Character-based Chinese Word Segmentation and Pos-tagging with Unsupervised Unknown Word Learning

Kun Yu
Graduate School of Information Science and Technology
The University of Tokyo
Tokyo, 113-8656, Japan
kunyu@kc.t.u-tokyo.ac.jp

Sadao Kurohashi
Graduate School of Informatics
Kyoto University
Kyoto, 606-8501, Japan
kuro@nlp.kuee.kyoto-u.ac.jp

Hao Liu
Graduate School of Information Science and Technology
The University of Tokyo
Tokyo, 113-8656, Japan
liuhao@kc.t.u-tokyo.ac.jp

Abstract

This paper proposed a character-tagging approach to do Chinese word segmentation and pos-tagging. Word-level features are used to overcome the known word detection problem of character-based method. Besides of that, an unsupervised unknown word learning method is applied to enlarge the lexicon for word-level feature detection. Experimental results based on Penn Chinese Treebank 5.1 showed that using word-level features gave 3.23% and 5.01% improvements on *F1* for word segmentation and pos-tagging, respectively. Moreover, by using the unsupervised unknown word learning method to enlarge the lexicon, the *F1* of word segmentation and pos-tagging was increased by 0.45% and 0.39% again.

1 Introduction

As one of the basic analysis tasks, word segmentation and pos-tagging are very crucial to natural language processing applications, especially for Asian languages such as Chinese and Japanese. Dealing with them together in character-level has been proved very useful regardless of its time costs (H.T.Ng and J.K.Low, 2004). But only using character information may bring difficulty to known word identification (T.Nakagawa, 2004).

Adding features based on a large lexicon is a promising way to solve this problem (F.C.Peng et al., 2004). But obtaining a large lexicon automatically is not easy for many applications. Some researchers tend to use unknown word learning to help create a large lexicon. For example, J.K.Low et al. (2005) learn unknown words from the segmentation result of other corpus to enlarge the lexicon. But they need gold-standard segmentation as criterion for unknown word detection, which restrict the type of corpus that can be used.

In this paper, we proposed a character-tagging based approach to do word segmentation and pos-tagging. It combines both character-level features and word-level features and works in three steps:

Step 1: do word segmentation and pos-tagging on testing data with a small lexicon only from training data;

Step 2: use an unsupervised unknown word learning method to learn unknown words from segmentation results without any gold-standard and add them into lexicon;

Step 3: use the new lexicon to do word segmentation and pos-tagging on testing data again.

We did experiments on Penn Chinese Treebank 5.1 (N.Xue et al.,2002). Results show that using word-level features gave improvements on *F1* for both word segmentation and pos-tagging. Moreover, by using the unsupervised unknown word learning method to enlarge the lexicon, the *F1* of word segmentation and pos-tagging can be increased again.

The rest of this paper is organized as follows. Section 2 gives an introduction to our proposed approach. Section 3 introduces the unsupervised unknown word learning method. Experimental results and discussions are described in Section 4. At last, Section 5 gives a brief conclusion and future work.

2 Word Segmentation and Pos-tagging by Character-tagging

2.1 Task Definition

The proposed approach looks word segmentation and pos-tagging as a tagging task and deals with them in character-level at the same time. The task is to find the tag sequence T^* with the highest probability given a sequence of characters $S=c_1c_2\ldots c_n$.

$$T^* = \arg\max_T P(T \mid S) \quad (1)$$

Then we assume that the tagging of one character is independent of each other, and modify eq. 1 as

$$T^* = \arg\max_{T=t_1t_2\ldots t_n} P(t_1t_2\ldots t_n \mid c_1c_2\ldots c_n) = \arg\max_{T=t_1t_2\ldots t_n} \prod_{i=1}^{n} P(t_i \mid c_i) \quad (2)$$

Four tags *B, I, E, S* are defined to get word boundary, in which *B* means the character is the beginning of one word, *I* means the character is inside one word, *E* means the character is the end of one word and *S* means the character is one word by itself. Besides of that, 33 pos-tags according to the pos-tag definition in Penn Chinese Treebank 5.1 are defined. Then, we combine the tags of word boundary and pos-tag together, and get 4×33 tags finally.

Beam search (n=3) (Ratnaparkhi, 1996) is applied for tag sequence searching. We only search the valid sequences to ensure the validity of searching result. SVM is selected as the basic classification model for tagging because of its robustness to over-fitting and high performance (Sebastiani, 2002). To simplify the calculation, the output of SVM is regarded as $P(t_i|c_i)$.

2.2 Character-level Feature

In the proposed approach, we define two types of features: character-level feature and word-level feature. The character-level features are listed in the following:

- C_n (n=-2,-1,0,1,2)
- $Pu(C_0)$

Feature C_n mean the Chinese characters appearing in different positions (the current character and two characters to its left and right), and they are binary features. Feature $Pu(C_0)$ means whether C_0 is in a punctuation character list. It is also binary feature and all the punctuations in the punctuation character list come from Penn Chinese Treebank 5.1.

2.3 Word-level Features

Only using character-level feature is good for unknown word detection, but it also makes some mistakes when identifying the boundary for known words. So we add word-level features based on a lexicon to solve this problem.

The word-level features are defined as:

- W_n (n=-1,0,1)

Feature W_n mean the lexicon words in different positions (the word containing C_0 and one word to its left and right) and they are also binary features. Here we select all the possible words in the lexicon that satisfy the requirements, not like only selecting the longest one in (J.K.Low et al.,2005). For example, for a character sequence '球拍卖完了(racket was sold out)', there are two groups of word-level features when we consider about character '拍' as C_0 (see Figure 1).

球拍卖完了

W_{-1}	W_0	W_1
球	拍卖	完
---	球拍	卖

Figure 1. Different word-level features for a character sequence

3 Unsupervised Unknown Word Learning

3.1 Learning Procedure

In section 2, a lexicon from training data is used to identify word-level features. While, only using this lexicon cannot detect the unknown words in testing data. To help enlarge the lexicon, we proposed an unsupervised unknown word learning method. In this method, we first apply the character-tagging approach introduced in Section 2 to testing data. Then we extract those words that not only meet some predefined criteria but also do not appear in the existed lexicon as unknown words. Finally we add the extracted unknown words into our lexicon.

3.2 Criteria for Unknown Word Detection

Because the segmentation result in the first iteration does not have 100% accuracy usually, we use the probability of one word being unknown word

$P(UW \mid w)$ as criterion to detect unknown words from the segmentation result. To calculate this probability, three feature functions are defined based on following assumptions.

Assumption 1: we suppose that if one word is between two known words, it is like to be a correct segmented unknown word. Under this assumption, we define the known word distance *kw_dis(w)* of one word (eq. 3). Here known word means the word in the lexicon.

$$kw\_dis(w) = \frac{1}{dis_{left}} + \frac{1}{dis_{right}} \quad (3)$$

In eq.3, dis_{left} means the distance between the first known word in the left side and word *w*, and dis_{right} means the distance between the first known word in the right side and word *w*.

Assumption 2: we suppose that if one segmented word occurs many times, it is more like to be one correct unknown word. Then we define the word frequency *freq(w)* (eq. 4), which means the occurrence times of word *w* compared with the occurrence times of character sequence $c_1c_2...c_n$.

$$freq(w = c_1c_2...c_n) = \frac{count(w)}{count(c_1c_2...c_n)} \quad (4)$$

Assumption 3: we assume that even if two words have the same frequency (eq. 4), the word that occurs more in the text is believed more like a correct segmentation. Under this assumption, we define the occurrence times of one word *occu(w)* (eq. 5), which means the total occurrence times of character sequence $c_1c_2...c_n$ as one word.

$$occu(w = c_1c_2...c_n) = count(c_1c_2...c_n) \quad (5)$$

To calculate the probability $P(UW \mid w)$ by above feature functions, the value of each feature function should be between 0 and 1. Among the three defined functions, only *freq(w)* meets this requirement. So we first normalize other feature functions between 0 and 1 by eq. 6 and eq. 7. Then we combine them with predefined weights to calculate $P(UW \mid w)$ (eq. 8). At last, we extract the word *w* with $P(UW \mid w) > thre$ as unknown word and add them into the lexicon.

$$kw\_dis_{nor}(w) = \frac{1}{2n}\sum_{i=1}^{n} kw\_dis(w^i) \quad (6)$$

(where w^i means the i_{th} occurrence of word *w*)

$$occu_{nor}(w = c_1c_2...c_n) = 1 - \frac{1}{occu(w = c_1c_2...c_n)} \quad (7)$$

$$P(UW \mid w) = \lambda_1 \times kw\_dis_{nor}(w) + \lambda_2 \times freq(w) + \lambda_3 \times occu_{nor}(w) \quad (8)$$

In our experiments, λ_i in eq. 8 are defined by hand as $\lambda_1 = 0.4$, $\lambda_2 = 0.3$ and $\lambda_3 = 0.3$. The difference among the three weights is small, because we believe that testing data follows the same segmentation criteria as the training data. The threshold for unknown word extraction is set as *thre* = 0.7.

4 Experimental Results and Discussion

4.1 Data Set and Experimental Setting

We use Penn Chinese Treebank 5.1 as training and testing data. All the data are divided into two parts: 90% for training and 10% for testing.

Precision, *recall*, and *F1* (eq. 9) are used as the basic evaluation metrics. In addition, *Roov* and *Riv*, which are the recall of *Out-Of-Vocabulary-Word* and the recall of *In-Vocabulary-Word*, are also used to evaluate the ability of known word and unknown word identification.

$$F1 = \frac{2 \times precision \times recall}{precision + recall} \quad (9)$$

We selected SVMlight (Joachims, 1999) as the SVM classifier toolkit. Linear kernel is applied for its rapidness.

Three models were tested in the experiment. ‘w/o word’ means doing word segmentation and pos-tagging with only character-level features. ‘w/ word’ means using both character and word level features, but without unknown word learning. ‘w/ word + UWL’ means using both character and word level features and unknown word learning.

4.2 Results

Table 1 lists the recall of known word and unknown word detection. It shows that compared with ‘w/o word’ model, the ‘w/ word’ model got improvement on *Riv* by 6.16%, but *Roov* dropped by 9.94% at the same time. It is because using word-level features can help known word detection but decrease the ability of unknown word identification simultaneously. But Table 2 shows that even in such case, *F1* of both word segmentation and pos-tagging in ‘w/ word’ model were improved by 3.23% and 5.01% respec-

tively, which proves the validity of using word-level features in character-tagging based approach.

Table 1 Recall of known word and unknown word detection

	Roov (%)	*Riv* (%)
w/o word	75.39	89.32
w/ word	65.45 (-9.94)	95.48 (+6.16)
w/ word + UWL	67.20 (+1.75)	95.52 (+0.04)

Table 2 Results of word segmentation and pos-tagging

	Word Segmentation			Pos-tagging		
	Pre. (%)	*Rec.* (%)	*F1* (%)	*Pre.* (%)	*Rec.* (%)	*F1* (%)
w/o word	90.86	90.68	90.77	83.18	83.03	83.11
w/ word	93.30	94.72	94.00 (+3.23)	87.45	88.79	88.12 (+5.01)
w/ word + UWL	93.92	94.98	94.45 (+0.45)	88.02	89.01	88.51 (+0.39)

In addition, Table 1 shows that compared with 'w/ word' model, 'w/ word + UWL' model got 1.75% and 0.04% increase on *Roov* and *Riv*. It is because the proposed unknown word learning method is helpful to balance known word detection and unknown word identification. For the same reason, the *F1* of word segmentation and pos-tagging were improved by 0.45% and 0.39% again in 'w/ word + UWL' model (see Table 2), which shows the effectiveness of the proposed unsupervised unknown word learning method.

4.3 Discussion

(H.T.Ng and J.K.Low, 2004) first applied character-tagging to Chinese word segmentation and pos-tagging. Because we use different testing data, it is difficult to compare the evaluation of our work with them. But analysis shows that our approach is different from theirs in two ways: first, we enlarged the window size of word-level features and selected all the possible words in the lexicon; second, we applied an unsupervised unknown word learning method to learn unknown word from segmentation results.

In our unknown word learning method, the weight setting of the feature functions is made by-hand currently. This empirical setting may not match the real data. We will consider about learning these weights in our future work.

5 Conclusion and Future Work

This paper brings forward a character-tagging based approach to do word segmentation and pos-tagging together. To solve the known word detection problem, word-level features are used. Besides of that, an unsupervised unknown word learning method is proposed to enlarge the lexicon, which is helpful to balance the known word and unknown word identification. Experimental results proved that using word-level features is helpful for both word segmentation and pos-tagging. In addition, the unsupervised unknown words learning method also gives help for both known word and unknown word identification.

While, there are still some works needed to do in the future, such as the learning of weight setting, the feature definition, and so on.

References

T.Joachims. 1999. Making large-scale SVM learning practical. *Advances in Kernel Methods - Support Vector Learning*. MIT-Press.

J.K.Low et al. 2005. A Maximum Entropy Approach to Chinese Word Segmentation. In *SIGHAN 4*.

T.Nakagawa. 2004. Chinese and Japanese Word Segmentation Using Word-level and Character-level Information. In *COLING 2004*.

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 *EMNLP 2004*.

F.C.Peng et al. 2004. Chinese Segmentation and New Word Detection using Conditional Random Fields. In *COLING 2004*.

A.Ratnaparkhi. 1996. A Maximum Entropy Model for Part-of-Speech Tagging. In *EMNLP 1996*.

F.Sebastiani. 2002. Machine learning in automated text categorization. *ACM Computing Surveys*, 34(1):1-47.

N.Xue et al. 2002. Building a Large-Scale Annotated Chinese Corpus. In *COLING 2002*.