5	$\overline{\text{R}}$			阳	光	明				阳光	光明
6	$\overline{\text{L}}$	光	明	媚	椰	树	明椰	光明	明媚	媚椰	椰树
7	$\overline{\text{R}}$	明	媚	椰	树	摇	媚树	明媚	媚椰	椰树	树摇
8	L	树	摇	风				树摇	摇风		

3.3 Training

In all, we collected 10 billion L-$\overline{\text{L}}$ and R-$\overline{\text{R}}$ examples, each from a comprehensive Web corpus.[1] To cope with so much training data, we use the partitioning method of Yamada and Matsumoto (2003). An alternative is the Vowpal Wabbit (fast on-line learning) algorithm.[2] Such an algorithm allows incremental training as more raw texts become available.

4. Evaluation

We evaluate our method with the data and scoring script provided by the SIGHAN 2005 Bakeoff. The data sets of Academia Sinica and City University of Hong Kong, which are in Traditional Chinese, are not used here because the raw corpus is mainly in Simplified Chinese. Table 4 gives the evaluation results on the data from Microsoft Research (MSR) and Peking University (PKU).

It seems our method is over 10% below state of the art in precision on the MSR data. However, we find that multiword expressions are consistently segmented into smaller words. Take the one multiword '中国艺术研究院中国文化研究所' [*Institute of Chinese*

1 Freely available for research purposes. See www.sogou.com/labs.
2 `http://hunch.net/~vw/`. We thank one of the anonymous reviewers for telling us about this implementation.

Culture, Chinese Academy of Arts] in the standard answer of the test data as an example. Our method segments it into six correct words '中国 艺术 研究院 中国 文化 研究所' [*China, art, academy, China, culture, institute*], all of which are considered wrong by the scoring script. This is arguable because the only difference is the granularity of the segmentation.

4.1 Influence of Granularity

We check every error detected by the scoring script on the MSR data, and find that for our method, 15,071 errors are actually correct segmentations of 5,463 multiwords, whereas for the MSR model, the corresponding counts are 858 and 355, respectively. The gold standard contains 106,873 words. These statistics combined with Table 4 allow us to calculate the metrics as in Table 5, if errors caused by correctly segmented multiwords are not counted.

We see that, when the influence of granularity is considered, our method is slightly better than the MSR model. However, as Table 4 shows, both models degrade on the PKU data due to the difference in segmentation standards. This kind of degradation was also documented by Peng, Feng, and McCallum (2004).

4.2 Named Entity List Recovery

The SIGHAN data sets contain relatively few OOV words (2.6% for the MSR data). What if the rate is much higher than that? We expect our model to be less vulnerable to OOV problems because it is trained with billions of examples from a large corpus. To verify this, we generate four data sets from each of these lists of names:

(a) 702 cities and counties of China seen in the MSR data

(b) 1,634 cities and counties of China not seen in the MSR data

(c) 7,470 Chinese personal names seen in the MSR data

(d) 20,000 Chinese personal names not seen in the MSR data

Table 4
Evaluation results on SIGHAN Bakeoff 2005 data sets.

	our method			the MSR model		
data set	P	R	F	P	R	F
MSR	84.8	91.3	87.9	96.0	95.6	95.8
PKU	84.2	86.1	85.1	85.2	82.3	83.7

Table 5
Amended evaluation results for MSR data.

	P	R	F
our method	98.0	96.7	97.3
the MSR model	96.7	96.0	96.3

Table 6
Results on tasks of named entity list recovery.

	our method			the MSR model		
data set	P	R	F	P	R	F
(a)	91.0	93.8	92.4	43.3	29.1	34.8
(b)	79.4	85.3	82.2	25.1	16.9	20.2
(c)	74.9	85.0	79.6	69.4	66.5	67.9
(d)	86.3	91.5	88.8	65.4	61.0	63.1

The generation method is: Randomly permute each list and then put the result into lines, with each line having about 30 names, and repeat this process until we get 1 million tokens for each data set. We use the MSR model and our method to segment these data sets. The results are in Table 6.

It is clear that our method performs better on these data sets. This provides evidence that it could handle situations where many OOV words turn up. Table 6 also indicates that, especially for the MSR model, recognition of Chinese personal names is easier than location names. This is reasonable because the former has more regularity than the latter. Besides, although there are no OOV words in data sets (a) and (c), many words occur very sparsely in the MSR data. Hence the MSR model doesn't do well even on these two data sets.

4.3 Unknown Words Recognition

To further test our model's ability to recognize unknown words, we make 27,470 sentences with the pattern 'X 是 Y 人 ， X 喜欢 Y 。' (X is a resident of Y, and X loves Y), where X and Y are the personal and location names in Section 4.2. The results on this data set are in Table 7. Again our method outperforms the MSR model by a large margin, proving once more that it is stronger in unknown word recognition. For both methods, the metrics in Table 7 are better than those in Table 6, reflecting the fact that unknown word recognition here is easier than the named entity list recovery task.

4.4 Summary

Evaluation shows that when there are many new words, the improvement of our method is obvious. In addition, a model is of limited use if it fits the SIGHAN data well, but can't maintain that accuracy elsewhere. Our model has a wider coverage through

Table 7
Results of unknown word recognition in 24,470 sentences.

	P	R	F
our method	96.2	97.9	97.1
the MSR model	88.3	84.5	86.3

mining the Web. It tends to segment long multiword expressions into their component words. This is not a disadvantage as long as the result is consistent.

5. Related Work

Punctuation gives naturally occurring unambiguous word boundaries. Gao et al. (2005) described how to remove overlapping ambiguities in an annotated corpus to train a model for resolving these ambiguities. A raw corpus doesn't play a role in that method, and the model involves no punctuation marks.

Chinese word segmentation based on position tagging was initiated by Xue (2003). This method and its subsequent developments have achieved state-of-the-art performance in word segmentation (Peng, Feng, and McCallum 2004; Low, Ng, and Guo 2005; Zhao, Huang, and Li 2006). Yet the system degrades when there are lots of previously unknown words, whereas our method performs particular well in this case thanks to the use of a huge Web corpus.

In the past decade, much work has been done in unsupervised word segmentation (Sun, Shen, and Tsou 1998; Peng and Schuurmans 2001; Feng et al. 2004; Goldwater, Griffiths, and Johnson 2006; Jin and Tanaka-Ishii 2006). These methods could also take advantage of the ever-growing amount of online text to model Chinese word segmentation, but usually are less accurate and more complicated than ours.

6. Conclusion

With a virtually unlimited supply of raw corpus data, punctuation marks give us ample training examples and thus can be quite useful as implicit annotations for Chinese word segmentation. We also note that shallow parsing (Sha and Pereira 2003) is a close analogy to word segmentation. Hence our method can potentially be applied to this task as well.

Appendix A: Input to the Training Algorithm

We give readers a feel for the input data used to train our probability models. First, to estimate Pr(L|*context*), the input to the learning algorithm for the maximum entropy models looks like this:

```
+L   C0=阳 C1=光 C2=明 C0C1=阳光 C1C2=光明
+L   C0=椰 C1=树 C2=摇 C0C1=椰树 C1C2=树摇
-L   C-2=光 C-1=明 C0=媚 C-2C-1=光明 C-1C0=明媚
+L   C-2=树 C-1=摇 C0=风 C-2C-1=树摇 C-1C0=摇风
```

Whereas to estimate Pr(R|context), the input data are something like the following

```
+R   C-2=光 C-1=明 C0=媚 C-2C-1=光明 C-1C0=明媚
+R   C-2=树 C-1=摇 C0=风 C-2C-1=树摇 C-1C0=摇风
-R   C0=阳 C1=光 C2=明 C0C1=阳光 C1C2=光明
-R   C0=椰 C1=树 C2=摇 C0C1=椰树 C1C2=树摇
```

To save space, not all features in Table 2 are included here. From this illustration, interested readers can get a general idea of our input to the learning algorithm in Section 3.3.

Acknowledgments

This work is supported by the National Science Foundation of China under Grant No. 60621062 and 60873174, and the National 863 Project under Grant No. 2007AA01Z148. We thank our reviewers sincerely for many helpful comments and suggestions which greatly improved this article. Thanks also go to `sogou.com` for sharing their Web corpora and entity names. The maximum entropy modeling toolkit used here is contributed by Zhang Le of the University of Edinburgh.

References

Berger, Adam L., Vincent J. Della Pietra, and Stephen A. Della Pietra. 1996. A maximum entropy approach to natural language processing. *Computational Linguistics*, 22(1):39–71.

Borthwick, Andrew. 1999. *A Maximum Entropy Approach to Named Entity Recognition*. Ph.D. thesis, New York University.

Feng, Haodi, Kang Chen, Xiaotie Deng, and Weimin Zheng. 2004. Accessor variety criteria for Chinese word extraction. *Computational Linguistics*, 30(1):75–93.

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

Goldwater, Sharon, Thomas L. Griffiths, and Mark Johnson. 2006. Contextual dependencies in unsupervised word segmentation. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 673–680, Sydney.

Jin, Zhihui and Kumiko Tanaka-Ishii. 2006. Unsupervised segmentation of Chinese text by use of branching entropy. In *Proceedings of the COLING/ACL on Main Conference Poster Sessions*, pages 428–435, Morristown, NJ.

Low, Jim Kiat, Hwee Tou Ng, and Wenyuan Guo. 2005. A maximum entropy approach to Chinese word segmentation. In *Proceedings of the Fourth SIGHAN Workshop on Chinese Language Processing*, pages 161–164, Jeju Island.

Peng, Fuchun, Fangfang Feng, and Andrew McCallum. 2004. Chinese segmentation and new word detection using conditional random fields. In *COLING '04: Proceedings of the 20th International Conference on Computational Linguistics*, pages 562–569, Morristown, NJ.

Peng, Fuchun and Dale Schuurmans. 2001. Self-supervised Chinese word segmentation. *Lecture Notes in Computer Science*, 2189:238–249.

Riley, Michael D. 1989. Some applications of tree-based modelling to speech and language. In *HLT '89: Proceedings of the Workshop on Speech and Natural Language*, pages 339–352, Morristown, NJ.

Sha, Fei and Fernando Pereira. 2003. Shallow parsing with conditional random fields. In *NAACL '03: Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology*, pages 134–141, Morristown, NJ.

Sun, Maosong, Dayang Shen, and Benjamin K. Tsou. 1998. Chinese word segmentation without using lexicon and hand-crafted training data. In *Proceedings of the 17th International Conference on Computational Linguistics*, pages 1265–1271, Morristown, NJ.

Xue, Nianwen. 2003. Chinese word segmentation as character tagging. *Computational Linguistics and Chinese Language Processing*, 8(1):29–48.

Yamada, Hiroyasu and Yuji Matsumoto. 2003. Statistical dependency analysis with support vector machines. In *Proceedings of the 8th International Workshop on Parsing Technologies (IWPT2003)*, pages 195–206, Nancy.

Zhao, Hai, Chang-Ning Huang, and Mu Li. 2006. An improved Chinese word segmentation system with conditional random field. In *Proceedings of the Fifth SIGHAN Workshop on Chinese Language Processing*, pages 162–165, Sydney.