Information Retrieval Oriented Word Segmentation based on Character Associative Strength Ranking

Yixuan Liu, Bin Wang, Fan Ding, Sheng Xu
Information Retrieval Group
Center for Advanced Computing Research
Institute of Computing Technology
Chinese Academy of Sciences
Beijing, 100190, P.R.China
{liuyixuan, wangbin, dingfan, xusheng}@ict.ac.cn

Abstract

This paper presents a novel, ranking-style word segmentation approach, called *RSVM-Seg*, which is well tailored to Chinese information retrieval(CIR). This strategy makes segmentation decision based on the ranking of the internal associative strength between each pair of adjacent characters of the sentence. On the training corpus composed of query items, a ranking model is learned by a widely-used tool Ranking SVM, with some useful statistical features, such as mutual information, difference of t-test, frequency and dictionary information. Experimental results show that, this method is able to eliminate overlapping ambiguity much more effectively, compared to the current word segmentation methods. Furthermore, as this strategy naturally generates segmentation results with different granularity, the performance of CIR systems is improved and achieves the state of the art.

1 Introduction

To improve information retrieval systems' performance, it is important to comprehend both queries and corpus precisely. Unlike English and other western languages, Chinese does not delimit words by white-space. Word segmentation is therefore a key preprocessor for Chinese information retrieval to comprehend sentences.

Due to the characteristics of Chinese, two main problems remain unresolved in word segmentation: segmentation ambiguity and unknown words, which are also demonstrated to affect the performance of Chinese information retrieval (Foo and Li, 2004). Overlapping ambiguity and combinatory ambiguity are two forms of segmentation ambiguity. The first one refers to that ABC can be segmented into AB C or A BC. The second one refers to that string AB can be a word, or A can be a word and B can be a word. In CIR, the combinatory ambiguity is also called segmentation granularity problem (Fan et al., 2007). There are many researches on the relationship between word segmentation and Chinese information retrieval (Foo and Li, 2004; Peng et al., 2002a; Peng et al., 2002b; Jin and Wong, 2002). Their studies show that the segmentation accuracy does not monotonically influence subsequent retrieval performance. Especially the overlapping ambiguity, as shown in experiments of (Wang, 2006), will cause more performance decrement of CIR. Thus a CIR system with a word segmenter better solving the overlapping ambiguity, may achieve better performance. Besides, it also showed that the precision of new word identification was more important than the recall.

There are some researches show that when compound words are split into smaller constituents, better retrieval results can be achieved (Peng et al., 2002a). On the other hand, it is reasonable that the longer the word which co-exists in query and corpus, the more similarity they may have. A hypothesis, therefore, comes to our mind, that different segmentation granularity can be incorporated to obtain better CIR performance.

In this paper we present a novel word segmentation approach for CIR, which can not only obviously reduce the overlapping ambiguity, but also introduce different segmentation granularity for the first time.

Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing, pages 1061–1069,
Honolulu, October 2008.

In our method, we first predict the ranking result of all internal association strength (IAS) between each pair of adjacent characters in a sentence using Ranking SVM model, and then, we segment the sentence into sub-sentences with smaller and smaller granularity by cutting adjacent character pairs according to this rank. Other machine-learning based segmentation algorithms (Zhang et al., 2003; Lafferty et al., 2001; Ng and Low, 2004) treat segmentation problem as a character sequence tagging problem based on classification. However, these methods cannot directly obtain different segmentation granularity. Experiments show that our method can actually improve information retrieval performance.

This paper is structured as follows. It starts with a brief introduction of the related work on the word segmentation approaches. Then in Section 3, we introduce our segmentation method. Section 4 evaluates the method based on experimental results. Finally, Section 5 makes summary of this whole paper and proposes the future research orientation.

2 Related Work

Various methods have been proposed to address the word segmentation problem in previous studies. They fall into two main categories, rule-based approaches that make use of linguistic knowledge and statistical approaches that train on corpus with machine learning methods. In rule-based approaches, algorithms of string matching based on dictionary are the most commonly used, such as maximum matching. They firstly segment sentences according to a dictionary and then resort to some rules to resolve ambiguities (Liu, 2002; Luo and Song, 2001). These rule-based methods are fast, however, their performances depend on the dictionary which cannot include all words, and also on the rules which cost a lot of time to make and must be updated frequently. Recent years statistical approaches became more popular. These methods take advantage of various probability information gained from large corpus to segment sentences. Among them, Wang's work (Wang, 2006) is the most similar to our method, since both of us apply statistics information of each gap in the sentence to eliminate overlapping ambiguity in methods. However, when combining different statistics, Wang decided the weight by a heuristic way which was too simply to be suitable for all sentences. In our method, we employ a machine-learning method to train features' weights.

Many machine-learning methods, such as HMM (Zhang et al., 2003), CRF (Lafferty et al., 2001), Maximum Entropy (Ng and Low, 2004), have been exploited in segmentation task. To our knowledge, machine-learning methods used in segmentation treated word segmentation as a character tagging problem. According to the model trained from training corpus and features extracted from the context in the sentence, these methods assign each character a positional tag, indicating its relative position in the word. These methods are difficult to get different granularity segmentation results directly. Our method has two main differences with them. Firstly, we tag the gap between characters rather than characters themselves. Secondly, our method is based on ranking rather than classification.

Then, we will present our ranking-based segmentation method, *RSVM-Seg*.

3 Ranking based Segmentation

Traditional segmentation methods always take the segmentation problem as classification problem and give a definite segmentation result. In our approach, we try to solve word segmentation problem from the view of ranking. For easy understanding, let's represent a Chinese sentence S as a character sequence:

$$C_{1:n} = C_1 C_2 \dots C_n$$

We also explicitly show the gap $G_i (i = 1 \dots n-1)$ between every two adjacent characters C_i and C_{i+1}:

$$C_{1:n} | G_{1:n-1} = C_1 G_1 C_2 G_2 \dots G_{n-1} C_n$$

$IAS_i (i = 1 \dots n)$ is corresponding to $G_i (i = 1 \dots n)$, reflecting the internal association strength between C_i and C_{i+1}. The higher the IAS value is, the stronger the associative between the two characters is. If the association between two characters is weak, then they can be segmented. Otherwise, they should be unsegmented. That is to say we could make segmentation based on the ranking of IAS value. In our ranking-style segmentation method, Ranking SVM is exploited to predict IAS ranking.

In next subsections, we will introduce how to take advantage of Ranking SVM model to solve our

problem. Then, we will describe features used for training the Ranking SVM model. Finally, we will give a scheme how to get segmentation result from predicted ranking result of Ranking SVM.

3.1 Segmentation based on Ranking SVM

Ranking SVM is a classical algorithm for ranking, which formalizes learning to rank as learning for classification on pairs of instances and tackles the classification issue by using SVM (Joachims, 2002). Suppose that $X \epsilon R^d$ is the feature space, where d is the number of features, and $Y = r_1, r_2, \ldots, r_K$ is the set of labels representing ranks. And there exists a total order between ranks $r_1 > r_2 > \ldots > r_K$, where $>$ denotes the order relationship. The actual task of learning is formalized as a Quadratic Programming problem as shown below:

$$\begin{aligned} &min_{\omega,\xi_{ij}} \frac{1}{2}\|\omega\|^2 + C\Sigma\xi_{ij} \\ &s.t. \langle \omega, x_i - x_j \rangle > 1 - \xi_{ij}, \forall x_i \succ x_j, \xi_{ij} \geq 0 \end{aligned} \tag{1}$$

where $\|\omega\|$ denotes l_2 norm measuring the margin of the hyperplane and ξ_{ij} denotes a slack variable. $x_i \succ x_j$ means the rank class of x_i has an order prior to that of x_j, i.e. $Y(x_i) > Y(x_j)$. Suppose that the solution to (1) is ω_*, then we can make the ranking function as $f(x) = \langle \omega_*, x \rangle$.

When applying Ranking SVM model to our problems, an instance (feature vector x) is created from all bigrams (namely $C_i C_{i+1}, i = 1 \ldots n-1$) of a sentence in the training corpus. Each feature is defined as a function of bigrams (we will describe features in detail in next subsection). The instances from all sentences are then combined for training. And Y refers to the class label of the IAS degree. As we mentioned above, segmentation decision is based on IAS value. Therefore, the number of IAS degree's class label is also correspondent to the number of segmentation class label. In traditional segmentation algorithms, they always label segmentation as two classes, segmented and unsegmented. However, for some phrases, it is a dilemma to make a segmentation decision based on this two-class scheme. For example, Chinese phrase "笔记本电脑(Notepad)" can be segmented as "笔记本(Note)" and "电脑(computer)" or can be viewed as one word. We cannot easily classify the gap between "本" and "脑" as segmented or unsegmented. Therefore, beside these two class labels, we define another class label, semisegmented, which means that the gap between two characters could be segmented or unsegmented, either will be right. Correspondingly, IAS degree is also divided into three classes, definitely inseparable (marked as 3), partially inseparable (marked as 2), and separable (marked as 1). "Separable" corresponds to be segmented"; "partially inseparable" corresponds to semisegmented; "definitely inseparable" corresponds to be unsegmented. Obviously, there exists orders between these labels' IAS values, namely $IAS(1) < IAS(2) < IAS(3), IAS(*)$ represents the IAS value of different labels. Next, we will describe the features used to train Ranking SVM model.

3.2 Features for IAS computation

Mutual Information: Mutual information, measuring the relationship between two variables, has been extensively used in computational language research. Given a Chinese character string 'xy' (as mentioned above, in our method, 'xy' refers to bigram in a sentence), mutual information between characters x and y is defined as follows:

$$mi(x, y) = log_2 \frac{p(x, y)}{p(x)p(y)} \tag{2}$$

where $p(x, y)$ is the co-occurrence probability of x and y, namely the probability that bigram 'xy' occurs in the training corpus, and $p(x), p(y)$ are the independent probabilities of x and y respectively. From (2), we conclude that $mi(x, y) \gg 0$ means that IAS is strong; $mi(x, y) \approx 0$ means that it is indefinite for IAS between characters x and y; $mi(x, y) \ll 0$ means that there is no association been characters x and y. However, mutual information has no consideration of context, so it cannot solve the overlapping ambiguity effectively (Sili Wang 2006). To remedy this defect, we introduce another statistics measure, difference of t-test.

Difference of t-score (DTS): Difference of t-score is proposed on the basis of t-score. Given a Chinese character string 'xyz', the t-score of the character y relevant to character x and z is defined

as:

$$t_{x,z}(y) = \frac{p(z|y) - p(y|x)}{\sqrt{\sigma_2(p(z|y)) + \sigma_2(p(y|x))}} \quad (3)$$

where $p(y|x)$ is the conditional probability of y given x, and $p(z|y)$, of z given y, and $\sigma_2(p(y|x))$, $\sigma_2(p(z|y))$ are variances of $p(y|x)$ and of $p(z|y)$ respectively. Sun et al. gave the derivation formula of $\sigma_2(p(y|x)), \sigma_2(p(z|y))$ (Sun et al., 1997) as

$$\sigma_2(p(z|y)) \approx \frac{r(y,z)}{r^2(y)} \quad \sigma_2(p(y|x)) \approx \frac{r(x,y)}{r^2(x)} \quad (4)$$

where $r(x,y)$, $r(y,z)$, $r(y)$, $r(z)$ are the frequency of string xy, yz, y, and z respectively. Thus formula (3) is deducted as

$$t_{x,z}(y) = \frac{\frac{r(y,z)}{r(y)} - \frac{r(x,y)}{r(x)}}{\sqrt{\frac{r(y,z)}{r^2(y)} + \frac{r(x,y)}{r^2(x)}}} \quad (5)$$

$t_{x,z}(y)$ indicates the binding tendency of y in the context of x and z: if $t_{x,z}(y) > 0$ then y tends to be bound with z rather than with x; if $t_{x,z}(y) < 0$, they y tends to be bound with x rather than with z.

To measure the binding tendency between two adjacent characters 'xy' (also, it refers to bigram in a sentence in our method), we use difference of t-score (DTS) (Sun et al., 1998) which is defined as

$$dts(x,y) = t_{v,y}(x) - t_{x,w}(y) \quad (6)$$

Higher $dts(x,y)$ indicates stronger IAS between adjacent characters x and y.

Dictionary Information: Both statistics measures mentioned above cannot avoid sparse data problem. Then Dictionary Information is used to compensate for the shortage of statistics information. The dictionary we used includes 75784 terms. We use binary value to denote the dictionary feature. If a bigram is in the dictionary or a part of dictionary term, we label it as "1", otherwise, we label is as "0".

Frequency: An important characteristic of new word is its repeatability. Thus, we also use frequency as another feature to train Ranking SVM model. Here, the frequency is referred to the number of times that a bigram occurs in the training corpus.

We give a training sentence for a better understanding of features mentioned above. The sentence is "中国建设银行网(China Construction Bank network)" We extract all bigrams in this sentence, compute the four above features and give the IAS a label for each bigram. The feature vectors of all these bigrams for training are shown in Table 1.

Algorithm 1 : Generate various granularity terms

Input: A Chinese sentence $S = C_1 : C_n$
$IAS = IAS_{1:n-1}$ $LB = 1; RB = n$
$Iterative(S, IAS)$:
while $length(S) \geq 3$ **do**
  $MB = FindMinIAS(IAS)$
  $SL = C_{LB:MB}$
  $SR = C_{MB+1:RB}$
  $IAS_L = IAS_{LB:MB}$
  $IAS_R = IAS_{MB+1:RB}$
  $Iterative(SL, IAS_L)$
  $Iterative(SR, IAS_R)$
end while

3.3 Segmentation scheme

In order to compare with other segmentation methods, which give a segmentation result based on two class labels, segmented and unsegmented, it is necessary to convert real numbers result given by Ranking SVM to these two labels. Here, we make a heuristic scheme to segment the sentence based on IAS ranking result predicted by Ranking SVM. The scheme is described in Algorithm 1. In each iteration we cut the sentence at the gap with minimum IAS value. Nie et.al. pointed out that the average length of words in usage is 1.59 (Nie et al., 2000). Therefore, we stop the segmentation iterative when the length of sub_sentence is 2 or less than 2. By this method, we could represent the segmentation result as a binary tree. Figure 1 shows an example of this tree. With this tree, we can obtain various granularity segmentations easily, which could be used in CIR. This segmentation scheme may cause some combinatory ambiguity. However, Nie et.al. (Nie et al., 2000) also pointed out that there is no accurate word definition, thus whether combinatory ambiguity occurs is uncertain. What's more, compared to overlapping ambiguity, combinatory ambiguity is not the fatal factor for information retrieval performance as mentioned in introduction. Therefore, this scheme is reasonable for Chinese information re-

Bigram	MI	DTS	Dictionary	Frequency	IAS
中国(China)	6.67	1985.26	1	1064561	3
国建	2.59	-1447.6	0	14325	1
建设(Construction)	8.67	822.64	1	200129	3
设银	5.94	-844.05	0	16098	2
银行(Bank)	9.22	931.25	1	236976	3
行网	2.29	-471.24	0	15282	1

Table 1: Example of feature vector

江西省交通地图
(Traffic map of JiangXi Province)
江西省 交通地图
(JiangXi Province) (Traffic map)
江西 省 交通 地图
(JiangXi) (Province) (Traffic) (Map)

Figure 1: Example 1

trieval.

4 Experiments and analysis

4.1 Data

Since the label scheme and evaluation measure (described in next subsection) of our segmentation method are both different from the traditional segmentation methods, we did not carry out experiments on SIGHAN. Instead, we used two query logs (QueryLog1 and QueryLog2) as our experiment corpus, which are from two Chinese search engine companies. 900 queries randomly from QueryLog1 were chosen as training corpus. 110 Chinese queries from PKU Tianwang[1] , randomly selected 150 queries from QueryLog1 and 100 queries from QueryLog2 were used as test corpus. The train and test corpus have been tagged by three people. They were given written information need statements, and were asked to judge the IAS of every two adjacent characters in a sentence on a three level scale as mentioned above, separable, partially inseparable, and definitely inseparable. The assessors agreed in 84% of the sentences, the other sentences were checked by all assessors, and a more plausible alternative was selected. We exploited SVM^{light2} as the toolkit to implement Ranking SVM model.

4.2 Evaluation Measure

Since our approach is based on the ranking of IAS values, it is inappropriate to evaluate our method by the traditional method used in other segmentation algorithms. Here, we proposed an evaluation measure RankPrecision based on Kendall's τ (Joachims, 2002), which compared the similarity between the predicted ranking of IAS values and the rankings of these tags as descending order. RankPrecision formula is as follows:

$$RankPrecision = 1 - \frac{\Sigma_{i=1}^{n} InverseCount(s_i)}{\Sigma_{i=1}^{n} CompInverseCount(s_i)} \quad (7)$$

where s_i represents the ith sentence (unsegmented string), $InverseCount(s_i)$ represents the number of discordant pairs inversions in the ranking of the predicted IAS value compared to the correct labeled ranking. $CompInverseCount(s_i)$ represents the number of discordant pairs inversions when the labels totally inverse.

4.3 Experiments Results

Contributions of the Features: We investigated the contribution of each feature by generating many versions of Ranking SVM model. RankPrecision as described above was used for evaluations in these and following experiments. We used Mutual Information(MI); Difference of T-Score(DTS); Frequency(F); mutual information and difference of t-score(MI+DTS); mu-

[1]Title field of SEWM2006 and SEWM2007 web retrieval TD task topics. See http://www.cwirf.org/

[2]http://svmlight.joachims.org/

Feature	Corpus			
	Train	Query Log1	Query Log2	Tian Wang
MI	0.882	0.8719	0.8891	0.9444
DTS	0.9054	0.8954	0.9086	0.9444
F	0.8499	0.8416	0.8563	0.9583
MI+DTS	0.9077	0.9117	0.923	0.9769
MI+DTS+F	0.8896	0.8857	0.9209	0.9815
MI+DTS+D	**0.933**	0.916	**0.9384**	**0.9954**
MI+DTS+F+D	0.932	**0.93**	0.9374	**0.9954**

Table 2: The segmentation performance with different features

Size of Train Corpus	Corpus			
	Train	Query Log1	Query Log2	Tian Wang
100	0.9149	0.9070	0.9209	0.9630
200	0.9325	0.9304	0.9446	0.9907
400	0.9169	0.9057	0.9230	0.9630
500	0.9320	0.9300	0.9374	0.9954
600	0.9106	0.9050	0.9312	0.9907
700	0.9330	0.9284	0.9353	0.9954
900	0.9217	0.9104	0.9240	0.9907

Table 3: The segmentation performance with different size training corpus

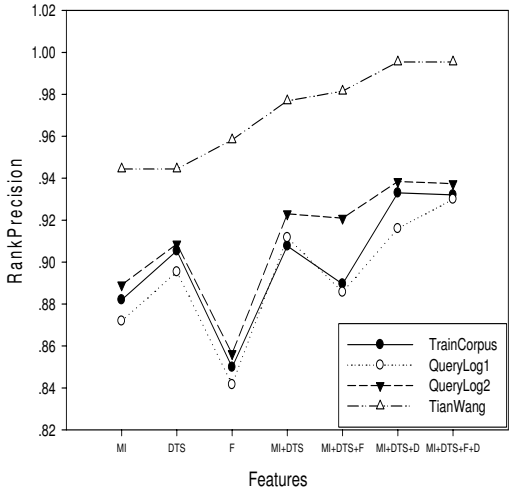

Figure 2: Effects of features

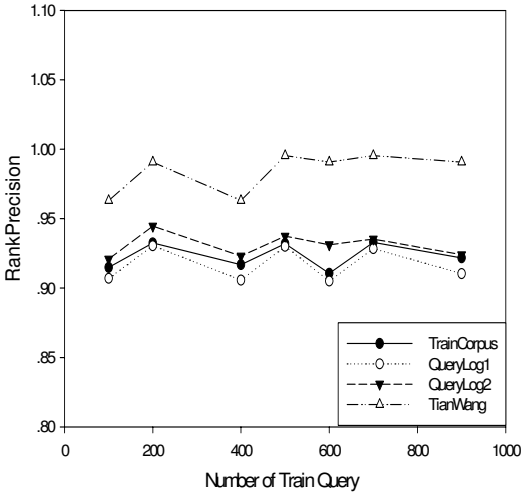

Figure 3: Effects of Corpus Size

tual information, difference of t-score and Frequency(MI+DTS+F); mutual information, difference of t-score and dictionary(MI+DTS+D); mutual information, difference of t-score, frequency and Dictionary(MI+DTS+F+D) as features respectively. The results are shown in Table 2 and Figure 2. From the results, we can see that:

- Using all described features together, the Ranking SVM achieved a good performance. And when we added MI, DTS, frequency, dictionary as features one by one, the RankPrecision improved step by step. It demonstrates that the features we selected are useful for segmentation.
- The lowest RankPrecision is above 85%, which suggests that the predicted rank result by our approach is very close to the right rank. It is shown that our method is effective.
- When we used each feature alone, difference of t-score achieved highest RankPrecise, frequency was worst on most of test corpus (except TianWang). It is induced that difference of t-test is the most effective feature for segmentation. It is explained that because dts is combined with the context information, which eliminates overlapping ambiguity errors.
- It is surprising that when mutual information and difference of t-score was combined with

frequency, the RankPrecision was hurt on three test corpus, even worse than dts feature. The reason is supposed that some non-meaning but common strings, such as "的人" would be took for a word with high IAS values. To correct this error, we could build a stop word list, and when we meet a character in this list, we treat them as a white-space.

Effects of corpus size: We trained different Ranking SVM models with different corpus size to investigate the effects of training corpus size to our method performance. The results are shown in Table 3 and Figure 3. From the results, we can see that the effect of corpus size to the performance of our approach is minors. Our segmentation approach can achieve good performance even with small training corpus, which indicates that Ranking SVM has generalization ability. Therefore we can use a relative small corpus to train Ranking SVM, saving labeling effort.

Effects on Finding Boundary: In algorithm 1, we could get different granularity segmentation words when we chose different length as stop condition. Figure 4 shows the "boundary precision" at each stop condition. Here, "boundary precision" is defined as

$$\frac{No.of\ right\ cut\ boundaries}{No.of\ all\ cut\ boundaries} \quad (8)$$

From the result shown in figure 4, we can see that as the segmentation granularity gets smaller, the boundary precision gets lower. The reason is obvious, that we may segment a whole word into smaller parts. However, as we analyzed in introduction, in CIR, we should judge words boundaries correctly to avoid overlapping ambiguity. As for combinatory ambiguity, through setting different stop length condition, we can obtain different granularity segmentation result.

Effects on Overlapping Ambiguity: Due to the inconsistency of train and test corpus, it is difficult to keep fair for Chinese word segmentation evaluation. Since ICTCLAS is considered as the best Chinese word segmentation systems. We chose ICTCLAS as the comparison object. Moreover, we chose Maximum Match segmentation algorithm, which is rule-based segmentation method, as the baseline.

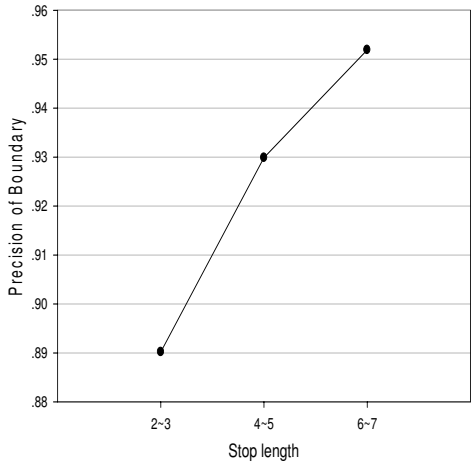

Figure 4: Precision of boundary with different stop word length conditions

Corpus	NOA (RSVM_Seg)	NOA (ICTCLAS)	NOA (MM)
Query Log1	7	10	21
Query Log2	2	6	16
Tian Wang	0	0	1

Table 4: Number of Overlapping Ambiguity

We compared the number of overlapping ambiguity(NOA) among these three approaches on test corpus QueryLog1, QueryLog2 and TianWang. The result is shown in Table 4. On these three test corpus, the NOA of our approach is smallest, which indicates our method resolve overlapping ambiguity more effectively. For example, the sentence "基础课件(basic notes)", the segmentation result of ICTCLAS is "基础课(basic class)/件(article)", the word "课件(notes)" is segmented, overlapping ambiguity occurring. However, with our method, the predicted IAS value rank of positions between every two adjacent characters in this sentence is "基3础1课2件", which indicates that the character "课" has stronger internal associative strength with the character "件" than with the character "础", eliminating overlapping ambiguity according to this ISA rank results.

Effects on Recognition Boundaries of new word: According to the rank result of all IAS values

Figure 5: Example of New Word boundary

in a sentence, our method can recognize the boundaries of new words precisely, avoiding the overlapping ambiguity caused by new words. For example, the phrase "海南中招录取(Hainan High School's Entry Recruitment)", the ICTCLAS segmentation result is "海南/中/招录/取", because the new word "中招" cannot be recognized accurately, thus the character "招" is combined with its latter character "录", causing overlapping ambiguity. By our method, the segmentation result is shown as figure 5, in which no overlapping ambiguity occurs.

Performance of Chinese Information Retrieval: To evaluate the effectiveness of *RSVM-Seg* method on CIR, we compared it with the FMM segmentation. Our retrieval system combines different query representations obtained by our segmentation method, *RSVM-Seg*. In previous TREC Terebyte Track, Markov Random Field(MRF) (Metzler and Croft, 2005) model has displayed better performance than other information retrieval models, and it can much more easily include dependence features. There are three variants of MRF model, full independence(FI), sequential dependence(SD), and full dependence(FD). We chose SD as our retrieval model, since Chinese words are composed by characters and the adjacent characters have strong dependence relationship. We evaluated the CIR performance on the Chinese Web Corpora CWT200g provided by Tianwang [3], which, as we know, is the largest publicly available Chinese web corpus till now. It consists of $37,482,913$ web pages with total size of 197GB. We used the topic set for SEWM2007 and SEWM2006 Topic Distillation (TD) task which contains 121 topics. MAP, R-Precision and GMAP (Robertson, 2006) were as main evaluation metrics. GMAP is the geometric mean of AP(Average Precision) through different queries, which was introduced to concentrate on difficult queries. The result is shown in 5. From the table, we can see that our segmentation method improve the CIR performance compared to FMM.

[3] http://www.cwirf.org/

Segmentation Method	MAP	R-P	GMAP
FMM	0.0548	0.0656	0.0095
RSVM-Seg	0.0623	0.0681	0.0196

Table 5: Evaluation of CIR performance

5 Conclusion and Future work

From what we have discussed above, we can safely draw the conclusion that our work includes several main contributions. Firstly, to our best known, this is the first time to take the Chinese word segmentation problem as ranking problem, which provides a new view for Chinese word segmentation. This approach has been proved to be able to eliminate overlapping ambiguity and also be able to obtain various segmentation granularities. Furthermore, our segmentation method can improve Chinese information retrieval performance to some extent.

As future work, we would search another more encouraging method to make a segmentation decision from the ranking result. Moreover, we will try to relabel SIGHAN corpus on our three labels, and do experiments on them, which will be more convenient to compare with other segmentation methods. Besides, we will carry out more experiments to search the effectiveness of our segmentation method to CIR.

Acknowledgments

This paper is supported by China Natural Science Founding under No. 60603094 and China National 863 key project under No. 2006AA010105. We appreciate Wenbin Jiang's precious modification advices. Finally, we would like to thank the three anonymous EMNLP reviewers for their helpful and constructive comments.

References

D. Fan, W. Bin, and W. Sili. 2007. A Heuristic Approach for Segmentation Granularity Problem in Chinese Information Retrieval. *Advanced Language Processing and Web Information Technology, 2007. ALPIT 2007. Sixth International Conference on*, pages 87–91.

S. Foo and H. Li. 2004. Chinese word segmentation and its effect on information retrieval. volume 40, pages 161–190. Elsevier.

H. Jin and K.F. Wong. 2002. A Chinese dictionary construction algorithm for information retrieval. *ACM Transactions on Asian Language Information Processing (TALIP)*, 1(4):281–296.

T. Joachims. 2002. Optimizing search engines using clickthrough data. *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 133–142.

J.D. Lafferty, A. McCallum, and F.C.N. Pereira. 2001. Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. *Proceedings of the Eighteenth International Conference on Machine Learning table of contents*, pages 282–289.

Q. Liu. 2002. Review of Chinese lexical and syntactic technology.

Z.Y. Luo and R. Song. 2001. Proper noun recognition in Chinese word segmentation research. *Conference of international Chinese computer*, 328:2001–323.

D. Metzler and W.B. Croft. 2005. A Markov random field model for term dependencies. *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 472–479.

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. *Proc of EMNLP*.

J.Y. Nie, J. Gao, J. Zhang, and M. Zhou. 2000. On the use of words and n-grams for Chinese information retrieval. *Proceedings of the fifth international workshop on on Information retrieval with Asian languages*, pages 141–148.

F. Peng, X. Huang, D. Schuurmans, and N. Cercone. 2002a. Investigating the relationship between word segmentation performance and retrieval performance in Chinese IR. *Proceedings of the 19th international conference on Computational linguistics-Volume 1*, pages 1–7.

F. Peng, X. Huang, D. Schuurmans, N. Cercone, and S.E. Robertson. 2002b. Using self-supervised word segmentation in Chinese information retrieval. *Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 349–350.

S. Robertson. 2006. On GMAP: and other transformations. *Proceedings of the 15th ACM international conference on Information and knowledge management*, pages 78–83.

Sili Wang. 2006. Research on chinese word segmentation for large scale information retrieval.

H.P. Zhang, H.K. Yu, D.Y. Xiong, and Q. Liu. 2003. HHMM-based Chinese Lexical Analyzer ICTCLAS. *Proceedings of Second SIGHAN Workshop on Chinese Language Processing*, pages 184–187.