Two-Phase LMR-RC Tagging for Chinese Word Segmentation

Tak Pang Lau and **Irwin King**
Department of Computer Science and Engineering
The Chinese University of Hong Kong
Shatin, N.T., Hong Kong
{tplau, king}@cse.cuhk.edu.hk

Abstract

In this paper we present a Two-Phase LMR-RC Tagging scheme to perform Chinese word segmentation. In the Regular Tagging phase, Chinese sentences are processed similar to the original LMR Tagging. Tagged sentences are then passed to the Correctional Tagging phase, in which the sentences are re-tagged using extra information from the first round tagging results. Two training methods, Separated Mode and Integrated Mode, are proposed to construct the models. Experimental results show that our scheme in Integrated Mode performs the best in terms of accuracy, where Separated Mode is more suitable under limited computational resources.

1 Introduction

The Chinese word segmentation is a non-trivial task because no explicit delimiters (like spaces in English) are used for word separation. As the task is an important precursor to many natural language processing systems, it receives a lot of attentions in the literature for the past decade (Wu and Tseng, 1993; Sproat et al., 1996). In this paper, we propose a statistical approach based on the works of (Xue and Shen, 2003), in which the Chinese word segmentation problem is first transformed into a tagging problem, then the Maximum Entropy classifier is applied to solve the problem. We further improve the scheme by introducing correctional treatments after first round tagging. Two different training methods are proposed to suit our scheme.

The paper is organized as follows. In Section 2, we briefly discuss the scheme proposed by (Xue and Shen, 2003), followed by our additional works to improve the performance. Experimental and bakeoff results are presented in Section 3. Finally, We conclude the paper in Section 4.

2 Our Proposed Approach

2.1 Chinese Word Segmentation as Tagging

One of the difficulties in Chinese word segmentation is that, Chinese characters can appear in different positions within a word (Xue and Shen, 2003), and *LMR Tagging* was proposed to solve the problem. The basic idea of LMR Tagging is to assign to each character, based on its contextual information, a tag which represents its relative position within the word. Note that the original tag set used by (Xue and Shen, 2003) is simplified and improved by (Ng and Low, 2004) . We shall then adopt and illustrate the simplified case here.

The tags and their meanings are summarized in Table 1. Tag *L*, *M*, and *R* correspond to the character at the beginning, in the middle, and at the end of the word respectively. Tag *S* means the character is a "single-character" word. Figure 1 illustrates a Chinese sentence segmented by spaces, and the corresponding tagging results.

After transforming the Chinese segmentation problem to the tagging problem, various solutions can be applied. *Maximum Entropy model* (MaxEnt) (Berger, S. A. Della Pietra, and

Original sentence:	大衛喜歡吃士多啤梨。
After segmentation:	大衛 喜歡 吃 士多啤梨 。
Tagging:	L R L R S L M M R S

Figure 1: Example of LMR Tagging.

V. J. Della Pietra, 1996; Ratnaparkhi, 1996) was proposed in the original work to solve the LMR Tagging problem. In order to make MaxEnt success in LMR Tagging, *feature templates* used in capturing useful contextual information must be carefully designed. Furthermore, it is unavoidable that invalid tag sequences will occur if we just assign the tag with the highest probability. In the next subsection, we describe the feature templates and measures used to correct the tagging.

Table 1: Tags used in LMR Tagging scheme.

Tag	Description
L	Character is at the beginning of the word (or the character is the leftmost character in the word)
M	Character is in the middle of the word
R	Character is at the end of the word (or the character is the rightmost character in the word)
S	Character is a "single-character" word

2.2 Two-Phase LMR-RC Tagging

In this section, we introduce our *Two-Phase LMR-RC Tagging* used to perform Chinese Text Segmentation. The first phase, *R*-phase, is called *Regular Tagging*, in which similar procedures as in the original LMR Tagging are performed. The difference in this phase as compared to the original one is that, we use extra feature templates to capture characteristics of Chinese word segmentation. The second phase, *C*-phase, is called *Correctional Tagging*, in which the sentences are retagged by incorporating the regular tagging results. We hope that tagging errors can be corrected under this way. The models used in both phases are trained using MaxEnt model.

Regular Tagging Phase

In this phase, each character is tagged similar to the original approach. In our scheme, given the contextual information (x) of current character, the tag (y^*) with highest probability will be assigned:

$$y^* = \arg\max_{y \in \{L,M,R,S\}} p(y|x).$$

The features describing the characteristics of Chinese segmentation problem are instantiations of the feature templates listed in Table 2. Note that feature templates only describe the forms of features, but not the actual features. So the number of features used is much larger than the number of templates.

Table 2: Feature templates used in *R*-phase. Example used is "32 個蘋果".

	Feature Type	Example – Features extracted of character " 個"
1	Characters within window of ± 2	C_{-2}="3", C_{-1}="2", C_0=" 個", C_1=" 蘋", C_2=" 果"
2	Two consecutive characters within window of ± 2	$C_{-2}C_{-1}$="32", $C_{-1}C_0$="2 個", C_0C_1=" 個蘋", C_1C_2=" 蘋果"
3	Previous and next characters	$C_{-1}C_1$=" 2蘋"
4	Current character is punctuation	–
5	ASCII characters within window of ± 2	A_{-2}, A_{-1} (as "3" and "2" are ASCII)
6	Current and character in window ± 1 belong to different types	D_{-1} (as "2" is digit, but " 個" is letter)

Additional feature templates as compared to (Xue and Shen, 2003) and (Ng and Low, 2004) are template 5 and 6. Template 5 is used to handle documents with ASCII characters. For template 6, as it is quite common that word boundary occurs in between two characters with different types, this template is used to capture such characteristics.

Correctional Tagging Phase

In this phase, the sequence of characters is retagged by using the additional information of tagging results after *R*-phase. The tagging procedure is similar to the previous phase, except extra features (listed in Table 3) are used to assist the tagging.

Table 3: Additional feature templates used in *C*-phase. Example used is "32 個蘋果" with tagging results after *R*-phase as "SSLMR".

	Feature Type	Example – Features extracted of character " 個"
7	Tags of characters within window of ± 2	T_{-2}="S", T_{-1}="S", T_0="L", T_1="M", T_2="R"
8	Two consecutive tags within window of ± 2	$T_{-2}T_{-1}$="SS", $T_{-1}T_0$="SL", T_0T_1="LM", T_1T_2="MR"
9	Previous and next tags	$T_{-1}T_1$="SM"

Training Method

Two training methods are proposed to construct models used in *R*- and *C*-phase: (1) *Separated Mode*, and (2) *Integrated Mode*. Separated Mode means the models used in two phases are separated. Model for *R*-phase is called *R*-model, and model for *C*-phase is called *C*-model. Integrated Mode means only one model, *I*-model is used in both phases.

The training methods are illustrated now. First of all, training data are divided into three parts, (1) Regular Training, (2) Correctional Training, and (3) Evaluation. Our method first trains using observations extracted from Part 1 (observation is simply the pair $(context, tag)$ of each character). The created model is used to process Part 2. After that, observations extracted from Part 2 (which include previous tagging results) are used to create the final model. The performance is then evaluated by processing Part 3.

Let O be the set of observations, with subscripts R or C indicating the sources of them. Let $TrainModel : O \rightarrow P$, where P is the set of models, be the "model generating" function. The two proposed training methods can be illustrated as follow:

1. Separated Mode

$$R - model = TrainModel(O_R),$$
$$C - model = TrainModel(O_C).$$

2. Integrated Mode

$$I - model = TrainModel(O_R \cup O_C).$$

The advantage of Separated Mode is that, it is easy to aggregate different sets of training data. It also provides a mean to handle large training data under limited resources, as we can divide the training data into several parts, and then use the similar idea to train each part. The drawback of this mode is that, it may lose the features' characteristics captured from Part 1 of training data, and Integrated Mode is proposed to address the problem, in which all the features' characteristics in both Part 1 and Part 2 are used to train the model.

3 Experimental Results and Discussion

We conducted closed track experiments on the Hong Kong City University (CityU) corpus in The Second International Chinese Word Segmentation Bakeoff to evaluate the proposed training and tagging methods. The training data were split into three portions. Part 1: 60% of the data is trained for *R*-phase; Part 2: 30% for *C*-phase training; and Part 3: the remaining 10% for evaluation. The evaluation part was further divided into six parts to simulate actual size of test document. The MaxEnt classifier was implemented using Java opennlp maximum entropy package from (Baldridge, Morton, and Bierner, 2004), and training was done with feature cutoff of 2 and 160 iterations. The experiments were run on an Intel Pentium4 3.0GHz machine with 3.0GB memory.

To evaluate our proposed scheme, we carried out four experiments for each evaluation data. For Experiment 1, data were processed with *R*-phase only. For Experiment 2, data were processed with both *R*- and *C*-phase, using Separated Mode as training method. For Experiment 3, data were processed similar to Experiment 2, except Integrated Mode was used. Finally for Experiment 4, data were processed similar to Experiment 1, with both Part 1 and Part 2 data were used for *R*-model training. The purpose of Experiment 4 is to determine whether the proposed scheme can perform better than just the single Regular Tagging under the same amount of training data. Table 4 summarizes the experimental results measured in F-measure (the harmonic mean of precision and recall).

From the results, we obtain the following observations.

1. Both Integrated and Separated Training modes

Table 4: Experimental results of CityU corpus measured in F-measure.

Data Set	Exp1	Exp2	Exp3	Exp4
1	0.918	0.943	**0.949**	0.947
2	0.913	0.939	**0.943**	**0.943**
3	0.912	0.935	**0.939**	0.937
4	0.914	0.940	**0.943**	0.942
5	0.921	0.942	**0.945**	**0.945**
6	0.914	0.941	**0.945**	0.942

in Two-Phase Tagging (Exp 2 and Exp 3) outperform single Regular Tagging (Exp 1). It is reasonable as more data are used in training.

2. Integrated Mode (Exp 3) still performs better than Exp 4, in which same amount of training data are used. This reflects that extra tagging information after *R*-phase helps in the scheme.
3. Separated Mode (Exp 2) performs worse than both Exp 3 and Exp 4. The reason is that the *C*-model cannot capture enough features' characteristics used for basic tagging. We believe that by adjusting the proportion of Part 1 and Part 2 of training data, performance can be increased.
4. Under limited computational resources, in which constructing single-model using all available data (as in Exp 3 and Exp 4) is not possible, Separated Mode shows its advantage in constructing and aggregating multi-models by dividing the training data into different portions.

The official BakeOff2005 results are summarized in Table 5. We have submitted multiple results for CityU, MSR and PKU corpora by applying different tagging methods described in the paper.

Table 5: Official BakeOff2005 results.
Keys:
F - Regular Tagging only, all training data are used
P1 - Regular Tagging only, 90% of training data are used
P2 - Regular Tagging only, 70% of training data are used
S - Regular and Correctional Tagging, Separated Mode
I - Regular and Correctional Tagging, Integrated Mode

Corpus	R	P	F	R_{OOV}	R_{IV}	Method
CityU	0.938	0.915	0.927	0.658	0.961	F
	0.936	0.913	0.925	0.656	0.959	P1
	0.925	0.896	0.910	0.639	0.948	P2
	0.937	0.922	0.929	0.698	0.956	I
MSR	0.946	0.933	0.939	0.587	0.956	F
	0.941	0.932	0.937	0.624	0.950	S
PKU	0.926	0.908	0.917	0.535	0.950	F
	0.917	0.903	0.910	0.600	0.937	P2
	0.918	0.915	0.917	0.621	0.936	I

4 Conclusion

We present a Two-Phase LMR-RC Tagging scheme to perform Chinese word segmentation. Correctional Tagging phase is introduced in addition to the original LMR Tagging technique, in which the Chinese sentences are re-tagged using extra information of first round tagging results. Two training methods, *Separated Mode* and *Integrated Mode*, are introduced to suit our scheme. Experimental results show that Integrated Mode achieve the highest accuracy in terms of F-measure, where Separated Mode shows its advantages in constructing and aggregating multi-models under limited resources.

Acknowledgements

The work described in this paper was fully supported by a grant from the Research Grants Council of the Hong Kong Special Administrative Region, China (Project No. CUHK4235/04E).

References

A. L. Berger, S. A. Della Pietra, and V. J. Della Pietra. 1996. A maximum entropy approach to natural language processing. *Computational Linguistics*, 22(1):39-71.

A. Ratnaparkhi. 1996. A maximum entropy model for part-of-speech tagging. In *Proceedings of the First Conference on Empirical Methods in Natural Language Processing*, pages 133-142.

H. T. Ng and J. K. Low. 2004. Chinese Part-of-Speech Tagging. One-at-a-Time or All-at-once? Word-Based or Character-Based? In *Proc. of EMNLP*.

J. Baldridge, T. Morton, and G. Bierner. 2004. The opennlp maxent package in Java. URL: *http://maxent.sourceforge.net* .

N. Xue and L. Shen. 2003. Chinese word segmentation as LMR Tagging. In *Proc. of SIGHAN Workshop*.

R. Sproat, C. Shih, W. Gale, and N. Chang. 1996. A stochastic finite-state word-segmentation algorithm for Chinese. *Computational Linguistics*, 22(3):377-404.

R. Sproat and T. Emerson. 2003. The first international Chinese word segmentation bakeoff. In *Proc. of SIGHAN Workshop*.

Z. Wu and G. Tseng. 1993. Chinese text segmentation for text retrieval: achievements and problems. *Journal of the American Society for Information Science*, 44(9):532-542.