Automatic Extraction of Chinese Terms

Yijiang CHEN, Changle ZHOU, Xiaodong SHI
Computer Science Department, Xiamen University,
Xiamen University Box 342, Xiamen, 361005
{cyj, dozero, mandel}@xmu.edu.cn

***Abstract*-This paper presents a hybrid method of Chinese Term Extraction, combining statistical information with linguistic knowledge. In this paper, we propose the Local Reoccurrence Measure, which greatly improves the automatic term recognition, especially for new terms and other terms with low frequency. This measure can also be applied to term extraction of foreign language and other applications.**

I. INTRODUCTION

Technical terms are important for information retrieval, machine translation, lexicon construction, and digital libraries. Researches for automatic extraction of English technical terms have been well done, which can be divided into three types: (1) pure rule based method [4]; (2) pure statistical method [5]; (3) hybrid method of syntax structure and statistical rule [2] [6].

Typically, the hybrid method of automatic English term extraction is divided in three steps [2] [6]:

Step 1: Term extraction via morphological analysis, part of speech tagging and shallow parsing;

Grammar filters: ((Adj|Noun)$^+$|((Adj|Noun)*

(NounPrep)(Adj|Noun)*Noun,

Step 2: Term weight with statistical information;

Step 3: Term selection, ranking and truncation of terminological lists by thresholds of weight.

Although approaches to term extraction work quite well for English, it is difficult to apply them directly to Chinese due to three differences between Chinese and English. First, Chinese sentences are written without spaces between words. Second, Chinese words have no explicit part of speech, so that Existing grammar filters has little effect on Chinese term extraction. Third, Chinese terms are often translated from foreign language or are so curt, such as '赫夫曼树'(Huffman tree) or '功放'(power amplifier). It also leads to difficulties in designing and using grammar filters. Therefore, more information suitable for Chinese term extraction should be involved in the improvement of recall rate and precision rate.

The study of automatic Chinese term extraction just began in recent years in China [1]. In [1], authors mainly use statistical machine learning method to learn from a Chinese term bank, and apply learned knowledge to recognize Chinese terms from technical corpora.

The domain-independent method we present in this paper extracts technical terms from Chinese technical corpus combining linguistic and statistical information.

II. CORPUS PRE-PROCESSING

In this section, we present a pre-processing approach to technical corpus to generate term candidates and their statistical information. We don't use grammar filters to generate term candidates, but generate all possible word sequences under some conditions. There are three steps in the pre-processing.

Step 1: Word segmentation using reverse maximum matching method.

Step 2: Replace stop-list words with space characters.

A stop-list is a list of words which are not expected to occur as term words in that domain. It is used to avoid the extraction of word sequences that are unlikely to be terms, improving the precision of the output list. Some example are:', ','。','; ','! ',': ','的','这些','由于','只能','无论','甚至','倘若','为了','所以','只要','很多',etc.

Another stop-list (we call it conditional stop-list) is special, including '个','中', etc. '个'can be replaced by a space character if: the word prior to '个' is '一','二', '三',etc.

More linguistic information used in this step will improve the precision rate of the output list and reduce the requirement of computer memory.

Step 3: Generate all possible word sequences and count the total frequency of occurrence of each word sequence from the technical corpus.

Space characters split the technical corpus into many small

0-7803-9361-9/05/$20.00 ©2005 IEEE

segments. For every segment $w_i w_{i+1} \ldots w_r$, generate all possible word sequences $W=w_j w_{j+1} \ldots w_k$ (i<=j, k<=r) as term candidates.

Moreover, we notice that a term $W=w_j w_{j+1} \ldots w_k$ often repeatedly occurs in some local areas in the technical corpus, but general word sequences are normally distributed equably to the whole corpus. We call occurrences of a word sequence W in a local area "a cluster". In this step, we also need to count the times of occurrences in a local area for a word sequence W. Details will be fully discussed later.

Now, consider the pre-processing of the sentence "下面的二叉树是平衡二叉树。".

After word segmentation,

下面 的 二 叉 树 是 平衡 二 叉 树 。

w_1 w_2 w_3 w_4 w_5 w_6 w_7 w_8 w_9 w_{10} w_{11}

Then, w_2 and w_{11} are replaced with blank characters. If $w_1 \in$ stop-list, w_1 will be also replaced with a blank character. w_6 may appear as a portion of a few terms. If you can tolerate the loss of the recall rate, replacing w_6 with a blank character will improve the precision rate.

Finally, generate all possible word sequences and count the total frequency of occurrence of each word sequence W. For example:

f(二)=2，f(二叉)=2，f(二叉树)=2，

f(二叉树是)=1，……，f(叉)=2，f(叉树)=2，f(叉树是)=1，

f(叉树是平衡)=1，……，……

The advantage of word segmentation is that the generation of non-term word sequences such as "叉树是平" will be reduced. The advantage of stop-list is that the generation of non-term word sequences will be also reduced by blank character's splitting the corpus into more small segments.

If word segmentation and the use of stop-list do not affect the generation of real terms in the technical corpus, the word sequence list will contain all real terms in the corpus, i.e., at this point the recall rate is 100% and the precision rate is very low.

Next, we will evaluate the termhood of each word sequence, sort the word sequence list according to their termhood and provide top M word sequences to the domain expert for manual selection.

III. EVALUATION OF TERMHOOD

A term, must be a phrase, and is domain-related. In the word sequence list, which are terms? We use following parameters to evaluate the termhood of every word sequence $W=w_1 w_2 \ldots w_k$.

A. Parameters for evaluation of termhood

(1) The frequency of occurrence measure

Many terms occur frequently in the technical corpus, so the frequency of occurrence may show the termhood of these terms. However, there are some terms with low frequency. Moreover, there are some word sequences with high frequency though these word sequences are not real terms. So, the frequency of occurrence of a word sequence is not suitable to be an independent parameter.

(2) The glue measure

When k>1, word sequence $W=w_1 w_2 \ldots w_k$ is a multi-word sequence (k-gram) .

A multi-word term at first must be a phrase, and the glue between words of k-gram must be high. Silva [3] summarized and improved six kinds of glue for general multi-word phrase extraction, such as the mutual information glue, the fair Loglike glue, the fair SCP glue, etc.

The fair Loglike glue is as follows:

$$Loglike\_f((w_1 \ldots w_n)) = 2*(\log l(pf1, kf1, nf1) + \log l(pf2, kf2, nf2) - \log l(pf, kf1, nf1) - \log l(pf, kf2, nf2)$$

Where

$$\log l(P, K, M) = K * \ln(P) + (M-K) * \ln(1-P)$$

$$Avx = \frac{1}{n-1} * \sum_{i=1}^{i=n-1} f(w_1 \ldots w_i) \qquad kf1 = f(w_1 \ldots w_n)$$

$$Avy = \frac{1}{n-1} * \sum_{i=2}^{i=n} f(w_i \ldots w_n) \qquad kf2 = Avx - kf1$$

$$nf1 = Avy \qquad nf2 = N - nf1$$

$$pf = \frac{kf1 + kf2}{N} \qquad pf1 = \frac{kf1}{nf1} \qquad pf2 = \frac{kf2}{nf2}$$

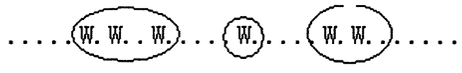
The glue measure can show the degree how close between $w_1, w_2, \ldots, w_k$. But it is only suitable under the condition: f(W)≥ f_threshold and k>1. Multi-word terms with high frequency often have very high glue.

(3) The local reoccurrence measure

In a technical corpus, for discussing an issue, related terms $W=w_1 w_2 \ldots w_k$ will often occur repeatedly in a local area. For

example, in the chapter "树", the term "二叉树" will often appear repeatedly at the same paragraph. Generic phrases will not or just occasionally. Though the frequency of a generic phrase is not low, but it will normally separate equally in the whole corpus. For this phenomenon, we propose the Local Reoccurrence Measure.

We consider the corpus as the time sequence of words.



Graph 3.1

For word sequence W, assumed that its frequency in the technical corpus is F=f(W), and positions in the corpus are $x_1(W)$, $x_2(W)$, ...,$x_F(W)$. If a position sub-sequence $x_i(W)$, $x_{i+1}(W)$, ..., $x_{i+c-1}(W)$ ($1\le i, i+c-1\le F$) satisfies the following conditions:

$\forall k \{i\le k<i+c-1, D(x_k(W),x_{k+1}(W)) \le di\}$

and $D(x_{i-1}(W),x_i(W))>di$

and $D(x_{i+c-1}(W),x_{i+c}(W))>di$

Where, the function D(x,y) is the distance between the position x and the position y, the parameter di is the threshold of distance, we define $x_i(W)$, $x_{i+1}(W)$, ..., $x_{i+c-1}(W)$ as "a cluster". c is the times of the cluster of word sequence W. When c=1, we call the cluster "lone cluster". Assumed that there are m clusters in the corpus, and times of occurrence in a cluster is Ci, $1\le i\le m$, the definition of local reoccurrence measure (LR) of word sequence W is as follows:

$$\delta(x)=\begin{cases}1 & x>1\\ 0 & x=1\end{cases}$$

$$LR(W)=\frac{\sum_{i=1}^{m}\delta(C_i)*C_i}{f(W)}$$

For example, in graph 3.1, there are 3 cluster of word sequence W, so, the local reoccurrence measure of word sequence W is

LR(W) = (3*1+1*0+2*1)/6=5/6

Distance function D(x,y) may have many forms, corresponding to many different local reoccurrence measure LR(W). For example:

① The criterion "x and y is adjacent when x and y are in the same paragraph ", i.e.,

$D_{P0}(x,y)=|Paragraph(x)-Paragraph(y)|$ di=0 Where,

Paragraph(x) is the paragraph number of x in the corpus. And, $LR_{P\ 0}(W)$ is the local reoccurrence measure under this criterion.

② Considering that when the term W occurred in position x, it will often reoccur in the position y after several paragraphs, the criterion "x and y is adjacent when x and y are within the t paragraphs ", i.e.,

$D_{Pt}(x,y)=|Paragraph(x)-Paragraph(y)|$ di=t Where,

Paragraph(x) is the paragraph number of x in the corpus. And, $LR_{P\ t}(W)$ is the local reoccurrence measure under this criterion.

The value of parameter t is related to the corpus, for example, t=5。

③The criterion "x and y is adjacent when x and y are in the same chapter ", i.e.,

$D_C(x,y)=|Chapter(x)-Chapter(y)|$ di=0

Where, Chapter(x) is the chapter number of x in the corpus. And, $LR_C(W)$ is the local reoccurrence measure under this criterion.

④ If the corpus is the set of many documents, the criterion "x and y is adjacent when x and y are in the same document ", i.e.,

$D_D(x,y)=|Document(x)-Document(y)|$ di=0 Where,

Document(x) is the document number of x in the corpus. And, $LR_D(W)$ is the local reoccurrence measure under this criterion.

Propositions of the local reoccurrence LR(W):

$$0\le LR(W)\le 1 \qquad (1)$$

Prove: $\because f(W)=\sum_{i=1}^{m}C_i$

$$\therefore LR(W)=\frac{\sum_{i=1}^{m}\delta(C_i)*C_i}{f(W)}\ge\frac{\sum_{i=1}^{m}0*C_i}{f(W)}=0$$

$$LR(W)=\frac{\sum_{i=1}^{m}\delta(C_i)*C_i}{f(W)}\le\frac{\sum_{i=1}^{m}1*C_i}{\sum_{i=1}^{m}C_i}=1$$

$$LR(W)=1-\frac{m-\sum_{i=1}^{m}\delta(C_i)}{f(W)} \qquad (2)$$

Prove:

$$\because f(W)=\sum_{i=1}^{m}C_i=\sum_{i=1}^{m}(1-\delta(C_i))+\sum_{i=1}^{m}\delta(C_i)*C_i$$

$$\therefore LR(W)=\frac{f(W)-\sum_{i=1}^{m}(1-\delta(C_i))}{f(W)}=1-\frac{m-\sum_{i=1}^{m}\delta(C_i)}{f(W)}$$

The local reoccurrence measure does not depend on the frequency of occurrence of word sequence W, but depend on the degree how repeatedly word sequence W occurs. If all of clusters of word sequence W are lone clusters, in spite of very high frequency of occurrence of word sequence W, the local reoccurrence measure LR(W) = 0. Contrarily, if all of clusters of word sequence W are not lone clusters, in spite of very low frequency of occurrence of word sequence W, the local reoccurrence measure LR(W)=1.

Existing statistical methods of term extraction mainly depend on the high frequency of occurrence in the corpus, since terms tend to appear with high frequencies. And these methods do not process terms each of which occur less 5 times in the corpus due to the consideration that statistical methods will not work under this situation (the glue measure we use faces the same problem). Therefore, new terms and other terms with low frequency will not be extracted. We can use the local reoccurrence measure to partially solve the problem. The local reoccurrence measure can also be used for the term extraction of foreign languages.

B. Termhood of word sequence W

Now, we use parameters above to evaluate the termhood of word sequence W= $w_1w_2...w_k$. Considering that "the glue" only suitable for the extraction of multi-word terms with high frequency of occurrence, we divide all word sequence W into three group: Multi-word sequence with high frequency (f(W)≥ f_threshold and k>1); Multi-word sequence with low frequency (1< f(W) < f_threshold, k>1) ; Uni-word sequence (k = 1)。 We do not consider multi-word sequence W with f(W)=1, since the value of the local reoccurrence measure of them are 0, that is, the local reoccurrence measure is not suitable for multi-word sequences with f(W)=1. Normally, the value of f_threshold is 5, but we think it depends on the size of the corpus.

The definition of termhood of word sequences for each group is as follows:

(1) The termhood of multi-word sequence W= $w_1w_2...w_k$ with high frequency ($f(w_1w_2...w_k)\geq$f_threshold and k>1),

$$\begin{aligned}\text{termhood1}(W)=&\lambda_1*\text{Ordinal}_{g(W)}(W)\\&+\lambda_2*\text{Ordinal}_{LR_{P0}(W)}(W)\\&+\lambda_3*\text{Ordinal}_{LR_{P1}(W)}(W)\\&+\lambda_4*\text{Ordinal}_{LR_{C}(W)}(W)\end{aligned}$$

Where, λ_i is the weight of its correspondent parameter, $\lambda_1+\lambda_2+\lambda_3+\lambda_4=1$.

$Ordinal_{h(W)}(W)$ is the order number of multi-word sequence W with high frequencies when all of these word sequences are sorted by h(W). We let t = 5.

(2) The termhood of multi-word sequence W= $w_1w_2...w_k$ with low frequency ($f(w_1w_2...w_k)$ <f_threshold and k>1),

$$\begin{aligned}\text{termhood2}(W)=&\{\lambda_5*\text{Ordinal}_{LR_{P0}(W)}(W)\\&+\lambda_6*\text{Ordinal}_{LR_{P1}(W)}(W)\\&+\lambda_7*\text{Ordinal}_{LR_{C}(W)}(W)\}*\log f(W)\end{aligned}$$

Where, $\lambda_5+\lambda_6+\lambda_7=1$,

$Ordinal_{h(W)}(W)$ is the order number of multi-word sequence W with low frequencies when all of these word sequences are sorted by h(W). We let t = 30.

(3) termhood of Uni-word sequence W= w_1

Assumed that frequencies of Uni-word sequence W= w_1 in the technical corpus (also called specific corpus) and generic corpus are $f_s(W)$ and $f_g(W)$, if $\frac{f_s(W)}{f_g(W)}$ is high, the probability of W as a term is high.

First, define the function h(W) as follows:

$$h(W)=\begin{cases}\frac{f_s(W)}{f_g(W)}*\log f_s(W) & f_g(W)\geq 1\\ f_s(W)*\log f_s(W) & f_g(W)=0\end{cases}$$

The termhood of Uni-word sequence W= w_1 is as follows:

$$\text{termhood3}(W)=\lambda_8*\text{Ordinal}_{h(W)}(W)+\lambda_9*\text{Ordinal}_{LR_{P0}(W)}(W)+\lambda_{10}*\text{Ordinal}_{LR_{Pt}(W)}(W)+\lambda_{11}*\text{Ordinal}_{LR_C(W)}(W)$$

$$\lambda_8+\lambda_9+\lambda_{10}+\lambda_{11}=1$$

If Termhood(W) is high, the probability of W as a term is high. In addition, Termhood1, Termhood2 and TermHood3 are incommensurable, since the evaluation criterion of each is different.

IV. POST-PROCESSING

The last word of many multi-word sequences W= $w_1w_2...w_k$ is an empty word, such as "二叉树中". Their probability as terms is very low. So, their termhood should be punished after calculating.

When word sequence W= $w_1w_2...w_k$ is generated in the pre-processing, W_1= $w_1w_2...w_{k-1}$ and W_2= $w_2w_3...w_k$ are also generated, Such as W="线索二叉树", W_1="线索二叉" and W_2="二叉树". If $f(W)/f(W_1)>T$, we think W_1 is not a phrase. So the termhood of W_1 should be published. T is a threshold, such as T=0.92. It is the same when $f(W)/f(W_2)>T$.

V. ALGORITHM

The whole algorithm for term extraction is as follows:

(1) Word segmentation of the technical corpus

(2) Use blank character to replace words which appear in stop-list.

(3) Generate all possible word sequences, count the frequency of occurrence of each word sequence, the number of "uni-paragraph lone cluster" lone_m_{P0} (W), the number of "multi-paragraph lone cluster" lone_m_{Pt}, the number of "chapter lone cluster" lone_m_C (W),

(4) Calculate the glue for multi-word sequences with high frequency.

(5) For each word sequence W, calculate Uni-paragraph local reoccurrence measure

$$LR_{P0}(W)=1-\frac{\text{lone\_m}_{P0}(W)}{f(W)}$$

Multi-paragraph local reoccurrence measure

$$LR_{Pt}(W)=1-\frac{\text{lone\_m}_{Pt}(W)}{f(W)}$$

Chapter local reoccurrence measure

$$LR_C(W)=1-\frac{\text{lone\_m}_C(W)}{f(W)};$$

(6) Calculate termhood1 for each multi-word sequence with high frequency. After sorting them by termhood1, get top M_1 word sequences for manual selection;

(7) Calculate termhood2 for each multi-word sequence with low frequency. After sorting them by termhood2, get top M_2 word sequences for manual selection;

(8) Calculate termhood3 for each uni-word sequence. After sorting them by termhood3, get top M_3 word sequences for manual selection;

In fact, step (1) (2) (3) can be fulfilled by one scanning of the technical corpus.

VI. RESULT ANALYSIS

Due to no standard technical corpus commonly used in papers published before for the evaluation of term extraction, we use the well-known text book "Data Structure"(C language version, written by Weimin Yan, 1997, Tsinghua University Press) as technical corpus, which is easy to get. The corpus contains 280K words. There are 417 terms gotten by manual selection. The lexicon contains 47130 words. After word segmentation, 85180 word sequences are generated, and total frequencies of occurrence of them are 234759 times. More details are in Table I. We set f_threshold =5.

For multi-word sequences with high frequency, after calculating the termhood1 and sorting by termhood1, the results are in Table II;

For multi-word sequences with low frequency, after calculating the termhood2 and sorting by termhood2, the results are in Table III;

For Uni-word sequences, after calculating the termhood3 and sorting by termhood3, the results are in Table IV;

TABLE I

DISTRIBUTION OF NUMBERS OF WORD SEQUENCES

	f(W) ≥5,k>1	1<f(W)<5, k>1	F(W)=1, k>1	k=1	All
The Number	3023	10471	68658	3028	85180
Term Number	313	23	3	78	417

TABLE II

CONDITIONS: $f(w_1w_2...w_k) \geq f\_threshold$ AND $k>1$

PARAMETERS: t=5, λ_1 =0.4, λ_2 =0.1, λ_3 =0.35, λ_4 =0.15

M_1	Terms	Recall	Precision
1000	286	0.913	0.286
1381	313	1	0.226

TABLE III

CONDITIONS: $f(w_1w_2...w_k) < f\_threshold$ AND $k>1$

PARAMETERS: t=30, λ_5 =0.25, λ_6 =0.5, λ_7 =0.25

M_2	Terms	Recall	Precision
500	10	0.435	0.02
2000	14	0.565	0.007

TABLE IV

CONDITIONS: k=1

PARAMETERS: t=5, λ_8 =0.45, λ_9 =0.05, λ_{10} =0.4, λ_{11} =0.1

M_3	Terms	Recall	Precision
400	71	0.910	0.178
739	78	1	0.105

When M_1=1000, M_2=500, M_3=400,

$$\text{Total Recall Rate} = \frac{286+10+71}{417} = 0.88$$

$$\text{Total Precision Rate} = \frac{286+10+71}{1000+500+400} = 0.193$$

If with no M_2,

$$\text{Total Recall Rate} = \frac{286+71}{313+78} = 0.913$$

$$\text{Total Precision Rate} = \frac{286+71}{1000+400} = 0.255$$

The recall rate (55.1%) and the precision rate (68.4%) for a Chinese computer engineering corpus are given in [1]. The recall rate of our method is higher than that of [1], however, the precision rate of our method is lower. But we think the recall rate may be more important, because our purpose is to extract more terms. And because the number of term is small (a thick book just has only 417 terms), the manual selection work is easy though our precision rate is low. Moreover, the method presented in [1] needs a term bank for machine learning, but our method has no such requirement.

Though the recall rate and precision rate of multi-word sequence with low frequency is very low, it is better than no consideration done by existing statistical methods.

The local reoccurrence measure is related to the selection of the corpus. We should select a proper corpus that can enhance the local reoccurrence measure of terms.

For a multi-word sequence with low frequency, the value of parameter t of multi-paragraph local reoccurrence measure should be higher (we set t=30) than for a multi-word sequence with high frequency (we set t=5).

Experiments show that the fair Loglike glue works much better than mutual information glue. So results shown above are based on the fair Loglike glue.

VII. CONCLUTION

The hybrid method in this paper works well for Chinese term extraction. The local reoccurrence measure greatly improves the recall rate and precision rate, and it is also a method for extracting low frequency terms. The measure can also be applied to foreign language term extraction and other applications.

REFERENCES

[1] Sui Zhifang, Chen Yirong, Wei Zhouchao Automatic Recognition of Chinese Scientific and technological Terms Using Integrated Linguistic Knowledge, IEEE International Conference on Natural Language Processing and Knowledge Engineering, 2003

[2] K. T. Frantzi and S. Ananiadou. The C-value / NC-value domain independent method for multiword for multiword term extraction. Journal of Natural Language Processing, 1999

[3] Silva, J. F. and Lopes, G. P. A Local Maxima Method and a Fair Dispersion Normalization for Extracting Multiword Units. In Proceedings of the 6th Meeting on the Mathematics of Language, 1999

[4] Didier Bourigault, Surface Grammatical Analysis for the Extraction of Terminological Noun Phrases, the 14th International Conference on Computational Linguistics, 1992

[5] Patrick Pantel,Dekang Lin, A Statistical Corpus-Based Term Extractor, Source Lecture Notes In Computer Science; Vol. 2056, 2001

[6] Anselmo Peñas, Felisa Verdejo, Corpus-based Terminology Extraction Applied to Information access, Proceedings of the Corpus Linguistics 2001 conference

[7] Ted Dunning, Accurate Methods for the Statistics of Surprise and Coincidence, Association for Computational Linguistics, 1993

[8] Li Qinghu,Chen Yujian,Sun Jiaguang, A New Dictionary Mechanism for Chinese Word Segmentation, Journal of Chinese Information Processing, 2002,Vol17,No4