An Integrated Method for Chinese Unknown Word Extraction[1]

LUO Zhiyong
College of Computer Science
Beijing University of Technology
Beijing, PRC 100022
Center for Language Information Processing
Beijing Language and Culture University
Beijing, PRC 100083
luo_zy@blcu.edu.cn

SONG Rou
Center for Language Information Processing
Beijing Language and Culture University
Beijing, PRC 100083
songrou@blcu.edu.cn

Abstract

Unknown word recognition is an important problem in Chinese word segmentation systems. In this paper, we propose an integrated method for Chinese unknown word extraction for off-line corpus processing, in which both context-entropy (on each side) and frequency ratio against background corpus are introduced to evaluate the candidate words. Both of the measures are computed efficiently on Suffix array with much less space overhead. Our method can also be reinforced when combined with a basic Segmentor by boundary-verification and arbitrary n-gram words can be extracted by our method. We test our method on Chinese novel *Xiao Ao Jiang Hu*, and obtain satisfactory achievements compared to traditional criteria such as Likelihood Ratio.

1 Introduction

The unique feature of Chinese writing system is that it is character-based, not word-based. The fact that there are no delimiters between words poses the well-known problem of word segmentation. Any Chinese Information Processing (CIP) systems beyond character level, such as information retrieval, automatic proofreading, text classification, text-to-speech conversion, syntactic parser, information extraction and machine translation, etc. should have a built-in word segmentation block. Currently, dictionary-based method is the basic and efficient one for word segmentation. A fixed Chinese electronic dictionary is required for most CIP systems. Yet there are many unknown words (out of the fixed dictionary) coming into being all the time. The unknown words are diverse, including proper nouns (person names, place names, organization names, etc.), domain-specific terminological nouns and abbreviations, even author-coined terms, etc. and they appear frequently in real text. This may cause ambiguity in Chinese word segmentation and lead to errors in the applications. Presently, many systems (Tan et al, 1999), (Liu, 2000), (Song, 1993), (Luo et al, 2001) focus on online recognition of proper nouns, and have achieved inspiring results in news-corpus but will be deteriorated in special text, such as spoken corpus, novels. As to the rests of unknown words types, it is still the obstacle of application systems, although they are really important for specific collections of texts.

For instance, according to our count on Chinese novel *Xiao Ao Jiang Hu* (《笑傲江湖》) (JIN Yong (金庸), 1967), there are almost 515 unknown word types (out of our 243,539-item general dictionary) of total 39,404 occurrences and total 112,654 characters, and there are 983,134 characters overall in this novel (that is, about 11.46% characters of the whole novel are occupied by unknown words.). And most of them, such as "东方不败"(person name), "辟邪剑谱"(normal noun), "日月神教"(organization name), etc. can't be recognized by most current CIP systems. It is important to note that without efficient unknown word extraction method, most CIP systems can't obtain satisfactory results.

2 Relative research works

Offline unknown word extraction can be treated as a special kind of Automatic Term Extraction (ATE). There are many research works on ATE. And most successful systems are based on statistics. Many statistical metrics have been proposed, including point-wise mutual information (MI) (Church et al, 1990), mean and variance, hypothesis testing (t-test, chi-square test, etc.), log-likelihood ratio (LR) (Dunning, 1993), statistic language model (Tomokiyo, et al, 2003), and so on. Point-wise MI is often used to find

[1] This paper is supported by NSFC (60272055) and 863 Project (2001AA114111)

interesting bigrams (collocations). However, MI is actually better to think of it as a measure of independence than of dependence (Manning et al, 1999). LR is one of the most stable methods for ATE so far, and more appropriate for sparse data than other metrics. However, LR is still biased to two frequent words that are rarely adjacent, such as the pair (the, the) (Pantel et al, 2001). On the other aspect, MI and LR metrics are difficult to extend to extract multi-word terms.

Relative frequency ratio (RFR) of terms between two different corpora can also be used to discover domain-oriented multi-word terms that are characteristic of a corpus when compared with another (Damerau, 1993). In this paper, RFR values between source corpus and background one will be used to rank the final candidate-list.

There are also many hybrid methods combined statistical metrics with linguistic knowledge, such as Part-of-Speech filters (Smadja, 1994). But POS filters are not appropriate for Chinese term extraction.

Since all the terms extraction approaches need to access all the possible patterns and find their frequency of occurrence, a highly efficient data structure based on PAT-tree (Chien, 1997), (Chien, 1998) and (Thian et al, 1999) has been used popularly for this purpose. However, PAT-tree still has much space overhead, and is very expensive for construction. Now, we introduce an alternative data structure as Suffix array, with much less space overhead, to commit this task.

In this paper, we propose a four-phase offline unknown word extraction method: (a) Construct the Suffix arrays of source text and background corpus. In this phase, Suffix arrays, sorted on both left and right sides context for each occurrence of Chinese character, are constructed. We call them Left-index and Right-index respectively; (b) Extract frequent n-gram candidate terms. In this phase, firstly we extract n-grams, appearing more than one time in different contexts according to Left-index and Right-index of source text, into Left-list and Right-list respectively. Then, we combine Left-list with Right-list, and extract n-grams which appear in both of them as candidates (C-list, for short). We also compute frequency, context-entropy and relative frequency ratio against background corpus for each candidate in this phase; (c) Filter candidates in C-list with context-entropy and boundary-verification coupled with General Purpose Word Segmentation System (GPWS) (Lou et al, 2001). In this phase, we segment each sentence, where each candidate appears, in the source text with GPWS and eliminate the candidates cross word boundary; (d) Output the final terms on relative frequency ratios.

The remainder of our paper is organized as follows: Section 2 describes the candidate terms extraction approach on Suffix array. Section 3 describes the candidates' filter approach on context-entropy and boundary-verification coupled with GPWS. Section 4 describes the relative frequency ratios and output of the final list. Section 5 gives our experimental result and Section 6 gives conclusion and future work.

3 Candidates extraction on Suffix array

Suffix array (also known as String PAT-array)(Manber et al, 1993) is a compact data structure to handle arbitrary-length strings and performs much powerful on-line string search operations such as the ones supported by PAT-tree, but has less space overhead.

Definition 1. Let $X = x_0x_1x_2..x_{n-1}x_n$ as a string of length n. For the sake of left and right context sorting, we have extended X by inserting two unique terminators ($, less than all of the characters) as sentinel symbols at both ends of it, i.e. $x_0 = x_n = \$$ in X. Let $LS_i = x_ix_{i-1}..x_0$ ($RS_i = x_ix_{i+1}..x_n$) as the left (right) suffix of X that starts at position i.

The Suffix array Left-index[0..n] (Right-index[0..n]) is an array of indexes of LS_i (RS_i), where $LS_{Left\text{-}index[i]} < LS_{Left\text{-}index[j]}$ ($RS_{Right\text{-}index[i]} < RS_{Right\text{-}index[j]}$), i<j, in lexicological order.

Let LLCP[i] (RLCP[i]), i=0..n-1, as the length of Longest Common Prefix (LCP) between two adjacent suffix strings, $LS_{Left\text{-}index[i]}$ and $LS_{Left\text{-}index[i+1]}$ ($RS_{Right\text{-}index[i]}$ and $RS_{Right\text{-}index[i+1]}$). These arrays on both sides are assistant data structures for speeding string search.

Figure 1 shows a simple Suffix array sorted on left and right context, coupled with the LCP arrays respectively.

We apply the sort-algorithm proposed by (Manber et al, 1993), which takes O(nlogn) in worst cases performance, to construct the Suffix arrays, and sort all the suffix strings in UNICODE order.

Figure 2 shows fragments of Suffix arrays of test corpus *Xiao Ao Jiang Hu* in readable style.

Sorted suffix arrays have clustered all similar n-grams (of arbitrary length) into continuous blocks and the frequent string patterns, as the longest common prefix (LCP) of adjacent strings, can be extracted by scanning through the suffix arrays sorted on left context and right respectively.

String “tobeornottobe”

#	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
String	$	t	o	b	e	o	r	n	o	t	t	o	b	e	$

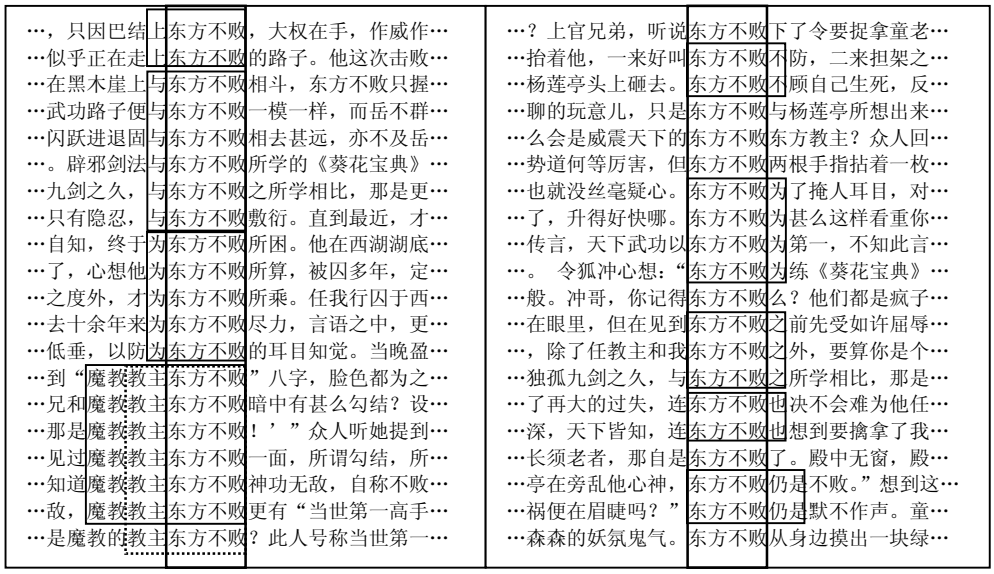
Suffix array

#	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Left-index	0	14	3	12	4	13	7	5	8	2	11	6	1	9	10
Right-index	0	14	12	3	13	4	7	11	2	5	8	6	10	1	9

LCP arrays on both sides

#	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
LLCP	0	0	3	0	4	0	0	1	1	2	0	0	1	1	/
RLCP	0	0	2	0	1	0	0	3	1	1	0	0	4	1	/

Figure 1: Suffix array example



*Left part is fragment of Left Suffix array, starts at position of Chinese character “败”
*Right part is fragment of Right Suffix array, starts at position of Chinese character “东”

Figure 2: Fragments of Suffix array of *Xiao Ao Jiang Hu*

As show in Figure 2, on right sorted part which starts at the position of Chinese Character “东”, we can extract the repeated n-grams, such as “东方不败不”, “东方不败为”, “东方不败之”, “东方不败也”, “东方不败仍是”, “东方不败”, etc., in turn and skip many substrings, such as “东方”, “东方不”, etc., because they are not the LCP of adjacent suffix strings and only appear in the upper string “东方不败” for their all occurrences. We can apply the same skill on left sorted part which start at the position of Chinese character “败”, and extract “上东方不败”, “与东方不败”, “为东方不败”, “魔教教主东方不败”, “教主东方不败”, “东方不败”, etc., as repeated n-grams and skip many substrings, such as “不败”, “方不败”, etc., for the same reasons.

To extract candidate terms, we can scan through both left and right Suffix arrays and select all repeated n-grams into Left-list and Right-list respectively. The terms, which appear in both lists, can be treated as candidates (denoted by C-list). Extraction procedure can be done efficiently by coupled with the arrays of length of LCP on both sides via stack operations. The length and frequency of candidates can also be computed in this procedure.

For example in Figure 2, term “东方不败” should appear in both Left-list and Right-list, and it is a good candidate. Yet n-grams “东方不败也” is not a candidate because even though “东方不败也” does appear in Right-list, it does not exist in our final Left-list (It always appears as a substring of direct upper string “连东方不败也” according to right part of Figure 2).

Term	TC	Left Context-entropy	Right Context-entropy	RFR
令狐冲	5922	6.6804	4.9900	22743.7
了一	1267	4.7974	3.8534	0.9
岳不群	1184	5.9656	4.8688	10104.9
也不	1123	4.8512	4.1473	1.0
盈盈	1053	5.5446	4.7758	89.8
林平之	929	5.7310	4.7623	7928.6
岳灵珊	919	5.5887	4.5220	7843.2
我行	532	0.0930	4.4570	170.2
任我	528	5.5960	0.0412	1013.9
任我行	525	5.5891	4.4294	4480.6
东方不败	320	4.6805	4.8253	2731.0
五岳剑	284	4.0897	0.0585	2423.8
五岳剑派	281	4.0624	3.7344	2398.2
丹青生	176	4.3386	4.0105	1502.0
辟邪剑谱	156	1.7374	2.0613	1331.3
莫大先生	153	4.6941	4.4650	1305.7
蓝凤凰	103	4.3266	3.4258	879.0
计无施	97	4.2815	3.1410	827.8
黑木崖	80	3.0207	2.7821	682.7
不戒和尚	73	3.6620	3.9186	623.0

Table 1: Examples of candidates order by TC

Table 1 lists many examples of candidates extracted from *Xiao Ao Jiang Hu*, order by term count (TC).

4 Filter candidate terms

As what show in Table 1, not all the terms in C-list extracted in Section 3 can be treated as significant terms because of their incomplete lexical boundaries. There two kinds of incomplete-boundary terms: (1) terms as substring of significant terms; (2) terms overlapping the boundaries of adjacent significant terms. In this section, we will take measures, including Context-entropy test and boundary-verification with common Segmentor (GPWS) with general lexicon, to eliminate these invalid candidates respectively.

4.1 Measure on Context-entropy

According to our investigation, significant terms in specific collection of texts can be used frequently and in different contexts. On the other hand substring of significant term almost locates in its corresponding upper string (that is, in fixed context) even through it occur frequently. In this part, we propose a metric Context-entropy as a measure of this feature to filter out substrings of significant terms.

Definition 2. Assume ω as a candidate term which appears n times in corpus X, $\alpha = \{a_1,a_2,\ldots,a_s\}$ ($\beta = \{b_1,b_2,\ldots,b_t\}$) as a set of left (right) side contexts of ω in X.

Left and right Context-entropy of ω in X can be define as:

$$LCE(\omega) = \frac{1}{n}\sum_{a_i \in \alpha} C(a_i,\omega)\log\frac{C(a_i,\omega)}{n}$$

$$RCE(\omega) = \frac{1}{n}\sum_{b_i \in \beta} C(\omega,b_i)\log\frac{C(\omega,b_i)}{n}$$

where $n = \sum_{a_i \in \alpha} C(a_i,\omega) = \sum_{b_i \in \beta} C(\omega,b_i)$,

$C(a_i, \omega)$ ($C(\omega, b_i)$) is count of concurrence of a_i and ω (ω and b_i) in X.

Significant terms, which can be used in different context, will get high values of Context-entropy on both sides. And the substrings, which almost emerge because of their upper strings, will get comparative low values. The 3rd and 4th columns of Table 1 show the values of Context-entropy on both sides of a list of candidate terms. Many candidates, which almost emerge because of their direct upper

strings, such as “我行”(in “任我行”(person name)), “任我”(in “任我行”(person name)) , “五岳剑”(in “五岳剑派”(organization name)), appear in relatively fixed contexts and should get much lower value(s) of one or both sides of Context-entropy.

4.2 Boundary-verification with GPWS

The candidate list of terms includes all of the n-grams, which appear in different context on both sides more than ones. The unique feature of Chinese writing system is that there are no delimiters between words poses a big problem: Many of candidate terms are invalid because of the overlapped factual words’ boundary, i.e. these candidates include several fragments of adjacent words, such as “山派”(overlapping the boundary of common word “华山”(Hua Mountain)), “令狐公”(overlapping the boundary of common word “公子”(Sir)), etc. listed in Table 2. We eliminate these candidates by verifying boundaries of them with a common Segmentor (GPWS (Lou et al, 2001)) and a general lexicon (with 243,539 words).

GPWS was built as shared framework undertaking different CIP applications. It has achieved very good performance and great adaptability across different application domains in disambiguation, identification of proper nouns (including Chinese names, Chinese place names, translated names of foreigners, organization and company names, etc.), identification of high-frequency suffix phrases and numbers. In this part, we ONLY use the utilities of GPWS to perform the Maximum Match (MM) to find the boundaries of words in lexicon, and all of the unknown words (out of our lexicon) will be segmented into pieces. Coupled with GPWS, we propose a voting mechanism for boundary-verification as follows:

```
For each candidate term in C-list as term
Begin
  Declare falseNum as integer for the number of invalid boundary-check of term;
  Declare trueNum as integer for the number of valid boundary-check of term;
  falseNum = 0;
  trueNum = 0;
  For each sentence, in which term appears, in foreground corpus, as sent
    Begin
      Segment sent with GPWS;
      Compare the term's position in sent with the segment result of GPWS;
      If term crosses the adjacent words boundary
        Set falseNum = falseNum+1;
      Else
        Set trueNum = trueNum+1;
    End
  If falseNum > trueNum
    Set boundary-verification flag of term to FALSE;
  Else
    Set boundary-verification flag of term to TRUE;
End
```

Assistant with the segmentor, we eliminate 38,697 items of total 117,807 in C-list in 96.85% of precision. Table 2 shows many examples of candidates eliminated by sides-verification with GPWS.

Candidate term	Segment result of GPWS for one sentence, in which term appears
山派	咱们/华山/派/却/也是/宁死不屈/。
令狐公	恭喜/令狐/公子/，/你/今日/大喜/啊/。
方证大	想起/那日/他/要/修书/荐/自己/去/见/少林寺/方/证/大师/，
那婆	那/婆婆/身子/也是/一/晃/，

Table 2: Examples of candidates eliminated by GPWS

5 Relative frequency ratio against background corpus

Relative frequency ratio (RFR) is a useful method to be used to discover characteristic linguistic phenomena of a corpus when compared with another (Damerau, 1993). RFR of term ω in corpus X compared with another corpus Y, RFR(ω;X,Y), simply compares the frequency of ω in X (denoted as f(ω,X)) to ω in Y (denoted as f(ω,Y)):

$$RFR(\omega;X,Y) = f(\omega,X)/f(\omega,Y)$$

RFR of term is based upon the fact that the significant terms will appear frequently in specific collection of text (treated as foreground corpus) but rarely or even not in other quite different corpus (treated as background corpus). The higher of RFR values of the terms, the more informative of the terms will be in foreground corpus than in background one.

However, selection of background corpus is an important problem. Degree of difference between foreground and background corpus is rather difficult to measure and it will affect the values of RFR of terms. Commonly, large and general corpora will be treated as background corpus for comparison. In this paper, for our foreground corpus (*Xiao Ao Jiang Hu*), we experientially select a group of novels of the same author excluding *Xiao Ao Jiang Hu* as compared background corpus for some reasons as follows:

(a) Same author wrote all of the novels, including foreground and background. The unique n-

grams in writing style of the author will not emerge on RFR values.

(b) All of the novels are in the same category. The specific n-grams for this category will not emerge on RFR values.

So, most of the candidate terms with higher RFR values will be more informative and be more significant for the source novel.

On the final phase, we will sort all of the filtered candidate terms on RFR values in desc-order so that the forepart of the final list will get high precision for extraction.

The last column of Table 1 shows the RFR values of many candidates compared with our background corpus. Many candidates, such as "了一", "也不", which are frequent in both foreground and background corpus, will get much lower RFR values and will be eliminated from our final top list.

6 Experimental result

We use novel *Xiao Ao Jiang Hu* as foreground corpus compared with the rest of novels of Mr. JIN Yong as background corpus. The total characters of foreground and background corpus are 983,134 and 7,551,555 respectively. We read through the novel *Xiao Ao Jiang Hu* and 5 graduates manually selected 515 new terms (out of our lexicon) with exact meaning in the novel as follows for the final test:

(a) Proper nouns, such as person names: "令狐冲", "东方不败", "令狐大哥", place names: "黑木崖", "思过崖", "恒山别院", organization names: "日月神教", "五岳剑派" etc.

(b) Normal nouns, such as "辟邪剑谱", "吸星大法", etc.

(c) Others, such as "剧斗", "惊怖", etc.

By our method, we extract 117,807 candidates in this novel. Table 3 shows the result after filtering with Context-entropy on both sides and boundary-verification on different total extracted numbers; We also compared our integrated method to traditional measure LR. On lower total number levels, LR will overrun our method in unknown-word recall, and in turn overrun by us on higher levels. As to precision, our method always keeps ahead.

We also notice that both of the methods have much low precision in extraction. To retrieve terms with much certain, we rank the entire final list on RFR values in final phase. Most significant terms will comes in the front of ranked list.

Table 3 shows that our method Table 4 shows the top 12 of final list, and Figure 3 shows the performance of our method on different top levels when ranks the final list on RFR values.

7 Conclusion

Unknown word recognition is an important problem in CIP systems. Suffix array based method is an efficient method for exact arbitrary-length frequent terms. And most of substring of significant terms, which almost appear in fixed contexts, can be eliminated by Context-entropy values. Large lexicon can help to verify the unknown word doundaris and filter incomplete-boundary n-grams. Most significant informative candidates list on the top of final list according to RFR values for subsequent manual confirmation, and on the other aspect, RFR also reflects the internal character of the extracted terms.

Total Number Extracted		Word in Dict	Unknown Words	Precision	Unknown-words Recall
534	Our method	306	57	**0.68**	0.11
	LR	222	103	0.61	**0.20**
1325	Our method	668	126	**0.60**	0.24
	LR	421	171	0.49	**0.33**
2996	Our method	1411	225	**0.55**	0.44
	LR	888	287	0.39	**0.56**
6498	Our method	2877	346	**0.50**	0.67
	LR	1608	366	0.30	**0.71**
11684	Our method	4,643	512	**0.44**	**0.99**
	LR	2,428	427	0.24	0.83

Table 3: Result of our method compared to LR

Term	TF	RFR	Left Context-entropy	Right Context-entropy
令狐冲	5922	22743.7	6.6804	4.9900
岳不群	1184	10104.9	5.9656	4.8688
林平之	929	7928.6	5.7310	4.7623
岳灵珊	919	7843.2	5.5887	4.5220
令狐冲道	915	7809.1	5.5789	4.2271
仪琳	729	6221.6	5.5360	4.4128
田伯光	722	6161.9	5.5751	4.7080
恒山派	553	4719.6	4.7371	3.8601
任我行	525	4480.6	5.5891	4.4294
向问天	516	4403.8	5.4427	4.1689
左冷禅	482	4113.6	5.3223	4.7837
方证	414	3533.3	5.2607	2.6043

Table 4: Top 12 terms of final list order by RFR

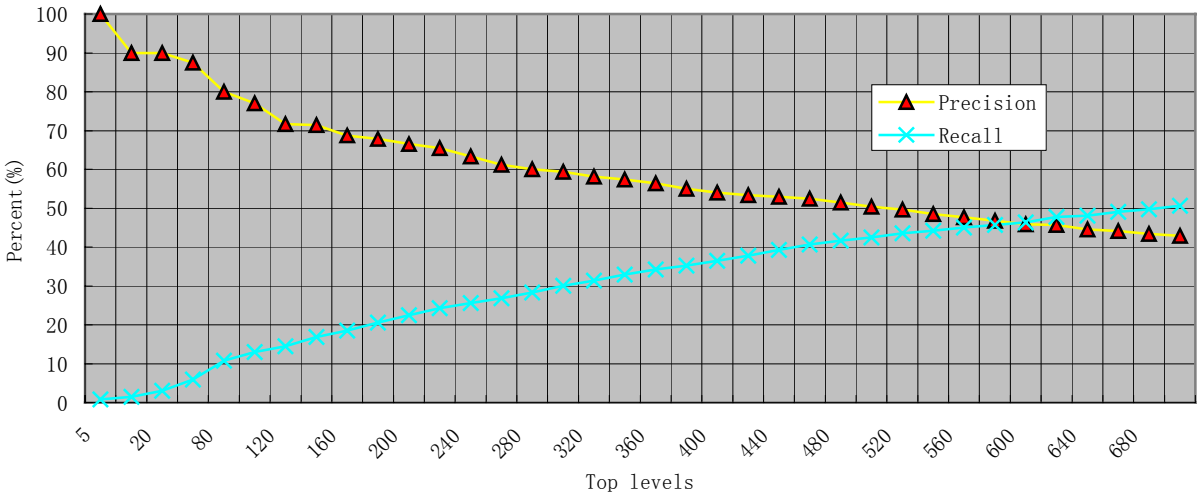


Figure 3: Test result on different top levels

References

Chien, L-F. 1997, *PAT-Tree-Based Keyword Extraction for Chinese Information Retrieval*. Proceedings of the 1997 ACM SIGIR, Philadelphia, PA, USA, pp. 50-58.

Chien, L-F. 1998, *PAT-Tree-Based Adaptive Key phrase Extraction for Intelligent Chinese Information Retrieval*. In special issue on Information Retrieval with Asian Languages, Information Processing and Management, Elsevier Press.

Christopher D. Manning, Hinrich Schutze. 1999. *Foundations of Statistical Natural Language Processing*, MIT Press.

Dekai Wu and Pascale Fung. 1994. *Improving Chinese tokenization with linguistic filters on statistical lexical acquisition*. In Proceedings of the Fourth ACL Conference on Applied Natural Language Processing (ANLP94), Stuttgart, Germany.

Frank Z. Smadja. 1994. *Retrieving collocations from text: Xtract*. Computational Linguistics, 19(1): 143-177.

Fred J. Damerau. 1993. *Generating and evaluating domain-oriented multi-word terms from texts*. Information Processing and Management, 29(4): 433–447.

H.Y. Tan, J.H. Zheng, K.Y. Liu. 1999. *A Study on the Automatic Recognition of Chinese Place Names*, Proceedings of the 5th Joint Conference on Computational Linguistics 99, Tsinghua University Press.

Kenneth W. Church and Patrick Hanks. 1990. *Word association norms, mutual information, and lexicography*. Computational Linguistics, volume 16.

Kunihiko Sadakane. 1998. *A fast algorithm for making suffix arrays and for Burrows-Wheeler transformation*, Proceedings of the ieee Data Compression Conference, pp. 129–138.

K.Y. Liu. 2000. *Automatic Segmentation and Tagging for Chinese Text,* Commercial Press.

Manber, U. and Myers, G. 1993. *Suffix Arrays: A New Method for On-Line String Searches*. SIAM Journal on Computing 22, 935-948.

R. Song. 1993. *Recognition of Personal Names Based on Corpus and Rules*, Journal of Computational Linguistics: Research and Applications, Beijing Language Institute Press.

R. Song. 1998. *The Geometric Structures of Chinese Words and Phrases*, International Conference on Chinese Grammers 98, Beijing.

Patrick Pantel and Dekang Lin. 2001. *A statistical corpus-based term extractor*. In E. Stroulia and S. Matwin, editors, Lecture Notes in Artificial Intelligence, pages 36–46. Springer-Verlag.

Ted E. Dunning. 1993. *Accurate methods for the statistics of surprise and coincidence*. Computational Linguistics, 19 (1): 61–74.

Thian-Huat Ong, Hsinchun Chen. 1999. *Updateable PAT-Tree Approach to Chinese Key Phrase Extraction using Mutual Information: A Linguistic Foundation for Knowledge Management*, Proceedings of the Second Asian Digital Library Conference, November 8-9, pp. 63-84.

T. Lou, R. Song, W.L. Li, and Z.Y. Luo. 2001. *The design and Implementation of a Modern General Purpose Segmentation System*, Journal of Chinese Information Processing, Issue No. 5.

T. Tomokiyo and M. Hurst. 2003. *A Language Model Approach to Keyphrase Extraction*. ACL-2003 Workshop on Multiword Expressions: Analysis, Acquisition and Treatment.

Z.Y. Luo, R. Song. 2001. *Integrated and Fast Recognition of Proper Noun in Modern Chinese Word Segmentation*, ICCC, Singapore.