View metadata, citation and similar papers at core.ac.uk

brought to you by CORE

provided by Crossref

Using Self-Supervised Word Segmentation in Chinese Information Retrieval

Fuchun Peng[1] Xiangji Huang[1] Dale Schuurmans[1] Nick Cercone[1] Stephen Robertson[2]
[1]Computer Science Department, University of Waterloo, Waterloo, Canada
{f3peng, jhuang, dale, ncercone}@uwaterloo.ca
[2]Microsoft Research, Cambridge, U.K. and City University, London, U.K.
ser@microsoft.com

ABSTRACT

We propose a self-supervised word-segmentation technique for Chinese information retrieval. This method combines the advantages of traditional dictionary based approaches with character based approaches, while overcoming many of their shortcomings. Experiments on TREC data show comparable performance to both the dictionary based and the character based approaches. However, our method is language independent and unsupervised, which provides a promising avenue for constructing accurate multilingual information retrieval systems that are flexible and adaptive.

Categories and Subject Descriptors

I.2 [**Computing Methodologies**]: artificial intelligence

General Terms

Experimentation

Keywords

Self-supervised word segmentation, Chinese IR

1. INTRODUCTION

Tokenization is the first step in Chinese information retrieval, where traditionally two approaches have been taken: dictionary based and character based [2, 3]. In the dictionary based approach, one pre-defines a lexicon containing a large number of Chinese words and then uses heuristic methods such as maximum matching to segment Chinese sentences. In the character based approach, sentences are tokenized simply by taking each character to be a basic unit. Both of these approaches have advantages and disadvantages. The dictionary based approach has the advantage of requiring a smaller inverted index file, hence achieving faster retrieval times while allowing additional linguistic information to be incorporated into the retrieval system (e.g. synonyms). The main disadvantage of the dictionary based approach however is that it requires a large pre-defined lexicon, which normally must be constructed by hand with significant amount of labor and time. In the character based approach, the most prominent advantage is that it does not require a pre-defined lexicon. Each character is simply considered as a basic unit. However, the disadvantages are that the character based approach requires a huge index file, resulting in slower retrieval speed, and creates difficulty in incorporating any additional linguistic information.

In this paper, we propose using an EM based segmentation technique for Chinese information retrieval, self-supervised segmentation [4], which has many of the advantages of the character based and dictionary based approaches, while overcoming many of the shortcomings of both methods.

2. SELF-SUPERVISED SEGMENTATION

In a general word segmentation task where there are no identifying markers between words, one could effectively exploit *known* words to guide the segmentation of unknown words. For example, if the word “*computer*” is already known then upon seeing the text “*computerscience*” it is natural to segment “*science*” as a possible new word. To exploit this observation, we develop an EM based word discovery method that is a variant of standard EM training, but avoids getting trapped in local maxima by keeping two lexicons: a *core* lexicon which contains words that are judged to be trustworthy, and a *candidate* lexicon which contains all other candidate words that are not in the core lexicon.

Assume we have a sequence of characters $C = c_1c_2...c_T$ that we wish to segment into chunks $S = s_1s_2...s_M$, where T is the number of characters in the sequence and M is the number of words in the segmentation. Here chunks s_i will be chosen from the core lexicon $V_1 = \{s_i, i = 1, ..., |V_1|\}$ or the candidate lexicon $V_2 = \{s_j, j = 1, ..., |V_2|\}$. If we already have the probability distributions $\theta = \{\theta_i|\theta_i = p(s_i), i = 1, ..., |V_1|\}$ defined over the core lexicon and $\phi = \{\phi_j|\phi_j = p(s_j), j = 1, ..., |V_2|\}$ over the candidate lexicon, then we can recover the most likely segmentation of the sequence $C = c_1c_2...c_T$ into chunks $S = s_1s_2...s_M$ as follows. First, for any given segmentation S of C, we can calculate the joint likelihood of S and C by

$$prob(S, C|\theta, \phi) = \prod_{i=1}^{M_1} \frac{p(s_i)}{2} \prod_{j=1}^{M_2} \frac{p(s_j)}{2} = \frac{1}{2^M} \prod_{k=1}^{M} p(s_k)$$

where M_1 is the number of chunks occurring in the core lex-

Copyright is held by the author/owner.
SIGIR'02, August 11-15, 2002, Tampere, Finland.
ACM 1-58113-561-0/02/0008.

icon, M_2 is the number of chunks occurring in the candidate lexicon, and s_k can come from either lexicon. (Note that each chunk s_k must come from exactly one of the core or candidate lexicons.) Our task is to find the segmentation S^* that achieves the maximum likelihood:

$$S^* = \underset{S}{argmax}\{prob(S, C|\theta, \phi)\} \qquad (1)$$

Given a probability distribution defined by θ and ϕ over the lexicon, the Viterbi algorithm can be used to efficiently compute the best segmentation S of character string C. Estimation of the probabilities can be done by the EM algorithm. The parameter re-estimation formulas are as follows.

$$\theta_i^{k+1} = \frac{\sum_S \#(s_i, S) \times prob(S, C|\theta^k, \phi^k)}{\sum_{s_i} \sum_S \#(s_i, S) \times prob(S, C|\theta^k, \phi^k)} \qquad (2)$$

$$\phi_j^{k+1} = \frac{\sum_S \#(s_j, S) \times prob(S, C|\theta^k, \phi^k)}{\sum_{s_j} \sum_S \#(s_j, S) \times prob(S, C|\theta^k, \phi^k)} \qquad (3)$$

where $\#(s_i, S)$ is the number of times s_i occurring the segmentation S.

The two lexicons are constructed automatically as follows. Let us define C_1, C_2 as the training corpus and the validation corpus respectively, and let V_1 and $V2$ be the core candidate lexicons respectively. Initially, V_1 is set to be empty and V_2 is initialized to contain all candidate "words" that are generated from the training corpus by enumerating contiguous character strings of lengths 1 to L for some predefined maximum length L. In a first pass, starting from the uniform distribution, EM is used to increase the likelihood of the training corpus C_1. When the training process stabilizes, the M words with highest probability are selected from V_2 and moved to V_1, after which all the probabilities are rescaled so that V_1 and V_2 each contain half the total probability mass. EM is then run again. The rationale for shifting half of the probability mass to V_1 is that this increases the influence of core words in determining segmentations and allows them to act as more effective guides in processing the training sequence. We call this procedure of successively moving the top M words to V_1 *forward selection*. Forward selection is repeated until the segmentation performance of Viterbi on the validation corpus C_2 leads to a decrease in F-measure (which means we must have included some erroneous words in the core lexicon). After forward selection terminates, M is decremented and we carry out a process of *backward deletion*, where the M words with the lowest probability in V_1 are moved back to V_2, and EM training is successively repeated until F-measure again decreases on the validation corpus C_2 (which means we must have deleted some correct core words). The two procedures of forward selection and backward deletion are alternated, decrementing M at each alternation, until $M \leq 0$;

3. EXPERIMENTAL RESULTS

We evaluate the IR performance of the *self-supervised* segmentation algorithm with the Okapi system [1, 2] on TREC data sets. Performance measures include *Average Precision*: average precision over all 11 recall points (0.0, 0.1, 0.2,..., 1.0) and *R Precision*: precision after the number of documents retrieved is equal to the number of known relevant documents for a query.

Our segmenter is trained on a 90M collection of data containing one year of *People's Daily* news service stories, and uses 2000 random sentences from the segmented Chinese Treebank as the validation set. Our segmentation accuracy is around 70-74% on the Chinese Treebank.

We compare the IR performance of our method against two standard tokenization methods. The first method uses a hand built dictionary of words, compound words, and phrases to index the texts. We refer to this method as the dictionary-based approach. The dictionary we use in the experiments contains 69,353 Chinese words and phrases. The second method we compare to is a character based approach where documents are indexed by the single Chinese characters that appear in the text. We show the experimental results of the three methods on TREC data sets in the following table. We set the dictionary based method as the baseline.

Results comparison on TREC-5 data set			
	character	dictionary	EM-based
Average precision	0.3795	0.3468	0.3661
Avg. pre. improvement	9.43%	baseline	5.57%
R-Precision	0.3963	0.3863	0.4027
R-Precision improvement	2.59%	baseline	4.25%

Results comparison on TREC-6 data set			
	character	dictionary	EM-based
Average precision	0.5348	0.5044	0.4970
Avg. pre. improvement	6.03%	baseline	-1.47%
R-Precision	0.5404	0.5055	0.5001
R-Precision improvement	6.90%	baseline	-1.07%

On TREC-5 data, we find the EM based segmentation gives a 5.57% improvement in average accuracy over the dictionary based method, but it does a little worse than the character based method. In terms of R-precision, the EM based method yields better performance than both the character based and dictionary based methods. On TREC-6 data, the EM based method yields slightly worse results than both the dictionary based and the character based methods.

In terms of retrieval time, the EM based methods are at the same level as the dictionary based methods, which are about three times faster than the character based approach.

4. CONCLUSIONS

Although our EM based segmentation method does not yield completely accurate segmentations by itself, it nevertheless performs well as a basis for Chinese IR. We achieve retrieval performance that this comparable (and sometimes even better) than the manual dictionary based and the expensive character based methods. Our results demonstrate the machine learning techniques can be successfully applied to Chinese IR to build an adaptable system.

5. REFERENCES

[1] M. Beaulieu, M. Gatford, X. Huang, S. Robertson, S. Walker, and P. Williams. Okapi at TREC-5. In *D.K.Harman (ed): Proceedings of TREC-5*, pages 143-166, 1997

[2] X. Huang and S. Roberton. A Probabilistic Approach to Chinese Information Retrieval: Theory and Experiments. In *Proceedings of the BCS-IRSG 2000.*

[3] J. Nie and F. Ren. Chinese information retrieval: using characters or words? In *Information Processing and Management*, 35:443-462, 1999.

[4] F. Peng and D. Schuurmans. Self-supervised Chinese Word Segmentation. In *Proceedings of IDA-01*, 2001.