An Information Retrieval Model Based On Word Concept

Chen Wu[1, 2], Quan Zhang[2], Xiangfeng Wei[2]
[1] Graduate School of Chinese Academy of Sciences, 100039, Beijing, China
[2] Institute of Acoustics, Chinese Academy of Sciences, 100080, Beijing, China
wuchen@mail.ioa.ac.cn

Abstract. Traditional approaches for information retrieval from texts depend on the term frequency. A shortcoming of these schemes, which consider only occurrences of the terms in a document, is that they have some limitations on extracting semantically exact indexes that represent the semantic content of a document. However, one word can always represent more than one meaning. The word sense ambiguities will also affect the system behavior. To address this issue, we proposed a brand new strategy – a concept extracting strategy to extract the concept of the word and to determine the semantic importance of the concepts in the sentences via analyzing the conceptual structures of the sentences. In this approach, a conceptual vector space model using auto-threshold detection is proposed to process the concepts, and a cluster searching model is also designed. This auto-threshold detection method can help the model to obtain the optimal settings of retrieval parameters automatically. An experiment on the TREC6 collection shows that the proposed method outperforms the other two information retrieval (IR) methods based on term frequency (TF), especially for the lower-ranked documents

Keywords: Information Retrieval, Word Concept, Sentence Category, Auto-threshold detection clustering method

1 Introduction

Searching in archives of documents becomes increasingly frequent for most of people. So, how to provide useful and efficient IR (Information Retrieval) tools becomes more and more important. Since 1968, when the first formal model for IR[1] comes into being, a number of IR models have been developed, such as vector space, probabilistic, fuzzy, logical, inference, language and so on[2-10].While considering all the former methods, we can find most of the methods are based on occurrences of terms in a document (or TF: Term Frequency) and seldom on the content of the document. These algorithms analyze only term occurrences and do not attempt to resolve the meaning of the terms. As we all know, the meaning of words is very helpful to IR. If we disregard the context and dispose the words separately, the performance of IR system will be lowered for the ubiquitous existence of word sense ambiguity. A word may have a lot of meanings. A particular meaning can also be represented by more than one word. Therefore whether we can determine the meaning of the word in a document will affect the accuracy of an IR system. Meanwhile, many studies[11-15] have shown that people understand things by comprehending the concepts represented by the things. The language works in the same way[16]. Consequently, some research teams began to investigate the language conceptual space and the expression of the space using symbolic system. The main part of the IR method we present is to discuss how to draw the conceptual expressions of a word and a sentence based on a symbolic system[16-19]. We consider the advantages of case grammar[20], transformational generative grammar[21-22] and Wordnet[23] fully, and try to form a strategy to extract the conceptual expressions, and then, process the information in a semantic way based on the conceptual expression.

The key rudder of Section 2 is to discuss how to use the concept symbols and the sentence category expressions to represent the words and the sentences via some analyzing strategies and knowledge bases.

Owing to the fact that a simple search usually retrieves a large collection of documents, clustering becomes an efficient tool. In fact, clustering methods[24][25] have been intensively studied in information retrieval for textual documents since 1990s. All these clustering methods need users to evaluate, more or less, the thresholds (e.g.: the value k in k-means). The values of thresholds will affect the results of clustering to some extend. Furthermore, the document vectors may always be of large dimension and sparse. So a fixed-threshold is certain to be inappropriate in some situations. Consequently, an auto-threshold detection clustering method is proposed in the paper. This clustering method uses the simulated curve of the document distances to find the thresholds. The processing object of this clustering method is the concept instead of the word.

In order to do cluster searching, a searching method is proposed. The method here uses the dispersion among individual maximum likelihood, partial maximum likelihood and global maximum likelihood to measure the appearance probability of queries, then to fulfill the task. This method will be more accurate.

Section 3 will discuss the clustering method which is used to generate clustering of the documents, and the searching method which is used to measure the document query similarity.

Section 4 compares the experiments using the proposed methodology and the traditional indexing schemes. The conclusions are given in Section 5.

2 WORD CONCEPT EXTRACTING STRATEGY

In this phase, we transfer the words and sentences to their conceptual forms, through three knowledge bases which we have constructed. We also proposed a good processing strategy which extracts the word concept via the knowledge bases.

2.1 Basic Theoretical Foundations

In the studies of using the conceptual expressions to express the words and the sentences, a conceptual language model has been introduced, which involves the new concepts presented below.

One important concept is **semantic chunk**[2] which is a semantic unit between word and sentence. But it is different from phrase or other traditional chunks. It is a semantic unit expressing a comprehensive concept. We classify the semantic chuck into two types: the **main chunk** and the **supplementary chunk**. The main chunk contains the necessary parts of a sentence from the view of conceptual association. It always describes the object and its actions. The supplementary chunk provides the background knowledge of a sentence, such as time, place etc.

Semantic chunk will be meaningful when it interacts with the **sentence category**[16]. Sentence category is another concept. It is composed of a set of symbolic expressions which stand for the categories of the sentences from the view of the semantics. The expression is named **sentence category expression**. Each expression contains the expressions of chunks and some conjunctive symbols which are used to describe the relationship among chunks. These expressions are designed in advance and able to describe not only the meaning but also the structure of the sentence. Huang concluded 57 types of primitive sentence category expressions and 57*56 compound ones[16][18]. As a result, not only Chinese sentences but also English sentences can be expressed by one of the 57 primitive forms or their compounds.

2.2 Knowledge Bases

The processing involves three knowledge bases: One is called WCK (Word Concept Knowledge base), the second is called **SCK** (Sentence Category Knowledge base), and the last is called SRK (Scheduler Rule Knowledge base). WCK stores the relationship between the word and the concept as well as the important features of the sentence if the word in the sentence can be interpreted by the concept it records. The main structure of WCK is shown in Table 1.

Table 1. The Main Structure Of WCK

column	comment
Word	Record the word itself, such as “prosecute”, “her”, “drink” etc.

Relative Concept Number (RCN)	The number of concepts which the word can be interpreted, such as 1/2, 2/2…
Concept Expression (CE)	One concept expression of the word, such as “va5a” (corresponding to “prosecute”), “p4003/jx621” (“her”), “v62221a” (“drink”).
Sentence Category Expression (SCE)	Sentence category expression which is determined by the content of CE, such as “T3R011*322”, one of the 57*56 types of forms in the SCK.
Characters of Chunks (CC)	Describe the characters of chunks decided by the SCE value, such as ”@S: TB: pea56” which interprets the characters of the main chunk.

WCK is the key knowledgebase. It provides the information about which sentence category expressions should be hypothesized and which characters can be used to validate the hypothesis according to the words the computer reads.

A set of rules have been carried out to tell the researcher how to construct the WCK using concept symbol system. A Chinese WCK containing 50,000 vocabularies have been made by us since 2000.

SCK is a standard sentence category database. It contains 57 sentence category expressions which we have introduced before. Each sentence in the nature language can find its counterpart in the database.

SRK tells the computer which sentence category expression should be hypothesized first, which one second, and so on, according to the words in the sentence.

2.3 Processing Strategy

The processing strategy is based on a conceptual language model, which can be concluded as the following expressions.

$$SCE = f(SCh_1, SCh_2, ..., SCh_n)$$

$$SCh = g(WC_1, WC_2, ..., WC_n)$$

where, SCE is the sentence category and SCh is the semantic chunk. WC refers to the word concept. **Word concept** is the conceptual expression of a word sense. It is expressed by a meaningful character string.

$f()$ and $g()$ are two functions which formalize the semantic relationship between the variables. This model considers the conceptual consistency of the sentence category with the semantic chunks and the word concepts in the sentence. The sentence category is determined by the semantic chunks in the sentence. The semantic chunk is determined by the word concepts in and around the semantic chunk. Meanwhile, the sentence category restricts the word concepts in the sentence in return.

Processing strategy focuses mainly on the processing logic which tells computer how to get the sentence category expression and word concepts through some specific procedures according to the conceptual language model. The diagram of processing strategy is shown as Fig.1.

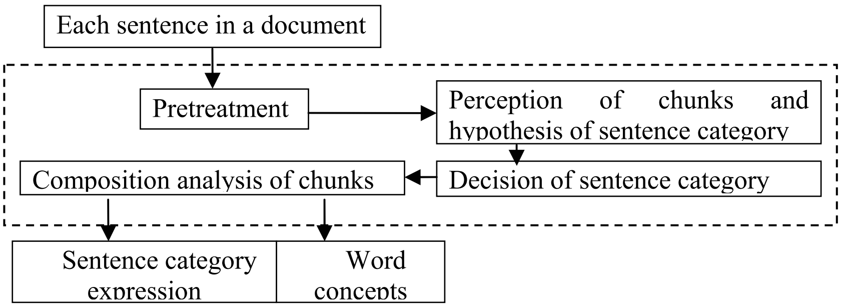
Each sentence in a document
Pretreatment
Perception of chunks and hypothesis of sentence category
Composition analysis of chunks
Decision of sentence category
Sentence category expression
Word concepts

Fig.1 The diagram of processing strategy

The processing strategy can be divided into four phases. The first phase is called pretreatment. The actual assignments in this phase depend on the language it processes. If English is processed, we will do stemming, phrase extracting etc. If Chinese is processed, word segmentation will be performed. In contrast to the traditional word segmentation, every possible segmentation instance will be considered in our processing strategy, which selects the correct segmentation using hypothesis and decision of sentence category.

Perception of chunks and hypothesis of sentence category expression is the second phase. In this phase, the chunks in the sentence and their corresponding possible sentence category expression will be hypothesized according to the words in the sentence and their concepts recorded in the "CE" of the WCK. The possible sentence category expression is stored in "SEC" of the WCK.

Decision of sentence category is the third phase. In this phase, the hypothesis made in the second phase will be checked according to the "CC" in the WCK. The right sentence category expression will be proved and obtained.

Composition analysis of chunks is the last phase. Through this phase, we can get all components of the chunks. Components are expressed by the exact word concepts.

After processing, the sentence category expression and the concept expression of each word in the sentence will be obtained. We use the sentence category expressions to analyze which concept is of high semantic importance and should server the clustering process and which one should be ignored. We take advantage of the concepts only in the clustering process and will disregard the sentence category expressions in the process.

The following example explains the strategy and produces the results directly.

Example: For six months~|| I || lived ||~without a job ~|| in New York.
The sentence category expression of this sentence is: Cn1Cn2S03J, S03 means this sentence is a transposition state sentence due to "live" is some kind of state. The main trucks of this sentence are "I", "lived in" and "New York". "lived in" is the kernel of the main chunk, "I" and "New York" are two general main chunks. The supplementary chunks are "For six months" and "without a job", which act as conditions (CN). Hence we add signs "CN1" and "CN2" in front of the sentence category expression. The concept of "I" in this sentence is "p4001" which means a special personal pronoun. The concept of "live" is "v65500214" which means to reside in a place, while "New York" is "fpj2*304/fpwj2*m1" which means a city in US, etc.

3 HNCIR SYS. BASED ON WORD CONCEPT

HNCIR uses clustering methods to complete the subsequent process, with auto-threshold detecting. Apparently, HNCIR takes the word concept as the processing object instead of the words themselves, which differs from the traditional clustering-based IR System.

3.1 Cluster Generation Methods

In cluster generation, determination of the number of clusters is baffling, especially using concept. There are no experiential parameters available in the literature. Consequently, how to generate the clusters symmetrically becomes most important. In this paper, we propose a new approach to detect the clusters.

HNCIR uses Kullback-Liebler distance algorithm to measure the correlation between documents, or between a document and a cluster. The objective function follows:

$$KL(d,c)=\sum_{w_i\in d}\frac{n(w_i,d)}{|d|}\log\frac{n(w_i,d)/|d|}{n(w_i,c)/|c|} \qquad (1)$$

where $KL(d,c)$ defines the degree of correlation between a document (or a collection of documents: cluster) d and a document (cluster) c. A low $KL(d,c)$ value indicates that the document (cluster) is highly correlated to another document (cluster). Naturally, these documents are to be clustered into the same cluster, which leads to $KL(d,d)=0$.

The $n(w_i,x)$ is the document-concept weight, which is a measure of the number of occurrences of concept w_i in the document (cluster) x.

$|x|$ is the number of concepts in the document (cluster) x.

In order to generate the clusters automatically, we need to find a set of thresholds to differentiate the correlated documents from others and form a cluster. The main tasks in finding the thresholds include: select a document, calculate the KL distance between this document and each document in the collection, arrange the documents with increasing KL. Finally, uses a function, $f(x)$, to simulate all points. The KL distance at the inflexion point of $f(x)$ is one threshold.

HNCIR adopts the "least squares" approach. HNCIR obtains $f(x)$ by minimizing the following function:

$$\delta = \sum_{i=1}^{n} \delta_i^2 = \sum_{i=1}^{n} [f(x_i) - y_i]^2 \qquad (2)$$

Where δ_i is the distance difference between point (x_i, y_i) and $f(x_i)$; $f(x) = c_0 + c_1x^1 + c_2x^2 + \ldots + c_mx^m$

Equating the partial derivatives of equation (2) with respect to c_j to 0, we will obtain m+1 equations. Solutions to these equations solve the function $f(x)$.

The inflexion point of $f(x)$ is obtained using $f''(x)$.

The basic idea of Cluster generation is as follows: (a) Selecting a document from a collection randomly, (b) measuring the distance between this document and other documents in the collection based on Eq. (1), (c) calculating the distance threshold, (d) marking off a cluster form the collection according to the calculated threshold, (e) removing the clustered documents from the collection, (f) repeating above steps till all the documents are processed or a stop condition is reached.

The generation process can be concluded to the following steps:

Step 1: selecting a document d from collection randomly.

Step 2: calculating the distance kl_{cj} between document *d* and other documents c_j in the collection *S* based on (1). A set of distance can be obtained: $KL_d = \{kl_{c1}, kl_{c2}, kl_{c3}, \ldots, kl_{cm}\}$.

Step 3: if $\min\{kl_{c1}, kl_{c2}, kl_{c3}, \ldots, kl_{cm}\} > MAXKL$, setting cluster *P*, *P*=$\{d\}$, go to Step 5.

Step 4: calculating the distance threshold $kl_{threshold}$ via $KL_d = \{kl_{c1}, kl_{c2}, kl_{c3}, \ldots, kl_{cm}\}$ according to section 3.1.2, then setting a document d-centered cluster *P*, $P = \{cn \mid kl_{cn} <= kl_{threshold}, cn \in S\} + \{d\}$.

Step 5: removing the clustered documents from collection *S*, namely *S* = *S*-*P*.

Step 6: checking whether the collection *S* is null, if it is null, go to Step8.

Step 7: selecting another document *d* which KL distance equal to $\max\{kl_{c1}, kl_{c2}, kl_{c3}, \ldots, kl_{cm}\}$ in collection *S*, repeat Steps 2-7.

Step 8: checking whether the number of clusters is within the range of an appropriate value settled in advance, if it is, go to Step17.

Step 9: selecting a cluster *P* from cluster collection *SC*.

Step 10: calculating the distance kl_{cj} between cluster *P* and other cluster p_j in the cluster collection *SC* based on equation (1).

Step 11: if $\min\{kl_{c1}, kl_{c2}, kl_{c3}, \ldots, kl_{cm}\} > MAXKL$, go to Step 13.

Step 12: calculating the distance threshold $kl_{threshold}$ via $KL_d = \{kl_{c1}, kl_{c2}, kl_{c3}, \ldots, kl_{cm}\}$ according to section 3.1.2, then merging and producing an new cluster *P*, $P = \{cn \mid kl_{pi} <= kl_{threshold}, cn \in Pi\} + P$.

Step 13: removing the cluster *P* from cluster collection *SC*.

Step 14: checking whether there are no unprocessed clusters in *SC*, if there are, go to Step 16.

Step 15: selecting another cluster *P* which KL distance equal to $\max\{kl_{p1}, kl_{p2}, kl_{p3}, \ldots, kl_{pm}\}$ in cluster collection *SC*, repeat Steps 10 - 15.

Step 16: checking whether the number of clusters is obviously decreased, if it is, repeat steps 8-16.

Step 17: calculating every distance between each document d in the collection and the cluster *Pj*, let $kl_{pi} = \min\{kl_{p1}, kl_{p2}, kl_{p3}, \ldots, kl_{pk}\}$, if document *d* does not belong to *Pi*, remove document d to cluster *Pi*

Step 18: the process is completed.

The whole process can be divided into 3 stages. The first stage is Steps 1- 7. It initializes the cluster. The second stage is Steps 8 - 16, which merges and rebuilds the clusters via iterated processing. The last stage is Step 18, which adjusts the results obtained from the second stage. The first stage is similar to the second stage. These two stages can be combined when coded into a program.

3.2 Cluster Searching

A searching method is proposed to complete the cluster searching based on Dirichlet smoothing method. Our method uses the linear interpolation technique processing individual maximum likelihood, partial maximum likelihood and global maximum likelihood to rank the documents with respect to the given query. Due to this, a new language model is introduced. Different from the traditional collection model, this model treats the unseen words in an indirect and clustering-based way.

In the same way, HNCIR uses word concept as the processing object. However, it is hard to confirm the concept of each query exactly for separate queries always lack strong phraseological relationship between each other. Therefore, we use the proportion of each concept candidate of a word to weighting the probabilities caused by each candidate.

The target function of HNCIR searching method follows:

$P_{Dir}(d_i \mid Q) = \prod_{q_j \in Q} \sum_n P(q_j, w_{jn}) \times \frac{n(w_{jn}, d_i) + \mu \sum_P P(w_{jn} \mid P_s) P(P_s \mid C)}{\mid d_i \mid + \mu}$ where $n(w_{jn}, d_i)$ is a measure of the number of occurrences of concept w_{jn} in document (cluster) d_i.

$\mid d_i \mid$ is the number of concepts in document (cluster) d_i.

$P(P_s \mid C)$ is the appearance probability of the cluster P_s under the clustering distribution of $P_1, P_2, \ldots, P_m$. $P(P_s \mid d_i) = \frac{1 - kl_{Ps}}{\sum_{Pk=P1}^{Pm} (1 - kl_{Pk})}$, where P_s refers to a certain cluster in $P_1, P_2, \ldots, P_m$. kl_{P_k} is the KL distance between document d_i and cluster P_k. Each distance has been calculated in Step 17 of section 3.1.3.

$P(w_{jn} \mid P_s)$ is a measure of the frequency of the key concept w_{jn} in cluster P_s. $P(w_{jn} \mid P_s) = \frac{n(w_{jn}, P_s)}{\mid P_s \mid}$.

$P(q_j \mid w_{jn})$ is the probability of word q_j translating to concept w_{jn}. $P(q_j \mid w_{jn}) = \frac{n(w_{jn}, q_j)}{n(w_{jn}, S)}$, where $n(w_{jn}, q_j)$ is a measure of how many concept w_{jn} are generated by word q_j; $n(w_{jn}, S)$ is the number of concept w_{jn} in the document collection *S*.

μ is Dirichlet parameter

4 EXPERIMENTAL RESULTS

Our experiments are made in a Chinese language circumstance. The test collections are chosen from TREC6. A 164,811 document collection including documents from both the People's Daily and the Xinhua News Agency was used.

There are many parameters in HNCIR that needs to be set. We set these parameters according to the performance of HNCIR in our experiments. We selected the values which lead to the best performance of system as their optimum values.

Firstly, the range of the value which is used to compare with the number of clusters in Step 8 of section 3.13 needs to be evaluated. The range of value is an order of magnitude and determined by the number of documents in the document collection. The algorithm is shown as follows.

$$V_{range} = \begin{cases} O(10\log D_{number}) & D_{number} \geq 1000 \\ O(D_{number}/50) & D_{number} < 1000 \end{cases}$$

where D_{number} is the number of the documents in the document collection; V_{range} is the value of range.

Secondly, the condition which is used to check whether the number of clusters is obviously decreased should be evaluated. The condition is whether the highest digit of a number varies. If it is, it is reasonable to believe that the number of clusters is obviously decreased.

Thirdly, the value of threshold MaxKL used to detect the isolated point should be fixed. The actual test results indicated that the algorithm will have a good tolerance of isolated points if the value of *MaxKL* is within the rage form 0.8 to 0.9, and the change of the value in this range will affect marginally the final results. In our experimental system, we chose the value as 0.85.

Lastly, the value of Dirichlet parameter μ should be evaluated. The results obtained by different queries are differently affected by the parameter μ. We tested a set of values of μ, and found that the highest performance of the system occurred when μ equal to 200, so we chose 200 as the standard value of μ in our experimental system.

In order to make comparison, two other IR models are also tested in our experiment. One is Jelinek-Mercer Model based on Term Frequency, the other is Bayesian Model based on Term Frequency. The test results of these 3 systems are given in Table 3.

Table 3. The Test Results

Precision Recall	Jelinek-Mercer	Bayesian	HNCIR
0	0.6000	0.6000	0.6100
0.1	0.5416	0.5521	0.5760
0.2	0.4812	0.4256	0.5164
0.3	0.4028	0.4229	0.4821
0.4	0.3837	0.4042	0.4478
0.5	0.2586	0.2776	0.4010
0.6	0.2103	0.2521	0.3130
0.7	0.1227	0.1731	0.2245
0.8	0.0834	0.1237	0.1804
0.9	0.0491	0.0687	0.0610
1	0.0312	0.0310	0.0306
AvgPre	24.95%	26.61%	34.93%

The experiment shows the precision of HNCIR increases more evidently than the other two systems when the recall increases. The reason of getting the good results of HNCIR is the word sense ambiguities are well solved by translating the words into their concept forms via the strategy of word concept extracting. On the other side, the cluster generation method can divide the clusters equably and can well serve our cluster searching method. Otherwise, we also found that the query results of HNCIR involved some inappropriate documents that consist of a lot of complex sentences which can not be analyzed accurately. It is an accessional factor that depresses the system behavior. Another factor that will affect the results is how to tackle the new words in the documents and query, this problem have been solved by finding the words via concept extracting strategy and processing them as an integrated concept.

5 CONCLUSIONS

One of the main difficulties in information retrieval is how to resolve the ambiguities of the meanings which the word represents. These ambiguities become more serious in the translation of the query terms in IR. In this paper, we use word extracting strategy to solve the word sense ambiguities and get the exact concept of the word. We also designed an auto threshold-detecting based clustering method to process the concepts. The reason of using the clustering method is that we are not able to extract the document conceptual structures well at present. Our team will focus on this issue in the following years. The test results show that the information retrieval based on concept can efficiently improve the IR precision in Chinese.

ACKNOWLEDGMENTS We would like to thank Huang ZengYang, George Qian, Efthimiadis and the anonymous reviewers for helpful comments on this work. This research was supported by the National Basic Research Program of China (973 Program, the contract No. 2004CB318104) and the Knowledge Innovation Program of Chinese Academy of Sciences (the contract No. 13CX04, the Institute of Acoustics).

References

1. Salton, G.. Automatic information organization and retrieval. New York: McGraw-Hill. 1968.
2. Van Rijsbergen, C. J. Information retrieval (2nd ed.). London: Butterworths. 1979.
3. Salton, G., & McGill, M. J. Introduction to modern information retrieval. New York: McGraw-Hill. 1983
4. Baeza-Yates, R., & Ribeiro-Neto, B. Modern information retrieval. Addison-Wesley. 1999
5. Bailey, P., Craswell, N., & Hawking, D. Engineering a multi-purpose test collection for Web retrieval experiments. Information Processing and Management, 39, 853–871. 2003.
6. Miyamoto, S. Fuzzy sets in information retrieval and clustering analysis. Kluwer Academic Press. 1990.
7. Crestani, F., Pasi, G. (Eds.). Soft computing in information retrieval. Germany: Physica Verlag and Co, ISBN:3790812994, 102–121. 2000
8. Belew, R. Finding out about. Cambridge University Press. 2000
9. Lalmas, M. Logical models in information retrieval: Introduction and overview. Information Processing and Management, 34(1), 19–33. 1998
10. Meadow, C. T., Boyce, B. R., & Kraft, D. H. Text information retrieval systems (2nd ed.). San Diego, CA: Academic Press. 1999
11. Schank R. Identification of conceptualizations underlying nature lauguage. In: Schank R, Colby K Eds. Computern Models of Thought and Language. San Francisco, CA: W H Freeman and Company. 1973
12. Schank R. Conceptual Information Processing. Amsterdam: North Holland. 1975
13. Schank R. The structure of episodes in money. In: Bobrow D, Collins A eds. Representation and Underastanding. New York: Academic Press
14. Schank R. Dynamic Memory. New York: Cambridge University Press. 1982
15. Schank R, Abelson R. 1977. Scripts, Plans, Goals and Understanding. Hillsdale, NJ: Erlbaum. 1977
16. HUANG Zengyang. HNC (Hierarchical Network Concept) Theory. Beijing: Tsinghua University Press. 1998
17. HUANG Zengyang. Mathematics and physics symbol system of language in language concept space. Beijing: Ocean Press. 2004
18. Miao Chuanjiang. Guide of HNC (Hierarchical Network Concept) Theory. Beijing: Tsinghua University Press. 2005
19. Zhang Quan, Xiao Guozheng. HNC and NLP (Nature Language Processing). Wuhan: Wuhan Institute of Technology. 2001
20. Fillmore C J. The case for case. In: Bach E, Harms R eds. Universals in Linguistic Theory. New York: Holt, Rinehart and Winston. 1968
21. Chomsky N. 1957. Syntactic Structures. Hague: Mouton. 1957
22. Chomsky N. 1965. Aspects of the Theory of Syntax. Cambridge, MA: MIT Press 1965
23. Fellbaum, C. WordNet: an electronic lexical database. The MIT Press. 1998
24. Agosti, M., Crestani, F., & Pasi, G. Lectures on information retrieval. Lecture notes in computer science 1980. Alltheweb. (2004). Available: http://www.alltheweb.com. 2001
25. Kohonen, T. Self-organizing maps (2nd ed.). Germany: Springer-Verlag. 1997