Investigating the Relationship between Word Segmentation Performance and Retrieval Performance in Chinese IR

Fuchun Peng and Xiangji Huang and Dale Schuurmans and Nick Cercone
School of Computer Science, University of Waterloo
200 University Ave. West, Waterloo, Ontario, Canada, N2L 3G1
{f3peng, jhuang, dale, ncercone}@uwaterloo.ca

Abstract

It is commonly believed that word segmentation accuracy is monotonically related to retrieval performance in Chinese information retrieval. In this paper we show that, for Chinese, the relationship between segmentation and retrieval performance is in fact *nonmonotonic*; that is, at around 70% word segmentation accuracy an over-segmentation phenomenon begins to occur which leads to a reduction in information retrieval performance. We demonstrate this effect by presenting an empirical investigation of information retrieval on Chinese TREC data, using a wide variety of word segmentation algorithms with word segmentation accuracies ranging from 44% to 95%. It appears that the main reason for the drop in retrieval performance is that correct compounds and collocations are preserved by accurate segmenters, while they are broken up by less accurate (but reasonable) segmenters, to a surprising advantage. This suggests that words themselves might be too broad a notion to conveniently capture the general semantic meaning of Chinese text.

1 Introduction

Automated processing of written languages such as Chinese involves an inherent word segmentation problem that is not present in western languages like English. Unlike English, Chinese words are not explicitly delimited by whitespace, and therefore to perform automated text processing tasks (such as information retrieval) one normally has to first segment the text collection. Typically this involves segmenting the text into individual words. Although the text segmentation problem in Chinese has been heavily investigated recently (Brent and Tao, 2001; Chang, 1997; Ge et al., 1999; Hockenmaier and Brew, 1998; Jin, 1992; Peng and Schuurmans, 2001; Sproat and Shih, 1990; Teahan et al, 2001) most research has focused on the problem of segmenting character strings into individual words, rather than useful constituents. However, we have found that focusing exclusively on words may not lead to the most effective segmentation from the perspective of broad semantic analysis (Peng et al, 2002).

In this paper we will focus on a simple form of semantic text processing: information retrieval (IR). Although information retrieval does not require a deep semantic analysis, to perform effective retrieval one still has to accurately capture the main topic of discourse and relate this to a given query. In the context of Chinese, information retrieval is complicated by the fact that the words in the source text (and perhaps even the query) are not separated by whitespace. This creates a significant amount of additional ambiguity in interpreting sentences and identifying the underlying topic of discourse.

There are two standard approaches to information retrieval in Chinese text: character based and word based. It is usually thought that word based approaches should be superior, even though character based methods are simpler and more commonly used (Huang and Robertson, 2000). However, there has been recent interest in the word based approach, motivated by recent advances in automatic segmentation of Chinese text (Nie et al, 1996; Wu and Tseng, 1993). A common presumption is that word segmentation accuracy should monotonically influence subsequent retrieval performance (Palmer and Burger, 1997). Consequently, many researchers have focused on producing accurate word segmenters for Chinese text indexing (Teahan et al, 2001; Brent and Tao, 2001). However, we have recently observed that low accuracy word segmenters often yield superior retrieval performance (Peng et al, 2002). This observation was initially a surprise, and motivated us to conduct a more thorough study of the phenomenon to uncover the reason for the performance decrease.

The relationship between Chinese word segmentation accuracy and information retrieval performance has recently been investigated in the literature. Foo and Li (2001) have conducted a series of experiments which suggests that the word segmentation approach does indeed have effect on IR performance. Specifically, they observe that the recognition of words of length two or more can produce better retrieval performance, and the existence of ambiguous words resulting from the word segmentation process can decrease retrieval performance. Similarly, Palmer and

Burger (1997) observe that accurate segmentation tends to improve retrieval performance. All of this previous research has indicated that there is indeed some sort of correlation between word segmentation performance and retrieval performance. However, the nature of this correlation is not well understood, and previous research uniformly suggests that this relationship is monotonic.

One reason why the relationship between segmentation and retrieval performance has not been well understood is that previous investigators have not considered using a variety of Chinese word segmenters which exhibit a wide range of segmentation accuracies, from low to high. In this paper, we employ three families of Chinese word segmentation algorithms from the recent literature. The first technique we employed was the standard maximum matching dictionary based approach. The remaining two algorithms were selected because they can both be altered by simple parameter settings to obtain different word segmentation accuracies. Specifically, the second Chinese word segmenter we investigated was the minimum description length algorithm of Teahan et al. (2001), and the third was the EM based technique of Peng and Schuurmans (2001). Overall, these segmenters demonstrate word identification accuracies ranging from 44% to 95% on the PH corpus (Brent and Tao, 2001; Hockenmaier and Brew, 1998; Teahan et al, 2001).

Below we first describe the segmentation algorithms we used, and then discuss the information retrieval environment considered (in Sections 2 and 3 respectively). Section 4 then reports on the outcome of our experiments on Chinese TREC data, and in Section 5 we attempt to determine the reason for the over-segmentation phenomenon witnessed.

2 Word Segmentation Algorithms

Chinese word segmentation has been extensively researched. However, in Chinese information retrieval the most common tokenziation methods are still the simple character based approach and dictionary-based word segmentation. In the character based approach sentences are tokenized simply by taking each character to be a basic unit. In the dictionary based approach, on the other hand, one pre-defines a lexicon containing a large number of words and then uses heuristic methods such as maximum matching to segment sentences. Below we experiment with these standard methods, but in addition employ two recently proposed segmentation algorithms that allow some control of how accurately words are segmented. The details of these algorithms can be found in the given references. For the sake of completeness we briefly describe the basic approaches here.

2.1 Dictionary based word segmentation

The dictionary based approach is the most popular Chinese word segmentation method. The idea is to use a hand built dictionary of words, compound words, and phrases to index the text. In our experiments we used the longest forward match method in which text is scanned sequentially and the longest matching word from the dictionary is taken at each successive location. The longest matched strings are then taken as indexing tokens and shorter tokens within the longest matched strings are discarded. In our experiments we used two different dictionaries. The first is the Chinese dictionary used by Gey et al. (1997), which includes 137,659 entries. The second is the Chinese dictionary used by Beaulieu et al. (1997), which contains 69,353 words and phrases.

2.2 Compression based word segmentation

The PPM word segmentation algorithm of Teahan et al. (2001) is based on the text compression method of Cleary and Witten (1984). PPM learns an n-gram language model by supervised training on a given set of hand segmented Chinese text. To segment a new sentence, PPM seeks the segmentation which gives the best compression using the learned model. This has been proven to be a highly accurate segmenter (Teahan et al, 2001). Its quality is affected both by the amount of training data and by the order of the n-gram model. By controlling the amount of training data and the order of language model we can control the resulting word segmentation accuracy.

2.3 EM based word segmentation

The “self-supervised” segmenter of Peng and Schuurmans (2001) is an unsupervised technique based on a variant of the EM algorithm. This method learns a hidden Markov model of Chinese words, and then segments sentences using the Viterbi algorithm (Rabiner, 1989). It uses a heuristic technique to reduce the size of the learned lexicon and prevent the acquisition of erroneous word agglomerations. Although the segmentation accuracy of this unsupervised method is not as high as the supervised PPM algorithm, it nevertheless obtains reasonable performance and provides a fundamentally different segmentation scheme from PPM. The segmentation performance of this technique can be controlled by varying the number of training iterations and by applying different lexicon pruning techniques.

3 Information Retrieval Method

We conducted our information retrieval experiments using the OKAPI system (Huang and Robertson, 2000; Robertson et al., 1994). In an attempt to ensure that the phenomena we observe are not specific to a particular retrieval technique, we experimented with a parameterized term weighting scheme which allowed us to control the quality of retrieval performance. We considered a refined term weighting scheme based on the the standard term weighting function

$$w_0 = log\frac{N - n + 0.5}{n + 0.5} \tag{1}$$

where N is the number of indexed documents in the collection, and n is the number of documents containing a specific term (Spark Jones, 1979). Many researchers have shown that augmenting this basic function to take into account document length, as well as within-document and within-query frequencies, can be highly beneficial in English text retrieval (Beaulieu et al., 1997). For example, one standard augmentation is to use

$$w_1 = w_0 * \frac{(c_1 + 1) * \mathit{tf}}{K + \mathit{tf}} * \frac{(c_2 + 1) * \mathit{qtf}}{c_2 + \mathit{qtf}} \tag{2}$$

where

$$K = c_1 * \left(1 - c_3 + c_3 \frac{dl}{avdl}\right)$$

Here *tf* is within-document term frequency, *qtf* is within-query term frequency, *dl* is the length of the document, *avdl* is the average document length, and c_1, c_2, c_3 are tuning constants that depend on the database, the nature of the queries, and are empirically determined. However, to truly achieve state-of-the-art retrieval performance, and also to allow for the quality of retrieval to be manipulated, we further augmented this standard term weighting scheme with an extra correction term

$$w_2 = w_1 \;\oplus\; k_d * y \tag{3}$$

This correction allows us to more accurately account for the length of the document. Here $\oplus$ indicates that the component is added only once per document, rather than for each term, and

$$y = \begin{cases} \ln(\frac{dl}{avdl}) + \ln(c_4) & \text{if } dl \leq \mathit{rel\_avdl} \\ \left(\ln(\frac{\mathit{rel\_avdl}}{avdl}) + \ln(c_4)\right)\left(1 - \frac{dl - \mathit{rel\_avdl}}{c_5 * avdl - \mathit{rel\_avdl}}\right) & \\ & \text{if } dl > \mathit{rel\_avdl} \end{cases}$$

where *rel_avdl* is the average relevant document length calculated from previous queries based on the same collection of documents. Overall, this term weighting formula has five tuning constants, c_1 to c_5, which are all set from previous research on English text retrieval and some initial experiments on Chinese text retrieval. In our experiments, the values of the five arbitrary constants c_1, c_2, c_3, c_4 and c_5 were set to 2.0, 5.0, 0.75, 3 and 26 respectively.

The key constant is the quantity k_d, which is the new tuning constant that we manipulate to control the influence of correction factor, and hence control the retrieval quality. By setting k_d to different values, we have different term weighting methods in our experiments. In our experiments, we tested k_d set to values of 0, 6, 8, 10, 15, 20, 50.

4 Experiments

We conducted a series of experiments in word based Chinese information retrieval, where we varied both the word segmentation method and the information retrieval method. We experimented with word segmentation techniques of varying accuracy, and information retrieval methods with varying performance. In almost every case, we witness a nonmonotonic relationship between word segmentation accuracy and retrieval performance, robustly across retrieval methods. Before describing the experimental results in detail however, we first have to describe the performance measures used in the experiments.

4.1 Measuring segmentation performance

We evaluated segmentation performance on the Mandarin Chinese corpus, PH, due to Guo Jin. This corpus contains one million words of *segmented* Chinese text from newspaper stories of the *Xinhua news agency of the People's Republic of China* published between January 1990 and March 1991.

To make the definitions precise, first define the original *segmented* test corpus to be S. We then collapse all the whitespace between words to make a second unsegmented corpus U, and then use the segmenter to recover an estimate $\hat{S}$ of the original segmented corpus. We measure the segmentation performance by precision, recall, and F-measure on detecting correct words. Here, a word is considered to be correctly recovered if and only if (Palmer and Burger, 1997)

1. a boundary is correctly placed in front of the first character of the word
2. a boundary is correctly placed at the end of the last character of the word
3. and there is no boundary between the first and last character of the word.

Let N_1 denote the number of words in S, let N_2 denote the number of words in the estimated segmentation $\hat{S}$, and let N_3 denote the number of words correctly recovered. Then the precision, recall and F measures are defined

$$\text{precision: } p = \tfrac{N_3}{N_2}$$
$$\text{recall: } r = \tfrac{N_3}{N_1}$$
$$\text{F-measure: } F = \tfrac{2 \times p \times r}{p + r}$$

In this paper, we only report the performance in F-measure, which is a comprehensive measure that combines precision and the recall.

4.2 Measuring retrieval performance

We used the TREC relevance judgments for each topic that came from the human assessors of the National Institute of Standards and Technology (NIST). Our statistical evaluation was done by means of the TREC evaluation program. The measures we report are *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. Detailed descriptions of these measures can be found in (Voorhees and Harman, 1998).

4.3 Data sets

We used the information retrieval test collections from TREC-5 and TREC-6 (Voorhees and Harman, 1998). (Note that the document collection used in the TREC-6 Chinese track was identical to the one used in TREC-5, however the topic queries differ.) This collection of Chinese text consists of 164,768 documents and 139,801 articles selected from the *People's Daily* newspaper, and 24,988 articles selected from the *Xinhua newswire*. The original articles are tagged in SGML, and the Chinese characters in these articles are encoded using the GB (Guo-Biao) coding scheme. Here 0 bytes is the minimum file size, 294,056 bytes is the maximum size, and 891 bytes is the average file size.

To provide test queries for our experiments, we considered the 54 Chinese topics provided as part of the TREC-5 and TREC-6 evaluations (28 for TREC-5 and 26 for TREC-6).

Finally, for the two learning-based segmentation algorithms, we used two separate training corpora but a common test corpus to evaluate segmentation accuracy. For the PPM segmenter we used 72% of the PH corpus as training data. For the the self-supervised segmenter we used 10M of data from the data set used in (Ge et al., 1999), which contains one year of *People's Daily* news service stories. We used the entire PH collection as the test corpus (which gives an unfair advantage to the supervised method PPM which is trained on most of the same data).

4.4 Segmentation accuracy control

By using the forward maximum matching segmentation strategy with the two dictionaries, Berkeley and City, we obtain the segmentation performance of 71% and 85% respectively. For the PPM algorithm, by controlling the order of the n-gram language model used (specifically, order 2 and order 3) we obtain segmenters that achieve 90% and 95% word recognition accuracy respectively. Finally, for the self-supervised learning technique, by controlling the number of EM iterations and altering the lexicon pruning strategy we obtain word segmentation accuracies of 44%, 49%, 53%, 56%, 59%, 70%, 75%, and 77%. Thus, overall we obtain 12 different segmenters that achieve segmentation performances of 44%, 49%, 53%, 56%, 59%, 70%, 71%, 75%, 77%, 85%, 90%, and 95%.

4.5 Experimental results

Now, given the 12 different segmenters, we conducted extensive experiments on the TREC data sets using different information retrieval methods (achieved by tuning the k_d constant in the term weighting function described in Section 3).

Table 1 shows the *average precision* and *R-precision* results obtained on the TREC-5 and TREC-6 queries when basing retrieval on word segmentations at 12 different accuracies, for a single retrieval method, $k_d = 10$. To illustrate the results graphically, we re-plot this data in Figure 1, in which the x-axis is the segmentation performance and the y-axis is the retrieval performance.

seg. accuracy	TREC-5	TREC-6
44%	0.2231/0.2843	0.3424/0.3930
49%	0.2647/0.3259	0.3848/0.4201
53%	0.2999/0.3376	0.4492/0.4801
56%	0.3056/0.3462	0.4473/0.4727
59%	0.3097/0.3533	0.4740/0.4960
70%	0.3721/0.3988	0.5044/0.5072
71%	0.3656/0.4088	0.5133/0.5116
75%	0.3652/0.4000	0.4987/0.5097
77%	0.3661/0.4027	0.4968/0.4973
85%	0.3488/0.3898	0.5049/0.5047
90%	0.3213/0.3663	0.4983/0.5008
95%	0.3189/0.3669	0.4867/0.4933

Table 1: Average precision and R-precision results on TREC queries when $k_d = 10$.

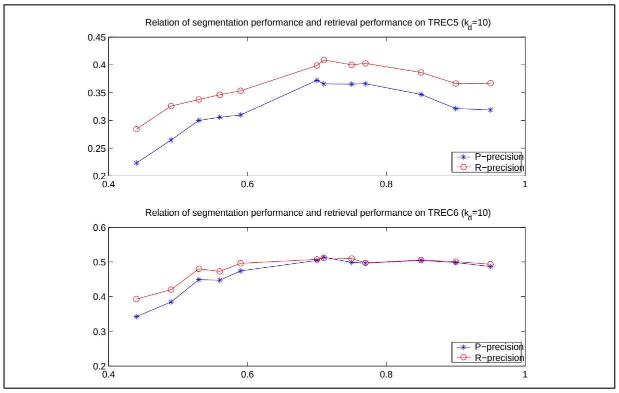

Figure 1: Retrieval F-measure (y-axis) versus segmentation accuracy (x-axis) for $k_d = 10$.

Clearly these curves demonstrate a nonmonotonic relationship between retrieval performance (on the both P-precision and the R-precision) and segmentation accuracy. In fact, the curves show a clear uni-modal shape, where for segmentation accuracies 44% to 70% the retrieval performance increases steadily, but then plateaus for segmentation accuracies between 70% and 77%, and finally decreases slightly when the segmentation performance increase to 85%,90% and 95%.

This phenomenon is robustly observed as we alter the retrieval method by setting $k_d = 0, 6, 8, 15, 20, 50$, as shown in Figures 2 to 7 respectively.

To give a more detailed picture of the results, Figures 8 and 9 we illustrate the full *precision-recall* curves for $k_d = 10$ at each of the 12 segmentation accuracies, for TREC-5 and TREC-6 queries respectively. In these figures, the 44%, 49% segmentations are marked with stars, the 53%, 56%, 59% segmentations are marked with circles, the 70%, 71%, 75%, 77% segmentations are marked with diamonds, the 85% segmentation is marked with hexagrams, and the 90% and 95% segmentations are marked with triangles. We can see that the curves with the diamonds are above the others, while the curves with stars are at the lowest positions.

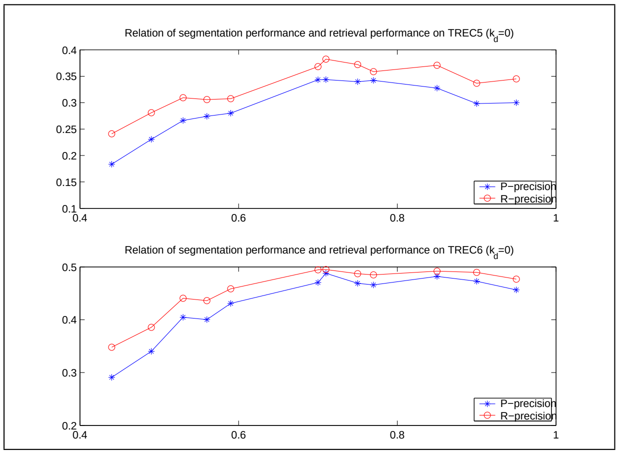

Figure 2: Results for $k_d = 0$.

5 Discussion

The observations were surprising to us at first, although they suggest that there is an interesting phenomenon at work. To attempt to identify the underlying cause, we break the explanation into two parts: one for the first part of the curves where retrieval performance increases with increasing segmentation accuracy, and a second effect for the region where retrieval performance plateaus and eventually decreases with increasing segmentation accuracy.

The first part of these performance curves seems easy to explain. At low segmentation accuracies the segmented tokens do not correspond to meaningful linguistic terms, such as words, which hampers retrieval performance because the term weighting procedure is comparing arbitrary tokens to the query. However, as segmentation accuracy improves, the tokens behave more like true words and the retrieval engine begins to behave more conventionally.

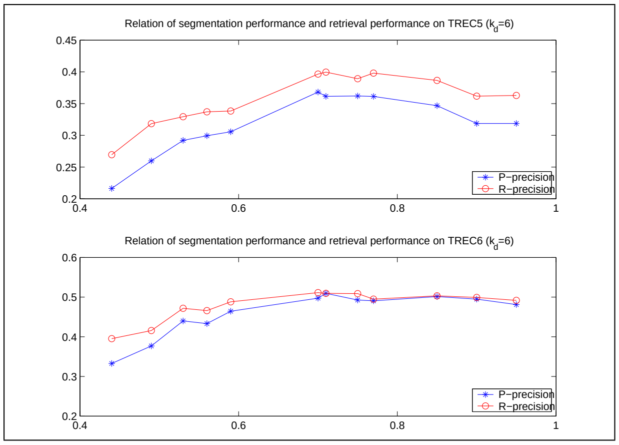

Figure 3: Results for $k_d = 6$.

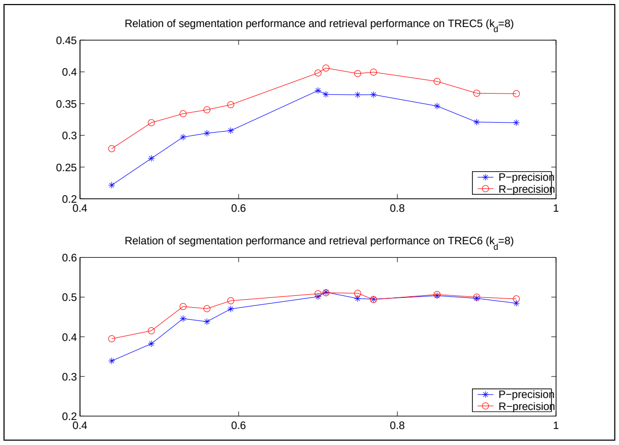

Figure 4: Results for $k_d = 8$.

However, after a point, when the second regime is reached, retrieval performance no longer increases with improved segmentation accuracy, and eventually begins to decrease. One possible explanation for this which we have found is that a weak word segmenter accidentally *breaks* compound words into smaller constituents, and this, surprisingly yields a beneficial effect for Chinese information retrieval.

For example, one of the test queries, Topic 34, is about the impact of droughts in various parts of China. Retrieval based on the EM-70% segmenter retrieved 84 of 95 relevant documents in the collection, whereas retrieval based on the PPM-95% segmenter retrieved only 52 relevant documents. In fact, only 2 relevant documents were missed by EM-70% but retrieved by PPM-95%, whereas 34 docu-

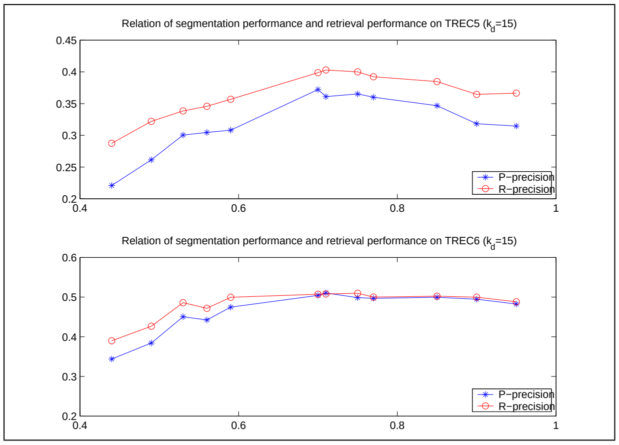

Figure 5: Results for $k_d = 15$.

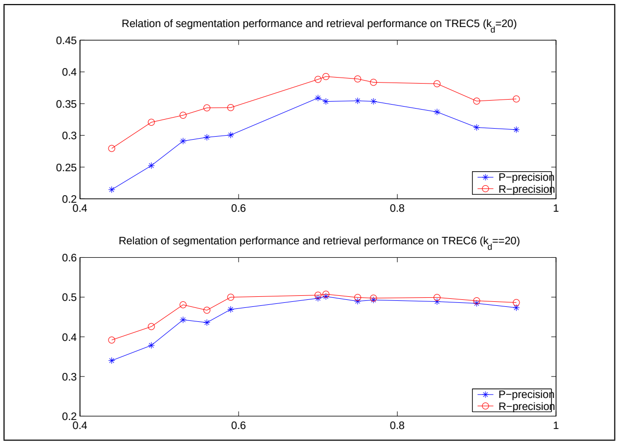

Figure 6: Results for $k_d = 20$.

ments retrieved by EM-70% and not by PPM-95%. In investigating this phenomenon, one finds that the performance drop appears to be due to the inherent nature of written Chinese. That is, in written Chinese many words can often legally be represented their subparts. For example, 农作物(agriculture plants) is sometimes represented as 作物(plants). So for example in Topic 34, the PPM-95% segmenter correctly segments 旱灾 as 旱灾(drought disaster) and 农作物 correctly as 农作物 (agriculture plants), whereas the EM-70% segmenter incorrectly segments 旱灾 as 旱(drought) and 灾(disaster), and incorrectly segments 农作物 as 农(agriculture) and 作物(plants). However, by inspecting the relevant documents for Topic 34, we find that there are many Chinese character strings in these documents that are closely related to the correctly segmented word 旱灾(drought disaster). These alternative words are 春旱，旱魔，受旱，干旱，抗旱，旱区 etc. For example, in the relevant document “pd9105-832”, which is ranked 60th by EM-70% and 823rd by PPM-95%, the correctly segmented word 旱灾 does

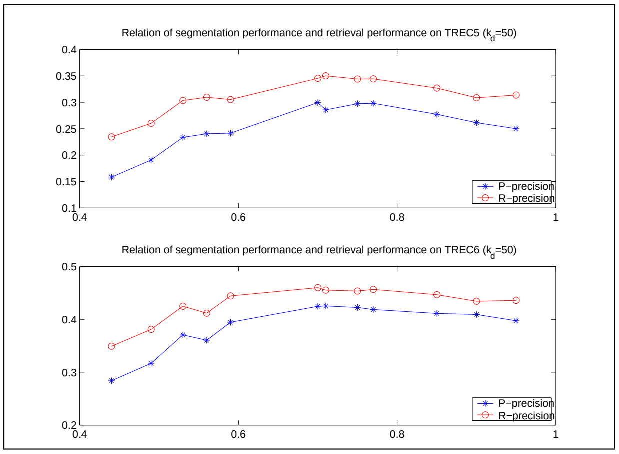

Figure 7: Results for $k_d = 50$.

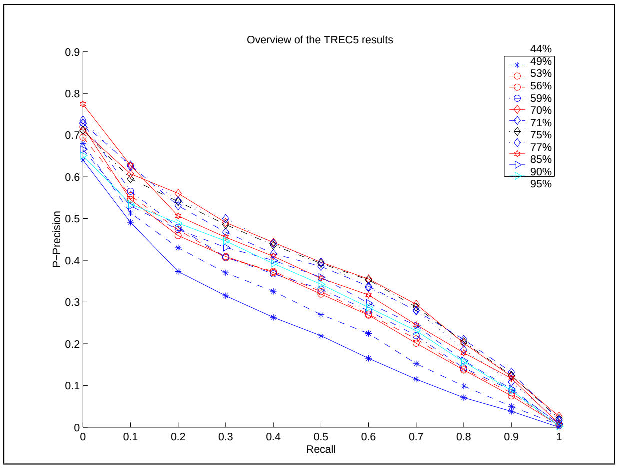

Figure 8: TREC5 precision-recall comprehensive view at $k_d = 10$

not appear at all. Consequently, the correct segmentation for 旱灾 by PPM-95% leads to a much weaker match than the incorrect segmentation of EM-70%. Here EM-70% segments 旱灾 into 旱 and 灾 , which is not regarded as a correct segmentation. However, there are many matches between the topic and relevant documents which contain only 旱. This same phenomenon happens with the query word 农作物 since many documents only contain the fragment 作物 instead of 农作物, and these documents are all missed by PPM-95% but captured by EM-70%.

Although straightforward, these observations suggest a different trajectory for future research on Chinese information retrieval. Instead of focusing on achieving accurate word segmentation, we should pay more attention to issues such as keyword weighting (Huang and Robertson, 2000) and query keyword extraction (Chien et al, 1997). Also, we find that the weak unsupervised segmentation method

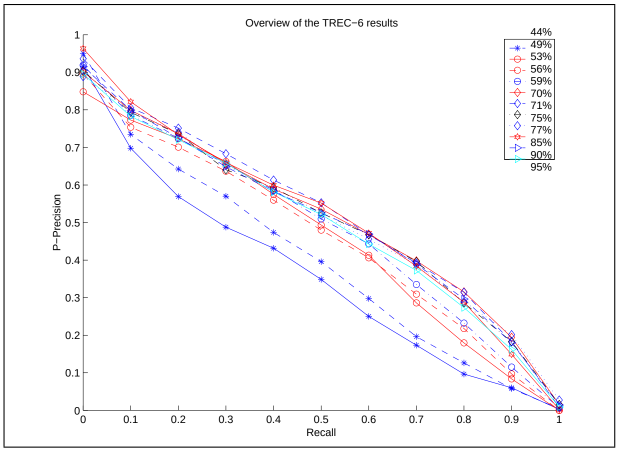

Figure 9: TREC6 precision-recall comprehensive view at $k_d = 10$

based yields better Chinese retrieval performance than the other approaches, which suggests a promising new avenue to apply machine learning techniques to IR (Sparck Jones, 1991). Of course, despite these results we expect highly accurate word segmentation to still play an important role in other Chinese information processing tasks such as information extraction and machine translation. This suggests that some different evaluation standards for Chinese word segmentation should be given to different NLP applications.

6 Acknowledgments

Research supported by Bell University Labs, MITACS and NSERC. We sincerely thank Dr. William Teahan for supplying us the PPM segmenters.

References

Beaulieu, M. and Gatford, M. and Huang, X. and Robertson, S. and Walker, S. and Williams, P. 1997. Okapi at TREC-5. In *Proceedings TREC-5*.

Brent, M. and Tao, X. 2001, Chinese Text Segmentation With MBDP-1: Making the Most of Training Corpora. In *Proceedings ACL-2001*.

Buckley, C., Singhal, A., and Mitra, M. 1997. Using query zoning and correlation within SMART: TREC-5. In *Proceedings TREC-5*.

Chang, J.-S. and Su, K.-Y. 1997, An Unsupervised Iterative Method for Chinese New Lexicon Extraction, In *Int J Comp Ling & Chinese Lang Proc.*

Chen, A. and He, J. and Xu, L. and Gey, F. and Meggs, J. 1997. Chinese Text Retrieval Without Using a Dictionary. In *Proceedings SIGIR-97.*

Chien L. and Huang, T. and Chien, M. 1997 In *Proceedings SIGIR-97.*

Cleary, J. and Witten, I. 1984. Data compression using adaptive coding and partial string matching. In *IEEE Trans Communications*, 32(4): 396-402.

Foo, S. and Li, H. 2001 Chinese Word Segmentation Accuracy and Its Effects on Information Retrieval. In *TEXT Technology.*

Ge, X., Pratt, W. and Smyth, P. 1999. Discovering Chinese Words from Unsegmented Text. In *Proceedings SIGIR-99.*

Gey, F., Chen, A., He, J., Xu, L. and Meggs, J. 1997 Term Importance, Boolean Conjunct Trainning Negative Terms, and Foreign Language Retrieval: Probabilistic Algorithms at TREC-5. In *Proceedings TREC-5.*

Hockenmaier, J. and Brew C. 1998. Error driven segmentation of Chinese. In *Comm. COLIPS*, 8(1): 69-84.

Huang, X. and Robertson, S. 2000. A probabilistic approach to Chinese information retrieval: theory and experiments. In *Proceedings BCS-IRSG 2000.*

Jin, W. 1992, Chinese Segmentation and its Disambiguation, Tech report, New Mexico State Univ.

Nie, J., Brisebois, M. and Ren, X. 1996. On Chinese text retrieval. In *Proceedings SIGIR-96.*

Palmer, D. and Burger, J. 1997. Chinese Word Segmentation and Information Retrieval. In *AAAI Symp Cross-Language Text and Speech Retrieval.*

Peng, F., Huang, X., Schuurmans, D., Cercone, N., and Robertson, S. 2002. Using Self-supervised Word Segmentation in Chinese Information Retrieval. In *Proceedings SIGIR-02.*

Peng, F. and Schuurmans, D. 2001. Self-supervised Chinese Word Segmentation. In *Proceedings IDA-01*, LNCS 2189.

Rabiner, L. 1989. A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition. In *Proceedings of IEEE*, 77(2).

Robertson, S. and Walker, S. 1994. Some Simple Effective Approximations to the 2-Poisson Model for Probabilistic Weighted Retrieval. *SIGIR-94.*

Sparck Jones, K. 1991 The Role of Artificial Intelligence in Information Retrieval *J. Amer. Soc. Info. Sci.*, 42(8): 558-565.

Sparck Jones, K. 1979. Search Relevance Weighting Given Little Relevance Information. In *J. of Documentation*, 35(1).

Sproat, R. and Shih, C. 1990. A Statistical Method for Finding Word Boundaries in Chinese Text, In *Comp Proc of Chinese and Oriental Lang*, 4.

Teahan, W. J. and Wen, Y. and McNab, R. and Witten I. H. 2001 A Compression-based Algorithm for Chinese Word Segmentation. In *Comput. Ling.*, 26(3):375-393.

Voorhees, E. and Harman, D. 1998. Overview of the Sixth Text REtrieval Conference (TREC-6), In *Proceedings TREC-6.*

Wu, Z. and Tseng, G. 1993. Chinese Text Segmentation for Text Retrieval: Achievements and Problems. In *J. Amer. Soc. Info. Sci.*, 44(9): 532-542.