A Character-Based Joint Model for CIPS-SIGHAN Word Segmentation Bakeoff 2010

Kun Wang and **Chengqing Zong**
National Laboratory of Pattern Recognition
Institute of Automation, Chinese Academy of Science
{kunwang,cqzong}@nlpr.ia.ac.cn

Keh-Yih Su
Behavior Design Corporation
kysu@bdc.com.tw

Abstract

This paper presents a Chinese Word Segmentation system for the closed track of CIPS-SIGHAN Word Segmentation Bakeoff 2010. This system adopts a character-based joint approach, which combines a character-based generative model and a character-based discriminative model. To further improve the cross-domain performance, we use an additional semi-supervised learning procedure to incorporate the unlabeled corpus. The final performance on the closed track for the simplified-character text shows that our system achieves comparable results with other state-of-the-art systems.

1 Introduction

The character-based tagging approach (Xue, 2003) has become the dominant technique for Chinese word segmentation (CWS) as it can tolerate *out-of-vocabulary* (OOV) words. In the last few years, this method has been widely adopted and further improved in many previous works (Tseng et al., 2005; Zhang et al., 2006; Jiang et al., 2008). Among various character-based tagging approaches, the character-based joint model (Wang et al., 2010) achieves a good balance between *in-vocabulary* (IV) words recognition and OOV words identification.

In this work, we adopt the character-based joint model as our basic system, which combines a character-based discriminative model and a character-based generative model. The generative module holds a robust performance on IV words, while the discriminative module can handle the extra features easily and enhance the OOV words segmentation. However, the performance of out-of-domain text is still not satisfactory as that of in-domain text, while few previous works have paid attention to this problem.

To further improve the performance of the basic system in out-of-domain text, we use a semi-supervised learning procedure to incorporate the unlabeled corpora of Literature (Unlabeled-A) and Computer (Unlabeled-B). The final results show that our system performs well on all four testing-sets and achieves comparable segmentation results with other participants.

2 Our system

2.1 Character-Based Joint Model

The character-based joint model in our system contains two basic components:

- The character-based discriminative model.
- The character-based generative model.

The character-based discriminative model (Xue, 2003) is based on a Maximum Entropy (ME) framework (Ratnaparkhi, 1998) and can be formulated as follows:

$$P(t_1^n \mid c_1^n) \approx \prod_{k=1}^{n} P(t_k \mid t_{k-1}, c_{k-2}^{k+2}) \qquad (1)$$

Where t_k is a member of {***B**egin*, ***M**iddle*, ***E**nd*, ***S**ingle*} (abbreviated as B, M, E and S from now on) to indicate the corresponding position of character c_k in its associated word. For example, the word “北京市 (Beijing City)” will be assigned with the corresponding tags as: “北/B (North) 京/M (Capital) 市/E (City)”.

This discriminative module can flexibly incorporate extra features and it is implemented with the ME package[1] given by Zhang Le. All training experiments are done with Gaussian prior 1.0 and 200 iterations.

The character-based generative module is a character-tag-pair-based trigram model (Wang et al., 2009) and can be expressed as below:

$$P([c,t]_1^n) \approx \prod_{i=1}^{n} P([c,t]_i \left| [c,t]_{i-2}^{i-1} \right.). \qquad (2)$$

In our experiments, SRI Language Modeling Toolkit[2] (Stolcke, 2002) is used to train the generative trigram model with modified Kneser-Ney smoothing (Chen and Goodman, 1998).

The character-based joint model combines the above discriminative module and the generative module with log-linear interpolation as follows:

$$\begin{aligned} Score(t_k) &= \alpha \times \log(P([c,t]_k \left| [c,t]_{k-2}^{k-1} \right.)) \\ &\quad + (1-\alpha) \times \log(P(t_k \left| t_{k-1}, c_{k-2}^{k+2} \right.)) \end{aligned} \qquad (3)$$

Where the parameter $\alpha\ (0.0 \le \alpha \le 1.0)$ is the weight for the generative model. *Score*(t_k) will be directly used during searching the best sequence. We set an empirical value ($\alpha = 0.3$) to this model as there is no development-set for various domains.

2.2 Features

In this work, the feature templates adopted in the character-based discriminative model are very simple and are listed below:

$(a)\ C_n (n = -2,-1,0,12);$

$(b)\ C_n C_{n+1} (n = -2,-1,0,1);$

$(c)\ C_{-1}C_1;$

$(d)\ T(C_{-2})T(C_{-1})T(C_0)T(C_1)T(C_2)$

In the above templates, C_n represents a character and the index *n* indicates the position. For example, when we consider the third character "奥" in the sequence "北京奥运会", template (a) results in the features as following: C_{-2}=北, C_{-1}=京, C_0=奥, C_1=运, C_2=会, and template (b) generates the features as: $C_{-2}C_{-1}$=北京, $C_{-1}C_0$=京奥, C_0C_1=奥运, C_1C_2=运会, and template (c) gives the feature $C_{-1}C_1$=京运.

Template (d) is the feature of character type. Five types classes are defined: dates ("年", "月", "日", the Chinese character for "year", "month" and "day" respectively) represents class 0; foreign alphabets represent class 1; Arabic and Chinese numbers represent class 2; punctuation represents class 3 and other characters represent class 4. For example, when we consider the character "，" in the sequence "八月，阿Q", the feature $T(C_{-2})T(C_{-1})T(C_0)T(C_1)T(C_2)$ will be set to "20341".

When training the character-based discriminative module, we convert all the binary features into real-value features, and set the real-value of C_0 to be 2.0, the value of $C_{-1}C_0$ and C_0C_1 to be 3.0, and the values of all other features to be 1.0. This method sounds a little strange because it is equal to duplicate some features for the maximum entropy training. However, it effectively improves the performance in our previous works.

2.3 Restrictions in constructing lattice

As the closed track allows the participants to use the character type information, we add some restrictions to our system when constructing the character-tag lattice. When we consider a character in the sequence, the type information of both the previous and the next character would be taken into account. The restrictions are list as follows:

- If the previous, the current and the next characters are all English or numbers, we would fix the current tag to be "M";
- If the previous and the next characters are both English or numbers, while the current character is a connective symbol such as "-", "/", "_", "\" etc., we would also fix the current tag to be "M";
- Otherwise, all four tags {B, E, M, S} would be given to the current character.

It is shown that in the Computer domain these simple restrictions not only greatly reduce the number of words segmented, but also speed up the system.

[1] http://homepages.inf.ed.ac.uk/lzhang10/maxent_toolkit.html

[2] http://www.speech.sri.com/projects/srilm/

Domain	Mark	OOV Rate	R	P	F1	R_{OOV}	R_{IV}
Literature	A	0.069	0.937	0.937	0.937	0.652	0.958
Computer	B	0.152	0.941	0.940	0.940	0.757	0.974
Medicine	C	0.110	0.930	0.917	0.923	0.674	0.961
Finance	D	0.087	0.957	0.956	0.957	0.813	0.971

Table 1: Official segmentation results of our system.

```
Algorithm 1: Semi-Supervised Learning
Given:
•  Labeled training corpus: $L$
•  Unlabeled training corpus: $U$

Use $L$ to train a segmenter $S_0$;
Use $S_0$ to segment the unlabeled corpus $U$
and then get labeled corpus $U_0$;
for $i = 1$ to $K$ do
    Add $U_{i-1}$ to $L$ and get a new corpus $L_i$;
    Use $L_i$ to train a new segmenter $S_i$;
    Use $S_i$ to segment the unlabeled corpus
    $U$ and then get labeled corpus $U_i$;
    if convergence criterion meets
        break
end for
Output: the last segmenter $S_K$
```

2.4 Semi-Supervised Learning

In the last decade, Chinese word segmentation has been improved significantly and gets a high precision rate in performance. However, the performance for out-of-domain text is still unsatisfactory at the present. Also, few works have paid attention to the cross-domain problem in Chinese word segmentation task so far.

Self-training and Co-training are two simple semi-supervised learning methods to incorporate unlabeled corpus (Zhu, 2006). In this work, we use an iterative self-training method to incorporate the unlabeled data. A segmenter is first trained with the labeled corpus. Then this segmenter is used to segment the unlabeled data. Then the predicted data is added to the original training corpus as a new training-set. The segmenter will be re-trained and the procedure repeated. To simplify the task, we fix the weight $\alpha = 0.3$ for the generative module of our joint model in the training iterations. The procedure is shown in Algorithm 1. The iterations will not be ended until the similarity of two segmentation results U_{i-1} and U_i reach a certain level. Here we used F-score to measure the similarity between U_{i-1} and U_i: treat U_{i-1} as the benchmark, U_i as a testing-set. From our observation, this method converges quickly in only 3 or 4 iterations for both Literature and Computer corpora.

3 Experiments and Discussion

3.1 Results

In this CIPS-SIGHAN bakeoff, we only participate the closed track for simplified-character text. There are two kinds of training corpora:

- Labeled corpus from News Domain
- Unlabeled corpora from Literature Domain (Unlabeled-A) and Computer Domain (Unlabeled-B).

Also, the testing corpus covers four domains: Literature (Testing-A), Computer (Testing-B), Medicine (Testing-C) and Finance (Testing-D). As there are only two unlabeled corpora for Domain A and B, we thus adopt different strategies for each testing-set:

- Testing-A: Character-Based Joint Model with semi-supervised learning, training on Labeled corpus and Unlabeled-A;
- Testing-B: Character-Based Joint Model with semi-supervised learning, training on Labeled corpus and Unlabeled-B;
- Testing-C and D: Character-Based Joint Model, training on Labeled corpus;

Table 1 shows that our system achieves F-scores for various testing-sets: 0.937 (A), 0.940 (B), 0.923 (C) and 0.957 (D), which are comparable with other systems. Among those four testing domains, our system performs unsatisfactorily on Testing-C (Medicine) even the OOV rate of this domain is not the highest. There are possible reasons for this result: (1) Semi-supervised learning is not conducted for this domain; (2) the statistical property between News and Medicine are significantly different.

Domain	Model	F1	R_{OOV}
A	J + R + S	0.937	0.652
	J + S	0.937	0.646
	J + R	0.936	0.646
	J	0.936	0.642
B	J + R + S	0.940	0.757
	J + S	0.931	0.721
	J + R	0.938	0.744
	J	0.927	0.699
C	J + R	0.923	0.674
	J	0.923	0.674
D	J + R	0.957	0.813
	J	0.954	0.786

Table 2: Performance of various approaches
J: Baseline, the character-based joint model
R: Adding restrictions in constructing lattice
S: Conduct Semi-Supervised Learning

3.2 Discussion

The aim of restrictions in constructing lattice is to improve the performance of English and numerical expressions, both of which appear frequently in Computer and Finance domain. Therefore, the improvements gained from these restrictions are significantly in these two domains (as shown in Table 2).

Besides, the adopted semi-supervised learning procedure improves the performance in Domain A and B., but the improvement is not significant. Semi-supervised learning aims to incorporate large amounts of unlabeled data. However, the size of unlabeled corpora provided here is too small. The semi-supervised learning procedure is expected to be more effective if a large amount of unlabeled data is available.

4 Conclusion

Our system is based on a character-based joint model, which combines a generative module and a discriminative module. In addition, we applied a semi-supervised learning method to the baseline approach to incorporate the unlabeled corpus. Our system achieves comparable performance with other participants. However, cross-domain performance is still not satisfactory and further study is needed.

Acknowledgement

The research work has been partially funded by the Natural Science Foundation of China under Grant No. 60975053, 90820303 and 60736014, the National Key Technology R&D Program under Grant No. 2006BAH03B02, and also the Hi-Tech Research and Development Program (“863” Program) of China under Grant No. 2006AA010108-4 as well.

References

Stanley F. Chen and Joshua Goodman, 1998. An empirical study of smoothing techniques for language modeling. *Technical Report TR-10-98, Harvard University Center for Research in Computing Technology*.

Wenbin Jiang, Liang Huang, Qun Liu and Yajuan Lu, 2008. A Cascaded Linear Model for Joint Chinese Word Segmentation and Part-of-Speech Tagging. In *Proceedings of ACL*, pages 897-904.

Adwait Ratnaparkhi, 1998. Maximum entropy models for natural language ambiguity resolution. University of Pennsylvania.

Andreas Stolcke, 2002. SRILM-an extensible language modeling toolkit. In *Proceedings of the International Conference on Spoken Language Processing*, pages 311-318.

Huihsin Tseng, Pichuan Chang, Galen Andrew, Daniel Jurafsky and Christopher Manning, 2005. A Conditional Random Field Word Segmenter for Sighan Bakeoff 2005. In *Proceedings of the Fourth SIGHAN Workshop on Chinese Language Processing*, pages 168-171.

Kun Wang, Chengqing Zong and Keh-Yih Su, 2009. Which is more suitable for Chinese word segmentation, the generative model or the discriminative one? In *Proceedings of the 23rd Pacific Asia Conference on Language, Information and Computation (PACLIC23)*, pages 827-834.

Kun Wang, Chengqing Zong and Keh-Yih Su, 2010. A Character-Based Joint Model for Chinese Word Segmentation. To *appear in COLING 2010*.

Nianwen Xue, 2003. Chinese Word Segmentation as Character Tagging. *Computational Linguistics and Chinese Language Processing*, 8 (1). pages 29-48.

Ruiqiang Zhang, Genichiro Kikui and Eiichiro Sumita, 2006. Subword-based Tagging for Confidence-dependent Chinese Word Segmentation. In *Proceedings of the COLING/ACL*, pages 961-968.

Xiaojin Zhu, 2006. Semi-supervised learning literature survey. *Technical Report 1530*, Computer Sciences, University of Wisconsin-Madison.