Chinese Word Segmentation and Information Retrieval

David D. Palmer and John D. Burger
The MITRE Corporation
202 Burlington Road
Bedford, MA 01730, USA
{palmer,john}@mitre.org

Abstract

In this paper we present results of experiments with Chinese word segmentation and information retrieval. Our experiments with three different word segmentation algorithms indicate that accurate segmentation measurably improves retrieval performance. We discuss the evaluation of word segmentation algorithms for the purpose of better indexing segmented texts for retrieval.

Introduction

The increased interest in crosslingual and multilingual information retrieval has revealed the new challenges inherent in retrieval in multiple languages. English IR has been extensively engineered for 30 years, with the development of stop lists, stemming, etc., but such resources are not available for many languages. Recent Text Retrieval Conferences (TREC-4 (Harman 1996b) and TREC-5 (Harman 1996a)) have included separate tracks for evaluation of retrieval in languages such as Spanish and Chinese, and a crosslingual track involving English, French, and German is planned for TREC-6.

Traditionally, information retrieval systems index words (or word stems), usually tokenizing words in a collection based on white space. However, unsegmented languages such as Chinese and Thai are written without explicit word boundaries. IR systems therefore must either index each individual character or be provided with information about word boundaries. The word boundaries cannot be determined with complete accuracy, and it is unclear how errors in word segmentation degrade IR performance.

At the recent TREC-5 conference, (Buckley, Singhal, & Mitra 1996) reported excellent retrieval results in the Chinese track using simply the character-as-word segmentation algorithm discussed below. This led them to suggest that "segmentation is a minor issue for retrieving Chinese and shouldn't be a major focus." However, while (Broglio, Callan, & Croft 1996) also report respectable results on Chinese retrieval using this simple segmentation algorithm, they show that their automatic HMM-based segmentation algorithm improves retrieval performance by 10%.

In this paper we present empirical evidence for the contribution of word segmentation to Chinese retrieval. Our experiments with three different segmentation algorithms indicate that word segmentation beyond the character-as-word algorithm measurably improves retrieval performance.

Segmentation algorithms

For our experiments with Chinese segmentation, we used the corpus from the TREC-5 Chinese track, which consisted of 170MB of texts from the Xinhua news service and People's Daily. Our baseline IR system was SMART, a publically-available vector-space system developed at Cornell. We made minor modifications to SMART to enable the system to process the extended character set used in the Chinese collection; these changes are similar to those described in Cornell's TREC-5 paper (Buckley, Singhal, & Mitra 1996).

For each experiment, we segmented the entire Chinese collection using a word segmentation algorithm and indexed the collection using SMART. In each case, the queries were also segmented using the same algorithm used to segment the collection[1].

Character-as-word

A simple initial segmentation for Chinese is to consider each character a distinct word, and this segmentation algorithm has been used to obtain excellent Chinese information retrieval performance (Buckley, Singhal, & Mitra 1996; Broglio, Callan, & Croft 1996). Such an algorithm is successful because the Chinese character set contains just a few thousand distinct charac-

[1] Predictably, experiments in which we segmented the queries using a different algorithm from that used to segment the collection each resulted in dismal retrieval performance.

ters, and most Chinese words consist of just one or two characters[2].

Since the character-as-word algorithm is the simplest segmentation algorithm and also produces good retrieval results, we will treat it as the baseline algorithm for determining the improvement provided by other segmentation algorithms. By doing this, we can investigate the effects of various segmentations on retrieval performance and directly compare the results.

Maximum matching

A very common approach to word segmentation is to use a variation of the *maximum matching* algorithm, commonly referred to as the "greedy algorithm." The greedy algorithm starts at the first character in a text and, using a word list for the language being segmented, attempts to find the longest word in the list starting with that character. If a word is found, the maximum-matching algorithm marks a boundary at the end of the longest word, then begins the same longest match search starting at the character following the match. If no match is found in the word list, the greedy algorithm simply segments that character as a word (as in the character-as-word algorithm above) and begins the search starting at the next character.[3] In this manner, an initial segmentation can be obtained that is more informed than a simple character-as-word approach. In our experiments applying the greedy algorithm, we used a list of 57472 Chinese words from the NMSU CHSEG segmenter (described in the next section).

NMSU segmenter

The Chinese segmenter CHSEG developed at the Computing Research Laboratory at New Mexico State University is a complete system for high-accuracy Chinese segmentation (Jin 1994). In addition to an initial segmentation module that finds words in a text based on a list of Chinese words, CHSEG additionally contains specific modules for recognizing idiomatic expressions, derived words, Chinese person names, and foreign proper names.

Scoring Word Segmentation

In order to evaluate the above segmentation algorithms for segmentation accuracy, we used a set of 2000 hand-segmented sentences (60187 words) from a corpus of Xinhua independent of the TREC collection. We evaluated each segmentation algorithm using three different common "scoring" algorithms.

Binary decision

In this simple scoring algorithm, the basic assumption is that in segmenting a string of text, a segmenter must make a boundary-placement decision after each character. The number of correct binary decisions divided by the total number of characters is the Binary Decision score (a percentage).

Boundary recall/precision

This algorithm combines the Binary Decision algorithm with the IR notions of recall and precision. Recall (R) is defined as the percentage of correct boundaries identified, while precision (P) is defined as the percentage of identified boundaries which are correct. The component recall and precision scores are then used to calculate a balanced *F-measure* or *F-score* (Rijsbergen 1979), where $F = 2PR/(P + R)$.

Word recall/precision

This algorithm works on the assumption that identifying complete words, rather than placing the most correct boundaries, is the main goal of segmentation. Word recall/precision is a much stricter measure than boundary recall/precision, in that a word is only correctly segmented if three conditions are met:

1. A boundary is correctly placed in front of the first character

2. A boundary is correctly placed after the last character

3. No boundary is placed between the first and last characters

In scoring segmentation using this algorithm, recall is defined as the percentage of actual words from the hand-segmented text identified in the corresponding positions in the text, while precision is defined as the percentage of identified words which are also in the same positions in the hand-segmented text. A word F-measure is then calculated based on the word recall and precision scores.

By each of these three scoring algorithms, the NMSU segmenter produced the best segmentation, consistently scoring better than the maximum matching segmentation and significantly better than the character-as-word segmentation. Table 1 shows a summary of the evaluation of the segmentation algorithms using these three metrics.

[2] We determined the average length of a word in a portion of our Chinese data to be 1.60 characters.

[3] A variation of the greedy algorithm segments a sequence of unmatched characters as a single word, but this is less accurate for Chinese, due to the short average word length.

Segmentation Algorithm	Binary Decision	Boundary F-score	Word F-score
Character-as-word	59.7	76.7	40.3
Max. matching	86.1	91.8	82.7
NMSU segmenter	89.6	94.7	86.9

Table 1: Summary of segmentation scores.

Retrieval results

Table 2 shows a summary of the Chinese experiments, which involved running SMART on the entire text collection segmented using each of the three segmentation algorithms (character-as-word, maximum matching, and NMSU CHSEG). The retrieval results are given in the familiar TREC metrics of Average Precision and R-precision. Also shown is the percentage improvement in retrieval performance the NMSU segmenter and maximum matching algorithm produced over the character-as-word segmentation.

Initial Algorithm	Average Precision	R-Precision
Character-as-word	.0817	.1450
Max. matching	.1071 (+31.1%)	.1753 (+20.9%)
NMSU segmenter	.0951 (+16.4%)	.1631 (+12.5%)

Table 2: Chinese results on TREC-5 collection (28 queries).

Interestingly, the best retrieval was produced using the maximum matching segmentation, although, as we saw above, the segmentation accuracy for this algorithm was lower than that of the NMSU segmenter. While this indicates that there does not appear to be a direct correlation between segmentation accuracy and retrieval performance, we should emphasize that our results are very preliminary and incomplete. Nevertheless, this is a result we intend to investigate fully, as described below.

Discussion and Future Work

Our experiments give empirical evidence that accurate segmentation improves retrieval performance, as our preliminary results show that segmenting the text with either NMSU or maximum matching improved the retrieval performance of the character-as-word algorithm. This result is encouraging, and there are many areas remaining to systematically investigate. Our further research in this area will focus on two main areas.

A logical area of investigation is algorithms for scoring word segmentation, with the goal being to develop scoring algorithms which better correlate segmentation accuracy with retrieval performance. A necessary factor in such scoring algorithms will be the types of errors which occur in the segmentation. While our small study involved only three algorithms producing a range of segmentation "scores", the errors these segmentation algorithms produce are very different. For example, since the NMSU segmenter was developed with extensive manual effort, some segmentation errors may be attributed to the complex interaction between its specialized modules. Errors made by the maximum matching algorithm, on the other hand, may be attributed largely to gaps in the word list used. Additionally, the recognition of names and unknown words are traditionally[4] the largest source of segmentation errors, and individually scoring performance on these subsets may provide more information about the interaction between word segmentation and information retrieval performance.

In addition to scoring algorithms, we plan to investigate a more complete range of segmentation algorithms and the resulting retrieval performance, including the following.

NMSU component modules The NMSU segmenter consists of an initial approximation followed by a sequence of iterative refinements. As described above, these refinement steps attempt to recognize idiomatic expressions, derived words, Chinese person names, and foreign proper names. It will be interesting to determine the contribution of each of these steps to the segmentation accuracy as well as the retrieval score.

Maximum matching word list Since the word list for the maximum matching experiment was taken from the NMSU segmenter, we can similarly systematically determine the effect of removing compounds, idioms, and proper names from the word list. We can also experiment with word lists taken from different sources.

Character grouping Since most Chinese words are one or two characters, a segmentation based on character bigrams, in addition to the character-as-word segmentation, may be useful in retrieval. Similarly, it may be helpful to use frequency-based phrase building, that is, segmentation based on character n-gram occurrences in the collection.

Transformation sequences We have developed an algorithm for improving existing segmentation accuracy using a sequence of transformations. With such a sequence, we can systematically experiment with segmentation accuracy and its resulting effect on retrieval. This transformation-based algorithm will assist us in

[4] See, for example, (Wu & Fung 1994) and (Sproat *et al.* 1996).

investigating segmentation scoring algorithms as well as retrieval performance.

References

Broglio, J.; Callan, J.; and Croft, W. B. 1996. Technical issues in building an information retrieval ssytem for chinese. CIIR Technical Report IR-86, University of Massachusetts, Amherst.

Buckley, C.; Singhal, A.; and Mitra, M. 1996. Using query zoning and correlation within SMART: TREC 5. in (Harman 1996a).

Harman, D. K. 1996a. Proceedings of the fifth text retrieval conference (TREC-5). Gaithersburg, MD, November 20-22.

Harman, D. K. 1996b. Proceedings of the fourth text retrieval conference (TREC-4). NIST Special Publication 500-236.

Jin, W. 1994. Chinese segmentation disambiguation. In *Proceedings of the International Computational Linguistics-94 (COLING'94)*.

Rijsbergen, C. J. V. 1979. *Information Retrieval*. London: Butterworths.

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

Wu, D., and Fung, P. 1994. Improving chinese tokenization with linguistic filters on statistical lexical acquisition. In *Proceedings of ANLP94*.