Learning Concepts from Text Based on the Inner-Constructive Model*

Shi Wang[1,2], Yanan Cao[1,2], Xinyu Cao[1,2], and Cungen Cao[2]

[1] Graduate University of Chinese Academy of Sciences, Beijing China, 100049
wangshi_frock@hotmail.com, caoyanan01@163.com, cxy8202@163.com
[2] Key Laboratory of Intelligent Information Processing, Institute of Computing Technology, Chinese Academy of Sciences, Beijing China, 100080
cgcao@ict.ac.cn

Abstract. This paper presents a new model for automatic acquisition of lexical concepts from text, referred to as *Concept Inner-Constructive Model* (CICM). The CICM clarifies the rules when words construct concepts through four aspects including (1) parts of speech, (2) syllable, (3) senses and (4) attributes. Firstly, we extract a large number of candidate concepts using lexico-patterns and confirm a part of them to be concepts if they matched enough patterns for some times. Then we learn CICMs using the confirmed concepts automatically and distinguish more concepts with the model. Essentially, the CICM is an instances learning model but it differs from most existing models in that it takes into account a variety of linguistic features and statistical features of words as well. And for more effective analogy when learning new concepts using CICMs, we cluster similar words based on density. The effectiveness of our method has been evaluated on a 160G raw corpus and 5,344,982 concepts are extracted with a precision of 89.11% and a recall of 84.23%.

Keywords: concepts acquisition, *Concept Inner-Constructive Model*, knowledge discovery, text mining.

1 Introduction

From the cognitive point of view, knowing concepts is a fundamental ability when human being understands the world. Most concepts can be lexicalized via words in a natural language and are called *Lexical Concepts*. Currently, there is much interest in knowledge acquisition from text automatically and in which concept extraction is the crucial part[1]. There are a large range of other applications which can also be benefit from concept acquisition including information retrieval, text classification, and Web searching, etc.[2–4]

Most related efforts are centralized in term recognition. The common used approaches are mainly based on linguistic rules[5], statistics [6, 7] or a combination of both [8, 9]. In our research, we realize that concepts are not just terms. Terms are domain-specific

* This work is supported by the National Natural Science Foundation of China under Grant No.60496326, 60573063, and 60573064; the National 863 Program under Grant No. 2007AA01Z325.

Z. Zhang and J. Siekmann (Eds.): KSEM 2007, LNAI 4798, pp. 255–266, 2007.

while concepts are general-purpose. Furthermore, terms are just restricted to several kinds of concepts such as named entities. So even we can benefit a lot from term recognition we can not use it to learn concepts directly.

Other relevant works are focused on concepts extraction from documents. Gelfand has developed a method based on the Semantic Relation Graph to extract concepts from a whole document[10, 11]. Nakata has described a method to index important concepts described in a collection of documents belonging to a group for sharing them[11]. A major difference between their works and ours is that we want to learn huge amount of concepts from a large-scale raw corpus efficiently rather than from one or several documents. So the analysis of documents will lead to a very higher time complexity and does not work for our purpose.

In this paper, we use both linguistic rules and statistical features to learn lexical concepts from raw texts. Firstly, we extract a mass of concept candidates from text using lexico-patterns, and confirm a part of them to be concepts according to their matched patterns . For the other candidates we induce an *Inner-Constructive Model* (CICM) of words which reveal the rules when several words construct concepts through four aspects: (1) parts of speech, (2) syllables, (3) senses, and (4) attributes.

The structure of the paper is as follows. Section 2 will show how to extract concept candidates from text using the lexico-patterns. The definition of CICM and CICM-based concepts learning algorithm will be discussed in Sect. 3. In Sect. 4, we will show experimental results. Conclusion and future works will be given in Sect. 5.

2 Extracting Concepts from Text Using Lexico-Patterns

In this research, our goal is to extract huge amount of domain-independent concept candidates. A possible solution is to process the text by Chinese NLU systems firstly and then identity some certain components of a sentence to be concepts. But this method is limited dut to the poor performance of the existing Chinese NLU systems, which still against many challenge at present for Chinese[12]. So we choose another solution based on lexico-patterns.

2.1 The Lexico-Patterns of Lexical Concepts

Enlightened Hearst's work[13], we adopt lexico-patterns to learning lexical concepts from texts. But first design a lot of lexico-patterns patterns manually, some of which are shown in Table 1.

Here is an example to show how to extract concepts from text using lexico-patterns:

Example 1. Lexico-Pattern_No_1 {
Pattern: <?C1><是><一|><个|种><?C2>
Restrict Rules:
not_contain(<?C2>,<!标点>) ∧
lengh_greater_than(<?C1>,1) ∧

Table 1. The Lexico-Patterns for Extracting Concepts from Text

(Only 20 are listed and detail restrictions about the patterns are omitted for simplicity. <?C> stands for concepts, and <?X> represents any characters.)

ID	Lexico-Patterns
1	<?C1><是><一|><个|种|><?C2>
2	<?C1><、><?C2><或者|或是|以及|或|等|及|和|与><其他|其它|其余>
3	<?C1><、><?C2><等等|等><?C3>
4	<?C1><如|象|像><?C2><或者|或是|或|及|和|与|、><?C3>
5	<?C1><、><?C2><是|为><?C3>
6	<?C1><、><?C2><各|每|之|这><种|类|些|样|流><?C3>
7	<?C1><或者|或是|或|等|及|和|与><其他|其它|其余><?C2>
8	<?C1><或者|或是|或|及|和|与><?C2><等等|等><?C3>
9	<?C1><中|里|内|><含|含有|包含|包括><?C2>
10	<?C1>由<?C2><组成|构成>
11	<?C1><包括|包含><?C2>在内的<?C3>
12	<?C1><是|作为|成为><?C2>部分<之一|>
13	<?C1>是<用|由><?C2><做|制|作|加工|炼|制造><而成>
14	<?C1><是|以|是以><?C2><为原料>
15	<?C1><作为|是><?C2>的<开始|开端|开头|中间过程|结束|结尾|结局>
16	<?C1><又称|简称|全称|俗称|旧称|今称|人称|史称|中文名称|英文名称><?C2>
17	<?C1><!左括号><?C2><!右括号><?X>
18	<?C1><的><?C2><是><?C3>
19	<?C1><称为><?C2>的<?C3>
20	<?C1><被视为|被称为|被誉为><?C2>

lengh_greater_than(<?C2>,1) ∧
lengh_less_than(<?C1>,80) ∧
lengh_less_than(<?C2>,70) ∧
not_end_with(<?C1>,<这\|那>) ∧
not_end_with(<?C2>,<的\|而已\|例子\|罢了>) ∧
not_begin_with(<?C2>,<这\|的\|它\|他\|我\|那\|你\|但>) ∧
not_contain(<?C2>,<这些\|那些\|他们\|她们\|你们\|我们\|他\|它\|她\|你\|谁>)
}

Sample sentences and the concepts extracted:

(1) 地球是一个行星，地球会爆炸吗？(The earth is a planet, will it blast?) →<?C1>=地球(The earch); <?C2>=行星(a planet)

(2) 很久很久以前地球是一个充满生机的星球.(Long long ago the Earth is a planet full of vitality.) →<?C1>=很久很久以前地球(Long long ago the Earth); <?C2>=充满生机的星球(a planet full of vitaligy)

How to devise good patterns to get as much concepts as possible? We summarized the following criteria through experiments:

(1) **High accuracy criterion.** Concepts distributing in sentences meet linguistics rules, so each pattern should reflect at least one of these rules properly. We believe that we should know linguistics well firstly if we want create to good patterns.

(2) **High coverage criterion.** We want to get as much concepts as possible. Classifying all concepts into three groups by their characteristics, (i.e. concepts which describe physical objects, concepts and the concepts which describe time) is a good methodology for designing good patterns to get more concepts.

2.2 Confirming Concepts Using Lexico-Patterns

Obviously, not all the chunks we got in section 2.1 are concepts, such as *<?C1>=很久很久以前地球(Long long ago the Earth)* in Example 1 above.

In order to identify concepts from the candidates, we introduce a hypothesis, called Hypothesis 1.

Hypothesis 1. *A chunk ck extracted using lexico-patterns in section 2.1 is a concept if (1) ck has been matched by sufficient lexico-patterns, or (2) ck has been matched sufficient times.*

To testify our hypothesis, we randomly draw 10,000 concept candidates from all the chunks and verify them manually. The association between the possibility of a chunk to be a concept and its matched patterns is shown as Fig. 1:

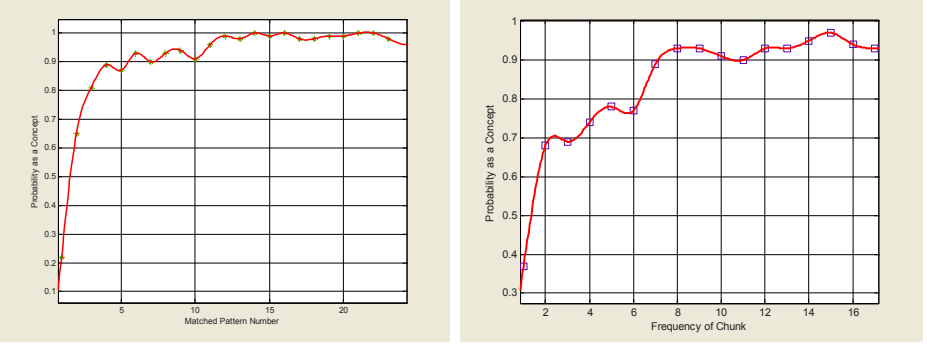

Fig. 1. Association between the lexico-patterns number / the times matched by all the patterns of chunks and their possibility of being concepts

The left chart indicates our hypothesis that the chunks which matched more patterns are more likely to be concepts and the right chart shows that the frequency of the chunk does work well to tell concepts from candidate chunks too. In our experiments, we take the number of patterns matchings to be 5 and threshold of matching frequency as 14, and single out about 1.22% concepts from all the candidate chunks with a precision rate of 98.5%. While we are satisfied with the accuracy, the recall rate is rather low. So in the next step, we develop CICMs to recognize more concepts from chunks.

3 Learning Concepts Using CICM

The CICM is founded on an instinctive hypothesis:

Hypothesis 2. *Most lexical concepts obey certain inner constructive rules.*

That means, when some words form a concept, each word must play a certain role and has certain features. We develop the hypothesis enlightened mainly from the knowledge of linguistics [14] and the cognitive process of human beings creating lexical concepts[15]. Some examples will be given to illuminate the Hypothesis 2 after present the definition of CICM.

3.1 Definition of CICM

According to Hypothesis 2, we can tell whether an unknown chunk is a concept or not by checking whether each word in it whether obeys the CICM. The problems are how to materialize these rules and how to get them. The POS models can reveal these rules using the parts of speech of words but is not precise enough and has many defections[16]. To get better performance we probe into the structure of concepts more deeply and find that besides POS, we must ensure each word's more definite role through at least other three aspects.

Definition 1. *The word model W=<PS, SY, SE, AT> of a word w is a 4-tuple where (1) PS is all the parts of speech of w; (2) SY is the number of w's syllable; (3) SE is the senses of w in HowNet; and (4) AT is the attributes of w.*

The *word models* are integrated information entities to model words. The reason of choosing these four elements listed above will be clarified when we construct CICMs.

Definition 2. *Given a concept cpt* $= w_1 \ldots w_{i-1} w_i w_{i+1}, w_n$ *with n words, the C-Vector of the word* w_i *towards cpt is a n-tuple:*

$$C - Vector(w_i) = <i, W_1, \ldots, W_{i-1}, W_{i+1}, \ldots, W_n> \quad (1)$$

The *C-Vector* of a word stands for one constructive rule when it forms concepts by linking other words and *i* is its position in the concept. A word can have same *C-Vectors* towards many different concepts. The *C-Vector* is the basis of CICM.

Definition 3. *The Concept Inner-Constructive Models (CICMs) of a word w is a bag of C-Vectors, in which each C-Vector is produced by a set of concepts contain w.*

Essentially, CICMs of words represent the constructive rules when they construct concepts. In the four elements of word models, PS and SY embody the syntactical information which have significant roles when conforming concepts in Chinese[14] and are universal for all types of words. SE and AT reveal the semantic information of words and are also indispensably. HowNet is an elaborate semantic lexicon attracted many attentions in many related works[17]. But there are still some words which are missing in it so we need to introduce attributes as a supplement. Attributes can tell the

Table 2. CICM of "生产"

ID	*C-Vectors*	Sample Concepts
1	<1, W(管理) >	生产 管理
2	<1, W(许可证) >	生产 许可证
3	<1,W(实习),W(报告)>	生产 实习 报告
4	<2,W(食品)>	食品 生产
5	<2,W(分布式)>	分布式 生产
6	<2,W(国民),W(总值)>	国民 生产 总值
7	<2,W(新疆),W(建设),W(兵团)>	新疆 生产 建设 兵团
8	<3,W(广东省),W(春耕)>	广东省 春耕 生产
9	<3,W(国家),W(安全),W(监督),W(管理局)>	国家 安全 生产 监督 管理局
…	…	…

semantic differences at the quantative level or qualitative level between concepts. Tian has developed a practicable approach to acquire attributes from large-scale corpora[18].

Table 2 displays the CICMs for the word *"生产(produce, production)"*:

Note that we omit the details of each word vector for simplicity.Taking "国民 生产 总值" for example, the full *C-Vector* is:

< 2,
<{n},2,{属性值,归属,国,人,国家},{有组成,有数量}>,
<{n},2,{数量,多少,实体},{有值域,是抽象概念}> >

3.2 Learning CICMs

Using CICMs as the inner constructive rules of concepts, our next problem is how to get these models. We use the confirmed concepts obtained in section 2.2 as a training set and learn CICMs hidden in them automatically. It is an instance learning process and the following procedure is implemented for this task:

Algorithm 1. *CICMs Instance Learning Algorithm:*

(1) Initializing the resources including (1.1) A words dictionary in which each one has fully parts of speech; (1.2) The HowNet dictionary; and (1.3) An attributes base of words[18].

(2) Constructing a model set MSet to accommodate all the words' models which is empty initially.

(3) For each concept cpt in the training set, segment it and create each word's C-Vector(w_i). Subsequently, if C-Vector(w_i) ∈ MSet(w_i), then just accumulate the frequency; otherwise add C-Vector(w_i) to MSet(w_i).

(4) Removing the C-Vectors which have low frequency for each word's MSet.

Based on experiments, we choose 10% as the threshold of the number of the concepts containing the word in the training set. We exclude the vectors which have low frequency, that is, if a $C-Vector$ for a word is supported by just a few concepts, we look at it as an exception.

4 Clustering Words for More Efficient Analogy

Essentially, CICMs are models of instance analogy. We want to learn new concepts by "recalling" the old ones just as human beings. For example, we can build CICMs for the word *"生产(produce, production)"* like Table 2 and then identify that *"药品 生产(pharmaceutical production)"* is also a concept, because the latter has the same constructive rule as *"食品 生产(food production)"*.

But unluckily, even we know *"药品 生产"* is a concept, our system still can not tell whether *"药品 制造(pharmaceutical manufacture)"* is also a concept for there are no CICMs for the word *"制造"*. The reason for this is that the system still can not make use of word similarity. Therefore, we need to cluster words based on the similarity of CICMs and then learn more new concepts.

4.1 Similarity Measurement of Words

The similarity measurement of CICMs is the basis of clustering words in our task. Our measurement is founded on the intuitive distribute hypothesis that:

Hypothesis 3. *In concepts, similar words have similar CICMs.*

According to Hypothesis 3, the similarity of two words w_1, w_2 is defined as:

$$sim(w_1, w_2) = sim(CICM(w_1), CICM(w_2)) \tag{2}$$

The commonly used similarity measure for two sets includes *minimum distance*, *maximum distance*, and *average distance*. Considering that there are still some noises in our training set which would result in some wrong $C-Vectors$ in CICMs, we choose the average distance for it is more stable for noisy data, that is:

$$\begin{aligned} sim(w_1, w_2) &= \frac{1}{|CICM(w_1|} \sum_{vec_i \in CICM(w_1)} sim(vec_i, CICM(w_2) \\ &= \frac{1}{|CICM(w_1||CICM(w_2|} \sum_{vec_i \in CICM(w_1)} \sum_{vec_j \in CICM(w_2)} sim(vec_i, vec_j) \end{aligned} \tag{3}$$

Now the problem is how to calculate the similarity of two $C-Vectors$ of two words now. For two *C-Vectors*:

$$C-Vector_i =< i, W_1, \ldots, W_n >, C-Vector_j =< j, W_1, \ldots, W_m > \tag{4}$$

We standardize them to an *N-Vector* that is:

$$C-Vector_i =< W^i_{-N}, \ldots, W^i_N >, C-Vector_j =< W^j_{-N}, \ldots, W^j_N > \tag{5}$$

and $W_k = \emptyset$ if there is no word model in position k for both of them We adopt the cosine similarity when compare two vectors, that is:

$$sim(vec_i, vec_j) = \cos(\overrightarrow{vec_i}, \overrightarrow{vec_j}) = \frac{\overrightarrow{vec_i} \cdot \overrightarrow{vec_j}}{|\overrightarrow{vec_i}| \times |\overrightarrow{vec_j}|} \tag{6}$$

4.2 Clustering Words Based on the Density

Among all the clustering methods using density functions has prominent advantages–anti-noisiness and the capability of finding groups with different Inspired by DENCLUE[19], we define a influence function of a word w_0 over another word w:

$$f_B^w = f_B(w_0, w) \tag{7}$$

which is a function proportionately to the similarity of w_0 and w, and reveals the influence degree w_0 over w. Commonly used influence functions include *Square Wave Function* and *Gauss Function*. The former is suitable for the data which dissimilar distinctly while the later is more suitable for reflect the smooth influence of w_0. Because a word is related with many other words in different degrees but no simply 1 or 0 in corpus, it is more reasonable to choose *Gauss Influence Function*:

$$f_{Gauss}^w(w_0) = e^{\frac{-(1-sim(w_0,w_1))^2}{2\sigma^2}} \tag{8}$$

We call Equation (8) the *Gauss Mutual Influence* of w, w_0 for $f_{Gauss}^w(w_0) = f_{Gauss}^{w_0}(w)$. It makes each word linked with many other words to some extent. According to it, we can cluster words into groups. Before giving the definition of a word group, we develop some definitions first for further discussing:

Definition 4. *Given a parameter ξ, $\xi\_region(w_0) = \{w|fGuess(w, w_0) > \xi\}$ is called ξ_region of w_0. Given a parameter $MinPts$, w_0 is called a $CoreWord$ if $|\xi\_region(w_0)| > MinPts$. The minimal ξ which makes w_0 to be a $CoreWord$ is called the $CoreDistance$ of w_0 and be marked as ξ^*.*

Definition 5. *We call w_0 is direct reachable to w' if w_0 is a $CoreDistance$ and $w' \in \xi\_region(w_0)$ and marked as $d_reachable(w_0, w')$. For a set of words $w_0, w_1, \ldots, w_n = w'$, if $d\_reachable(w_i, w_{i+1})$ for all $w_i, 1 \leq i < n$, then w_0 is reachable to w', that is, $reachable(w_0, w')$.*

Based on the definitions above, a word group can be seen as the maximal words set based on the reachable property. The corresponding clustering algorithm is given below:

(1)Taking ξ=ξ^* and for all the words w perform the following operation:

$\xi_{cur} = \xi^*; cw\_cur = \{w\}$
  $while(\xi_{cur} < 1)\{$
    $cw\_pre = cw\_cur;$
    $if(|\xi_{cur}\_region(w_0)| > MinPts)\{$

Build a word group cv_{cur} *which contains all the words in* $\xi_{cur}\_region(w)$ *and takes w as the CoreWord of it.*

$if(\frac{|cv_{cur} - cv_{pre}|}{|cv_{cur}|} < \alpha)\{break;\}$

$\}//if$

$else\{break;\}$

$\xi_{pre} = \xi_{cur}; \xi_{cur} += \Delta;$

$\}//while$

$\}cw = cw_{pre};$

(2)For each pair of $CoreWords\ w_i, w_j$

$if(d\_reachable(w_i, w_j)$

$Merge\ cw_i, cw_j$ *into a new group* cw_{i+j} *which has two CoreWords* cw_i *and* cw_j

(3)Repeate (2) until no new groups are generated.

Many groups with different density will be generated in (2) for we set value for ξ not a single number but a large range of field. The groups with high density will be created firstly and be covered by the dilute groups. We escape choosing the parameter of ξ by doing this.

4.3 Identifying Concepts Using CICMs

Having the learned CICMs and word cluster, identifying method of new concepts is straightforward. Given a chunk, we just create its local $L-Vector$ and judge whether it satisfies one of its or its similar words' $C-Vector$ we have learned.

Definition 6. *For a chunk* $c_k = w_0 \ldots w_n$*, the Local C-Vector for a word* w_i *in it :* $L\_Vector(w_i, c_k) =< i, W_0, \ldots, W_{i-1}, W_{i+1}, \ldots, W_n, >$.

Theorem 1. *For a chunk* $c_k = w_0 \ldots w_n$*, for each word* w_i *in it, there is* $L\_Vector(w_i, c_k) \in CICM(gw_i)$*, then* c_k *is a concept, where* gw_i *is the similar word group of* w_i.

5 Experimental Result and Discussion

5.1 Measurement and Result

Our system is called *Concept Extractor* (CptEx) and use the following formulae to evaluate its performance:

$$p = \frac{||m_a \bigcap m_m||}{||m_a||}, r = \frac{||m_a \bigcap m_m||}{||m_m||}, F - Measure = \frac{2 \times p \times r}{p + r} \tag{9}$$

where m_a are the concepts CptEx extracts and m_m are the ones built manually. To calculate the performance, we selected 1000 chunks from the raw corpus and label the concepts in them manually. We compare the results based on CICMs with those based the Syntax Models and the POS Models as shown in Table 3:

Table 3. Performance of CptEx

Measurement	Syntax Models	POS Models	CICM
p	98.5%	86.1%	89.1%
r	1.2%	87.8%	84.2%
F-measure	2.3%	86.9%	86.6%

Having adopted CICMs to distinguish concepts from the chunks extracted by lexico-patterns, the precision rate drops down to 89.1% while the recall rate flies to 84.2%. The precision rate reduces because there are still some improper CICMs which will confirm fake concepts. The samples below in Table 4 will show this.

Compared with POS Models, CICMs has a higher accuracy rate because we consider more factors to clarify the inner constructive rules rather than using part of speech only. On the other hand, our stricter models result in a lower recall rate.

5.2 Limitations and Analysis

Table 4 shows some chunks and their output result after introducing CICMs in CptEx.

Table 4. sample concepts extracted by CptEx

ID	Chunks	m_m	m_a
1	照相/ 器材/	照相器材	照相器材
2	打印/过程/	打印过程	打印过程
3	学生/ 思想/ 政治/ 工作/	学生思想政治工作	学生思想政治工作
4	国际/ 特/ 奥/ 会/ 董事会/	国际特奥会董事会	国际
			董事会
5	网上/ 交易/ 可以/ 降低/ 经营/ 成本	网上交易	网上交易
		经营成本	经营成本
6	在/ 整个/ 数码/ 处理/ 过程/	整个数码处理过程	整个数码处理过程
7	加盟/ 汉堡/	加盟汉堡	汉堡

We look into the chunk $ck1$ =“照相/ 器材/”. The $L-Vectors$ of words “照相” and “器材” are as below:

+ L_Vector(“照相”, ck1) = <1, <{n},2,{器具},{有形状,有功能,是物质}>>
+ L_Vector(“器材”, ck1) = <2, <{v},2,{拍摄},{是行为}>>

and the CICMs of the two words are as below:

+ CICM(“照相”) ={
  - <2, <{b, n},2,{器具},{有形状,有功能}>, <{n},2,{器具},{有形状,有功能}>>
  - <1, <{n},2,{器具},{有形状,有功能,是物质}>>
  - <1, <{n},1,{场所，商，设施，机构},{有形状}>>
  - ...

  }
+ CICM (“器材”) ={
  - <3, <{n},2,{地点},{有功能,是物质}><{k, n},2,{事务,传播,信息},{有功能,是抽象}>, <{n},1,{场所,制造,厂房},{有形状,有功能}>>
  - <2, <{k, n},2,{事务,传播,信息},{有功能,是抽象}>, <{n},1,{场所,制造,厂房},{有形状,有功能}>>
  - <2, <{v},2,{保护},{是行为}>>
  - <2, <{v},2,{拍摄},{是行为}>>
  - <1, <{n},1,{树,木},{有形状,有高度,是植物}>>
  - ...

  }

Then we can see that (1) L_Vector(“照相”, ck1) ∈ CICM (“照相”) ,and (2) L_Vector (“器材”, ck1) ∈ CICM(“器材”). and can confirm “照相器材” is a concept. Chunks 2/3/5/7 are the same.

On the other side, for the chunk *ck*4= “国际特奥会董事会”, CptEx doesn’t work at present. The reason for this is that “特奥会” is a abbreviation but be wrongly parsed. CICMs of these three separate characters don’t be helpful for there are not much similar training concepts.

The chunk “加盟汉堡队” has been wrongly extracted for the CICMs of “加盟”, “汉堡队”. And for Chunk *ck*8=“加盟/v 汉堡/n”, the *L_Vectors* are all in the CICMs and then produce errors. We are going to cope with this problem through setting confidence of each $C - Vector$ of CICMs according to the frequency of training samples which support it. And the confidence of the concept confirmed by CICMs with the same confidence of the $C - Vector$ in *L_Vector* of it. The concept with a low confidence will be discarded or validated again in open corpus.

6 Conclusion and Future Works

We have described a new approach for automatic acquisition of concepts from text based on Syntax Models and CICMs of concepts. This method extracted a large number of candidate concepts using lexico-patterns firstly, and then learned CICMs to identify more concepts accordingly. Experiments have shown that our approach is efficient and effective. We test the method in a 160G free text corpus, and the outcome indicates the utility of our method.

There are still some more works be done to get better performance for there are some improper CICMs. We plan to validate concepts in an open corpus such as in the World Wide Web in the future.

References

1. Cao, C., et al.: Progress in the Development of National Knowledge Infrastructure. Journal of Computer Science & Technology 17(5, 1), C16 (2002)
2. Ramirez, P.M., Mattmann, C.A.: ACE: improving search engines via Automatic Concept Extraction. In: Proceedings of the 2004 IEEE International Conference, pp. 229–234. IEEE Computer Society Press, Los Alamitos (2004)
3. Zhang, Y.-T., Gong, L., Wang, Y.-C., Yin, Z.-H.: An Effective Concept Extraction Method for Improving Text Classification Performance. Geo-Spatial Information Science 6(4) (2003)
4. Acquemin, C., Bourigault, D.: Term Extraction and Automatic Indexing. Oxford University Press, Oxford (2000)
5. Chen, W.L., Zhu, J.B., Yao, T.: Automatic learning field words by bootstrapping. In: Proc. of the JSCL. Beijing: Tsinghua University Press, pp. 67–72 (2003)
6. Zheng, J.H., Lu, J.L.: Study of an improved keywords distillation method. Computer Engineering 31(194), C196 (2005)
7. Agirre, E., Ansa, O., Hovy, E., Martinez, D.: Enriching very large ontologies using the WWW. In: Proc. of the ECAI 2004 Workshop on Ontology Learning (2004)
8. Du, B., Tian, H.F., Wang, L., Lu, R.Z.: Design of domain-specific term extractor based on multi-strategy. Computer Engineering 31(14), 159–C160 (2005)
9. Velardi, P., Fabriani, P., Missikoff, M.: Using text processing techniques to automatically enrich a domain ontology. In: Proc. of the FOIS, pp. 270–284. ACM Press, New York (2001)
10. Gelfand, B., Wulfekuler, M., Punch, W.F.: Automated concept extraction from plain text. In: AAAI 1998 Workshop on Text Categorization, Madison, WI, pp. 13–17 (1998)
11. Nakata, K., Voss, A., Juhnke, M., Kreifelts, T.: Collaborative Concept Extraction from Documents. In: Reimer, U. (ed.) PAKM 1998. Proc. Second International Conference on Practical Aspects of Knowledge Management, Basel (1998)
12. Zhang, C., Hao, T.: The State of the Art and Difficulties in Automatic Chinese Word Segmentation. Journal of Chinese System Simulation 17(1), 138–C147 (2005)
13. Hearst, M.A.: Automatic acquisition of hyponyms from large text corpora. In: COLING 1992. Proceedings of the 14th International Conference on Computational Linguistics, pp. 539–545 (1992)
14. Lu, C., Liang, Z., Guo, A.: The semantic networks: a knowledge representation of Chinese information process. In: ICCIP 1992, pp. 50–57 (1992)
15. Laurence, S., Margolis, E.: Concepts: Core Readings. MIT Press, Cambridge, Mass (1999)
16. Yu, L.: A Research on Acquisition and Verification of Concepts from Large-Scale Chinese Corpora. A dissertation Submitted to Graduate School of the Chinese academy of Sciences for the degree of master. Beijing China (May 2006)
17. Dong, Z., Dong, Q.: HowNet and the computation of meaning. World Scientific Publishing Co., Inc., Singapore (2006)
18. Tian, G.: Research os Self-Supervised Knowledge Acquisition from Text based on Constrained Chinese Corpora. A dissertation submitted to Graduate University of the Chinese Academy of Sciences for the degree of Doctor of Philosophy. Beijing China (May 2007)
19. Hinneburg, A., Keim, D.: An efficient approach to clustering in large multimedia databases with noise. In: Proceedings of the 4th International Conference on Knowledge Discovery and Data Mining (1998)