Computational Linguistics and Chinese Language Processing
Vol. 7 , No. 2, August 2002, pp. 37-58

A Study on Word Similarity using Context Vector Models

Keh-Jiann Chen*, Jia-Ming You*

Abstract

There is a need to measure word similarity when processing natural languages, especially when using generalization, classification, or example-based approaches. Usually, measures of similarity between two words are defined according to the distance between their semantic classes in a semantic taxonomy. The taxonomy approaches are more or less semantic-based that do not consider syntactic similarities. However, in real applications, both semantic and syntactic similarities are required and weighted differently. Word similarity based on context vectors is a mixture of syntactic and semantic similarities.

In this paper, we propose using only syntactic related co-occurrences as context vectors and adopt information theoretic models to solve the problems of data sparseness and characteristic precision. The probabilistic distribution of co-occurrence context features is derived by parsing the contextual environment of each word, and all the context features are adjusted according to their IDF (inverse document frequency) values. The agglomerative clustering algorithm is applied to group similar words according to their similarity values. It turns out that words with similar syntactic categories and semantic classes are grouped together.

1. Introduction

It is well known that word-sense is defined by a word's co-occurrence context. The context vectors of a word are defined as the probabilistic distributions of its left and right co-occurrence contexts. Conventionally, the similarity between two context vectors is measured based on their cosine distance [Alshawi and Cater, 1994; Grishman and Sterling, 1994; Pereira *et al*., 1993; Ruge, 1992; Salton, 1989]. However, the conventional measurement

* Institute of Information Science, Academia Sinica
E-mail: kchen@iis.sinica.edu.tw; swimming@hp.iis.sinica.edu.tw

suffers from the following drawbacks. First of all, the information in the context vectors is vague. All co-occurrence words are collected without distinguishing whether they are syntactically or semantically related. Second, the coordinates are not pair-wise independent (i.e., the axes are not orthogonal), and it is hard to apply singular value decomposition to find the orthogonal vectors [Schutze, 1992]. In this paper, we propose to use only syntactic related co-occurrences as context vectors [Dekang Lin, 1998] and adopt information theoretic models to solve the above problems. In our study, the context vectors of a word are defined as the probabilistic distributions of its thematic roles and left/right co-occurrence semantic classes. The context features are derived from a treebank. All context features are weighted according to their TF×IDF values (the product of the term frequency and inverse document frequency) [Salton, 1989]. For the context features, the Cilin semantic classes (a Chinese thesaurus) are adopted. The Cilin semantic classes are divided into 4 different levels of granularity. In order to cope with the data sparseness problem, the weighted average of the similarity values at four different levels will be the similarity measure of two words. The weight for each level is equal to the information-content of that level[Shannon, 1948; Manning and Schutze 1999].

A agglomerative clustering algorithm is applied to group similar words according to the above defined similarity measure. Obviously, words with similar behavior in the corpus will be grouped together. We have compared the clustering results to the Cilin classifications. It turns out that words in the same synonym class and with the same syntactic categories have higher similarity values than the words with different syntactic categories.

2. Data Resources

Ideally, to derive context vectors, a large corpus with semantic tags is required. Furthermore, to extract co-occurrence words along with their exact syntactic and semantic relations, the corpus structure has to be annotated. However, such an ideal corpus does not exist. Therefore, in this paper we will adopt the resources that are available and try to derive a useful but imperfect Chinese tree bank. Since the similarity measure based on the vector space model is a rough estimation, minor errors made at the stage of context vector extraction are acceptable.

2.1 Sinica Corpus

The Sinica corpus contains 12,532 documents and nearly 5 million words. Each sentence in the corpus was parsed by a rule parser [Chen, 1996]. The parsed trees were tagged with the structure brackets, syntactic categories and thematic roles of each constituent [Huang *et al*., 2000] as exemplified below, (Sinica corpus: http://www.sinica.edu.tw/ftms-bin/kiwi.sh):

Original sentence: **小** 'small' **狗** 'dog' **跳舞** 'dance'

Parsed tree : S(Agent:NP:(Property:Adj:'**小** small' | Head: N: '**狗** dog')|Head:V: **跳舞** dance')

Although these labels may not be exactly correct, we believe that, even with these minor errors, the majority of word-to-word relations extracted from the trees are correct. However, the semantic label is not provided for each word in the parsed trees. In this paper, we will use Cilin classifications for semantic labeling.

2.2 Cilin- a Chinese Thesaurus

Cilin provides the Mandarin synonym sets in a hierarchical structure [Mei *et. al.*, 1984]. It contains 51,708 Chinese words, and 3918 classes. There are five levels in the Cilin semantic hierarchy, denoted in the format L_1-L_2-L_3-L_4-L_5. For example, the Cilin class of the word **我們** 'we' is "A-a-02-2-01". In level 1, "A", denotes the semantic class of human; in level 2, "a", indicates a group of general terms; level 3, "02", means pronouns in the first person, and in level 4, "2" represents the plural property. In level 5, "01" represents the order rank in the level 4 group. This means that "01" in level 5 is the first prototypical concept representation of "A-a-02-2". In the rest of this paper, only the first four levels will be used. The fifth level is for sense disambiguation only (section 2.3).

2.3 Sense Disambiguation

A polysemous word has more than one Cilin semantic class. In order to tag appropriate Cilin classes, we have designed a simple sense tagging method as follows [Wilks, 1999]. The sense tagging algorithm is based on the facts that the syntactic categories of each word in the tree bank are assigned uniquely, and that each Cilin class has its own major syntactic category. If a word has multiple Cilin classes, we select the sense class whose major syntactic category is the same as the tagged category of this word. For example, **計畫** "Jie-Hwa" has two meanings. One is for "project" as a noun and the other is "attempt", therefore, if **計畫** "Jie-Hwa" was tagged with a noun category, we will assign the Cilin class whose major category is "project". Sense ambiguity can be distinguished by measure of syntactic properties for most words. However, there are still cases in which the syntactic category constraints cannot resolve the sense ambiguities. Then, we simply choose the prototypical sense class, i.e, the word that has the highest rank in this sense class with respect to all its sense classes in Cilin.

2.4 The Extraction of Co-occurrence Data

The extracted syntactically related pairs have either a head-modifier relation or head-argument relation. For instance, two syntactically related pairs extracted from the example in section 2.1 are:

(<Thematic role > <Cilin> <word1>), (<Thematic role> <Cilin> <word2>)

(agent Bi-07-2 **狗** 'dog'), (Head(S) Hh-04-2 **跳舞** 'dance')

(property Ea-03-3 小'small'), (Head(NP) Bi-07-2 狗'dog')

The context data of the word$_1$ 狗 "dog" consists of its thematic role "agent" and the Cilin class "B-i-07-2" ; the word$_2$ 跳舞 'dance' consists the thematic role "Head(S)" and the Cilin class "H-h-04-2" and so on. The word 小 "small" and 跳舞 "dance" are not syntactically related even though they co-occur. Therefore, they will not be extracted.

3. Context Vector Model

There are three context vectors of a word: role vector, left context vector and right context vector. The role vector is a fixed 48-dimension vector, and each dimension value is equal to the probabilistic distribution of its thematic roles. The left/right context vectors are closer to the probabilistic distributions of its left/right co-occurrence words and their semantic classes. The role vector characterizes a word based more on syntax and less on semantics, but the left/right context vectors are just the opposite. The cosine distance between their context vectors is a measure of the similarity of the two words. We will illustrate the derivations of context vectors and their similarity rating with a simplified example using (貓 "cat", 狗 "dog"). The role vector of "dog" is {127, 207, 169…, 0}$_{48}$, which represents the values of "agent", "goal", "theme"… and "topic" respectively, generated by Equation (1). The role vector of "cat" is {28, 73, 56…, 0}$_{48}$, which is also acquired by Equation (1).

Role vector of word W = { V_1, V_2,…,V_{48} }$_{48}$

Vi = Frequency (R_i) × log ($1/P_i$) (1)

R_i: We label thematic roles "agent", "goal"… and "topic" from R_1 to R_{48} listed in Table2 in the Appendix.

Frequency (R_i): The frequency of R_i played by word W in the corpus.

P_i: = Total frequency of R_i in the corpus / Total frequency of all roles in the corpus.

log($1/P_i$): The information–context of R_i [Shannon, 1948; Manning and Schutze, 1999]

The derivation of left/right context vectors is a bit more complicated. The syntactically related co-occurrence word pairs are derived first as illustrated in section 2.4. We will illustrate the derivation of the left context vector only. The right context vector can be derived similarly. The left co-occurrence word vector of word W is generated from frequency(word$_i$), where word$_i$ precedes and is syntactically related to W in the corpus. Due to the data sparseness problem, the feature dimensions of context vectors are generalized into Cilin classes instead of co-occurrence words. The generalization process reduces the effect of data sparseness. On the other hand, it also reduces the precision of characterization since each

word has different information content and two words that have the same co-occurrence semantic classes may not share the same co-occurrence words. In order to resolve the above dilemma, when we compare the similarity between $word_X$ and $word_Y$, 4 levels of left context vectors and right context vectors for $word_X$ and $word_Y$ are created[1]. The weighting of each feature dimension is adjusted using the TF*IDF value if $word_X$ and $word_Y$ have shared context words. Equation (2) illustrates the creation of the 4th level left context vector of $word_X$. The other context vectors for $word_X$ and $word_Y$ are created by a similar way.

Left context vector of $word_X$ = $\{f1, f2, \ldots, f_{3918}\}_{L4}$

Where fi = Sum of

$$\begin{cases} TF(word_j) \times IDF(word_j) & \text{if } word_X \text{ has the same neighbor } word_j \text{ with } word_Y \\ TF(word_j) & \text{if } word_X \text{ does not have the same neighbor } word_j \text{ with } word_Y \end{cases}$$

; for every left co-occurrence $word_j$ with the *i*th Cilin semantic class. (2)

$TF(word_j)$: *the frequency of pair ($word_j$, Cilin($word_j$)) in wordx's co-occurrence context.*

$IDF(word_j)$: *-log(the number of the documents that contains the $word_j$/total document number of the corpus)*

In Equation (2), we adjust the term weight using TF×IDF, which is commonly used in the field of information retrieval [Salton, 1989] to adjust the discrimination power of each feature dimension. We will examine the difference in the adjustment of weights using TF×IDF and TF in section 4.2. We will next give a simplified example. Assume that the word **狗** "dog" has only three left syntactically related words: (**小** "small" Ea033) with frequency 30, (**可愛** "cute" Ed401) with frequency 5 and (**養** "raise" Ib011) with frequency 10; and assume that the word **貓** "cat" has only two left syntactically related words: (**黑** "black" Ec043) and (**養** "raise" Ib011). Assume that we are measuring the similarity between **狗** "dog" and **貓** "cat". Then, we can compute the left context data of **狗** "dog" as $\{TF(Aa011), \ldots, TF(Ea033), \ldots, TF(Ed041), \ldots, TF(Ib011) \times IDF(\textbf{養}\text{'raise'})$[2], …, $TF(La064)\}_{L4}$[3]

---

[1] IDF(**養**'raise') = 4.188, the IDF values of all words range from 0.19 to 9.12.

[2] The granularities of the 4 levels of semantic classes are partially shown in Figure2. The four left context vectors and their dimensions are shown below and the right context vectors are similarly derived.

<LeftCilin1>$_{L1}$ A vector of 12 dimensions from "A" to "L".

<LeftCilin2>$_{L2}$ A vector of 94 dimensions from "Aa" to "La".

<LeftCilin3>$_{L3}$ A vector of 1428 dimensions from "Aa01" to "La06".

<LeftCilin4>$_{L4}$ A vector of 3918 dimensions from "Aa011" to "La064".

$= \{0_{Aa011},\ldots,30_{Ea033},\ldots,5_{Ed041},\ldots,10\times 4.188_{Ib011},\ldots 0_{La064}\}_{L4}$

$= \{0_{Aa011},\ldots,30_{Ea033},\ldots 5_{Ed041},\ldots,41.88_{Ib011},\ldots 0_{La064}\}_{L4}$, since ,they share only the same left context word **養** "raise". The other levels of left context vectors of **狗** "dog" are $\{0_{Aa01},\ldots,30_{Ea03},\ldots 5_{Ed04},\ldots,41.88_{Ib01},\ldots 0_{La06}\}_{L3}$, $\{0_{Aa},\ldots,30_{Ea},\ldots 5_{Ed},\ldots,41.88_{Ib},\ldots 0_{La}\}_{L2}$, $\{0_{A},\ldots,35_{E},\ldots,41.88_{I},\ldots 0_{L}\}_{L1}$. The value of the E dimension is 35 because it is the sum of the values of Ea(30) and Ed(5) from $\{\ldots,30_{Ea},\ldots 5_{Ed},\ldots\}_{L2}$.The right context vectors of $<\text{RightCilin1}>_{R1}$ to $<\text{RightCilin4}>_{R4}$ are derived in a similar way.

3.1 Similarities between Two Context Vectors

Once we know the feature vectors of these two words, we can calculate the cosine distance of two vectors as shown in Equation (3).

vector A= $<a1,a2\ldots,an>$,vector B= $<b1,b2\ldots,bn>$

$$\cos(A,B)=\frac{\sum_{i=1}^{n} ai\times bi}{\sqrt{\sum_{i=1}^{n} ai^{2}}\times\sqrt{\sum_{i=1}^{n} bi^{2}}} \quad \ldots \qquad (3)$$

Therefore, the similarity of the two words x and y can be calculated as the linear combination of the cosine distances of all the feature vectors as shown in the Equation 4. The weight of each feature vector can be adjusted according to different requirements. For instance, if the syntactic similarity is more important, we can increase the weight w. On the other hand, if the semantic similarity is more important, the weights w1 to w4 can be increased. If more training data is available, the level 4 vector will be more reliable. Hence, the weight w4 should increase.

$$\begin{aligned}
&\text{similarity}(x,y)=w\times\cos(<\text{role vector}>x,<\text{role vector}>y)\\
&+w1\times\{w11\cos(<\text{LeftCilin1}>x,<\text{LeftCilin1}>y)+w12\cos(<\text{RightCilin 1}>x,<\text{RightCilin 1}>y)\}\\
&+w2\times\{w21\cos(<\text{LeftCilin2}>x,<\text{LeftCilin2}>y)+w22\cos(<\text{RightCilin 2}>x,<\text{RightCilin 2}>y)\}\\
&+w3\times\{w31\cos(<\text{LeftCilin3}>x,<\text{LeftCilin3}>y)+w32\cos(<\text{RightCilin 3}>x,<\text{RightCilin 3}>y)\}\\
&+w4\times\{w41\cos(<\text{LeftCilin4}>x,<\text{LeftCilin4}>y)+w42\cos(<\text{RightCilin 4}>x,<\text{RightCilin 4}>y)\}
\end{aligned} \qquad (4)$$

$$wk1 = \frac{|< \text{LeftCilin K} > x| + |< \text{LeftCilin K} > y|}{|< \text{LeftCilin K} > x| + |< \text{LeftCilin K} > y| + |< \text{RighttCilin K} > x| + |< \text{RightCilin K} > y|}$$

$$wk2 = \frac{|< \text{RightCilin K} > x| + |< \text{RightCilin K} > y|}{|< \text{LeftCilin K} > x| + |< \text{LeftCilin K} > y| + |< \text{RighttCilin K} > x| + |< \text{RightCilin K} > y|}$$

$k = 1,2,3,4$

$w + w1 + w2 + w3 + w4 = 1$

$|<$ vector $>|$ means the vector length

In the experiments, w = 0.3, w1 = 0.1×0.7, w2 = 0.1×0.7, w3 = 0.4×0.7, and w4 = 0.4×0.7.

4. Similarity Clustering

Because of the lack of objective standards for evaluating of similarity measures, a agglomerative clustering algorithm is applied to group similar words according to a similarity value. It turns out that words with similar syntactic usage and similar semantic classes are grouped together. We will evaluate our algorithm by comparing the automatic clustering results with manual classifications of Cilin.

4.1 Clustering Algorithm

To evaluate the proposed similarity measure, we tried to group words according to various parameters. We adopted bottom-up agglomerative clustering algorithm to group words. In order to compare the clustered results with Cilin classifications and reduce the data sparseness, we picked the 1000 highest frequency words in Cilin for testing. First of all, we produced a 1000×1000 symmetric similarity matrix called SMatrix, where SMatrix (x, y) = similarity ($word_X$, $word_Y$), for all x < y. The rest of the matrix was set to - INFINITY. Below are the details of the clustering algorithm

---

Bottom-up Agglomerative Clustering (the greedy algorithm):

Initialize:

Assign the threshold; (a value ranging from 0.1 to 0.85)

Assign each word to its own group named Group(word)

Loop

Find the entry [x,y] of SMatrix with the maximal value and let the value be M = SMatrix[x][y];

If M is less than the threshold

exit loop

else

Grouping ($word_X$, $word_Y$)

Recalculate SMatrix

End Loop

Grouping ($word_X$, $word_Y$)

Merge Group($word_Y$) to Group($word_X$)

Recalculate SMatrix

{ Smatrix (i)(y) = Smatrix (y)(j)= -INFIITE, where j != y, i != y

$$SMatrix[x][i] = \frac{\sum_{p=1}^{m}\sum_{q=1}^{n} similarity(word_p, word_q)}{m \times n}$$

$0 < x < i$

Group (wordx) contains $word_p, 1 \le p \le m$

Group (wordj) contains $word_q, 1 \le q \le n$

$$SMatrix[j][x] = \frac{\sum_{p=1}^{m}\sum_{q=1}^{n} similarity(word_p, word_q)}{m \times n}$$

$0 < j < x$

Group (wordx)contains $word_p, 1 \le p \le m$

Group (wordj) contains $word_q, 1 \le q \le n$

}

4.2 Clustering Results vs Cilin Classification

We will make a comparison between the clustering results and Cilin classifications. There are two simple examples shown in Figure 1 and Figure 2 representing the clustering results and Cilin classifications, respectively.

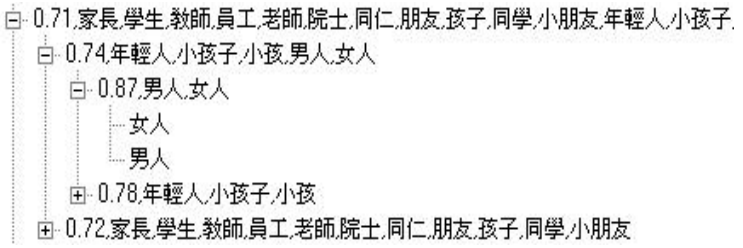

Figure 1 Clustering results with threshold = 0.7

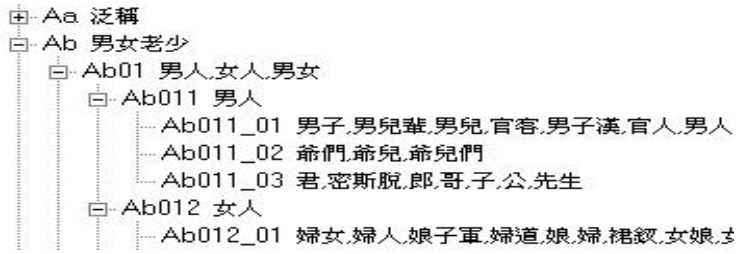

Figure 2 Examples of Cilin classification

After the clustering algorithm is applied, the words are distributed into m groups, i.e., $Group_1$, $Group_2$, $Group_3$…, $Group_m$. Then, we can define the recall and precision of the classification as follows.

$$\text{recall } k = \frac{\sum_{i=1}^{m} Gki}{\text{The number of words that are clustered in the kth level of Cilin}}, (5)$$

$$\text{precision } k = \frac{\sum_{i=1}^{m} Gki}{\text{The number of words that are clustered by this greedy algorithm}}, (6)$$

Gki = representing the maximum number of words in Groupi that are classified in the same Cilin class in level k.

Among our 1000 testing words, the number of words that were clustered in the 4^{th} level of Cilin was 658; i.e., they were labeled with 459 different level-4 Cilin classes and among

them, 342 classes contained only one testing word, and the classes with multiple testing words contained a total of 658 testing words. With the threshold=0.7, our method clustered 830 words, and only 167 words of them were clustered in the correct Cilin class. Therefore, by Equation (5), recall 4 = 167/658 = 0.25 and with Equation (6) precision 4 = 167/830 = 0.20.

We adopted two methods for measuring similarity; one used Equation TF×IDF, and the other used Equation TF. The results are shown in Figure 3 to Figure 6 in the Appendix. We measured the performance by computing the F-score, which is (recall+precision)/2. We discovered that the best F-score of level1 was that 0.7648 located at a threshold equal to 0.65, the best F-score of level2 was that 0.5178 located at a threshold equal to 0.7, the best F-score of level3 was that 0.3165 located at a threshold equal to 0.8, and that the best F-score of level4 was that 0.2476 located at a the threshold equal to 0.8. All were obtained using TF×IDF strategy. Hence, we can see that the TF×IDF equation achieves better performance than the TF equation does. We list the detailed F-socre data for various parameters in appendix. Although the clustering results didn't fit Cilin completely, they are still alike to some degree. From the results, we find that they are similar to syntax taxonomy under a lower threshold and close to semantic taxonomy under higher thresholds.

5. Cilin classifications re-examined

To examine the practicability of our proposed method, we also inspected the similarity values of these 658 testing words which were clustered into 117 4th level Cilin classes. For each semantic class, the average similarity between words in the class and their standard deviation was computed. The results are listed in Table1 in the Appendix. We expected that synonyms would have high similarity values, but this was not always the case.

According to the assumption noted above, synonyms might have similar syntactic and semantic contexts in language use. Therefore, the average similarity should be pretty high, and the standard deviation should be quite low. However, some of the results didn't follow the assumption. We analyzed the data offer explanations in the following.

a) Synonyms with different POS: Words with the same semantic classification in Cilin could have different parts of speech (POS). (as shown below.)

Word set	Average similarity	Stand deviation
思想,考量 考慮,思考	0.544195	0.229534

The contexts of the noun (**思想**, "thinking") and the verbs (**考慮**,**考量**,**思考**, "think", "consider", "deliberate") were quite different. As a result, the average

similarity value was quite low, and the standard deviation was very high. After we removed the noun from the word-set, we recomputed the values and obtained the table shown below:

Word set	Average similarity	Stand deviation
考量,考慮 思考	0.770616	0.0429353

The results conform to our assumption. They also reveal that the context of synonyms may vary from POS to POS.

b) Error in Cilin Classification: The classifications in Cilin could be arbitrary. For example, the three words, **數量** "quantity", **多少** "how many" and **人數** "the number of people", were classified in a Cilin group. They might be slightly related, but grouping them together seems inappropriate according to the following table:

Word set	Average similarity	Stand deviation
數量,多少 人數	0.379825	0.253895

c) Different uses: Differences in their usage cause synonyms to behave differently. For example, when we measured the similarity of **美國** "America" to **日本** "Japan" and to **中國** "China", the results we obtained were 0.86 and 0.62,respectively, for each pair. Accroding to human intuition, they simply refer to names of countries and should not have such different similarity values. The reason for these result is that the corpus we adopted is an original Taiwan corpus. As a result, the usage of **中國** "China" is different from that of **美國** "America" and **日本** "Japan".

d) Polysemy: The word senses that Cilin adopted were not those frequently used in the corpus. See the following table:.

Word set	Average similarity	Stand deviation
十分,非常, 特別	0.45054	0.305209

Although the three words, **十分** "very/ten points", **非常** "very" and **特別**"special, extraordinary" might seem to be very close in meaning to "very", the polysemous word **特別** "special, extraordinary" is different in its major sense. This influenced the result.

e) Words with similar contexts might not be synonyms: A disadvantage does exist when the context vector model is used. Words that are similar in terms of their contexts might not be similar in meaning. For example, the similarity value of 結婚 "marry" and 長大 "grow" is 0.8139. Although the two words have similar contexts, they are not alike in meaning. Therefore, the vector space model should incorporate the taxonomy approach to solve this phenomenon.

6. Conclusions

In this paper, we have adopted the context vector model to measure word similarity. The following new features have been proposed to enhance the context vector models: a) The weights of features are adjusted by applying TF×IDF. b) Feature vectors are smoothed by using Cilin categories to reduce data sparseness. c) Syntactic and semantic similarity is balanced by using related syntactic contexts only.

The performance of our method might have been influenced by the small scale of the Chinese corpus and accuracy of the extracted relations. Further more, Cilin was published a long time ago and has not been update recently, which may have influenced our results. However, our experimental results are encouraging. They supports the theory that using context vectors to measure similarity is feasible and worthy of further research.

References

Alshawi and Carter (1994) "Training and scaling preference functions for disambiguation." *Computational Linguistics*,20(4):635-648

Chen, K.J. (1996) "A Model for Robust Chinese Parser", *Computational Linguistics and Chinese Language Processing*, Vol. 1, pp.183-204.

Grishman and Sterling (1994) "Generalizing automatically generated selectional patterns." *In Proceeding of COLING-94*, pages 742-747, Kyoto, Japan.

Huang, Chu-ren, F.Y. Chen, Keh-Jiann Chen, Zhao-ming Gao, and Kuang-Yu Chen (2000) " Sinica Treebank: Desigm Criteria, Annotation Guidelines and On-line Interface", *Proceedings of ACL workshop on Chinese Language Processing*, pp.29-37.

Lin, Dekang(1998) "Automatic Retrieval and Clustering of Similar Words" *COLING-ACL98*, Montreal, Canada.

Manning, Christopher D. & Hinrich Schutze (1999) *Foundations of Statistical Natural Language Processing*, the MIT Press, Cambridge, Massachusetts.

Mei, Gia-Chu etc., 1984 同義詞詞林.(Cilin - thesaurus of Chinese words). Hong Kong, 商務印書館香港分館.

Miller, George A., and Walter G. Charles (1991) " Contexture Correlates of Semantic Similarity," *Language and Cognitive Processes* 6:pp.1-28.

Salton, Gerard (1989) Automatic Text Processing: The Transformation, Analysis, and Retrieval of Information by Computer, Reading, MA: Addison Wesley.

Ruge (1992) "Experiments on linguistically based term associations." *Information Processing & Management*, 28(3):317-332

Schutze, Hinrich (1992) " Context Space", In Robert Goldman, Peter Norvig, Eugene Charniak, and Bill Gale (eds.), *Working Notes of the AAAI Fall Symposium on Probabilistic Approaches to Natural Language*, pp. 113-120, Menlo Park, CA. AAAI Press.

Shannon, Claude E. (1948) "A Mathematical Theory of Communication", *Bell System Technical Journal* 27: pp. 39-423, 623-656.

Wilks, Yorick (1999) "Is Word Sense Disambiguation just one more NLP Task?" arXiv: CS. CL/9902030 vl.

Appendix

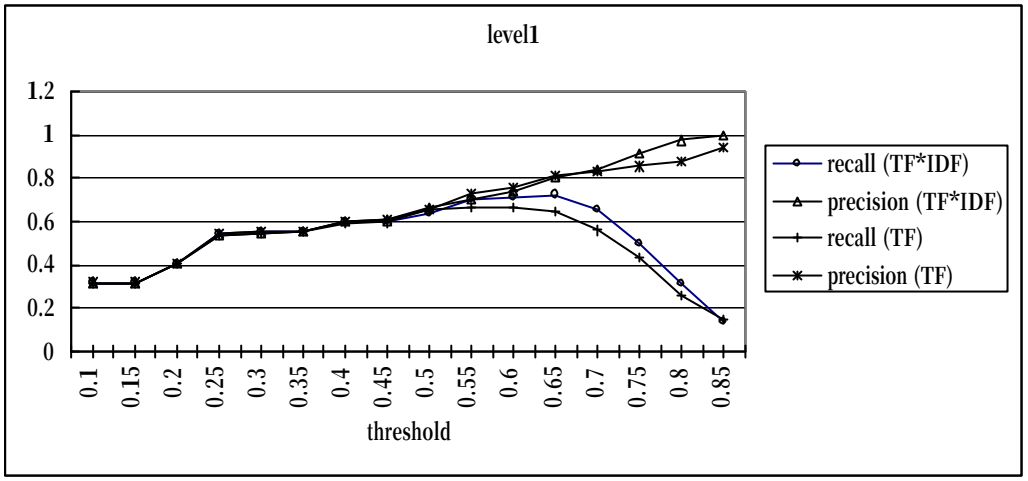

Figure 3 Clustering recall and precision at levels one of Cilin's semantic hierarchy

A partial data of Figure 3

	0.55	0.6	0.65	0.7	0.75	0.8	0.85
recall (TF*IDF)	0.706	0.709	0.725	0.657	0.498	0.313	0.138
precision (TF*IDF)	0.700738	0.736726	0.804756	0.840602	0.909853	0.973881	1
recall (TF)	0.665	0.666	0.643	0.559	0.434	0.261	0.148
precision (TF)	0.727085	0.757479	0.810962	0.829545	0.854202	0.874302	0.938776
F-Score (TF*IDF)	0.703369	0.722863	0.76488	0.748801	0.703927	0.643441	0.569
F-Score (TF)	0.696043	0.71174	0.726981	0.694273	0.644101	0.567651	0.543388

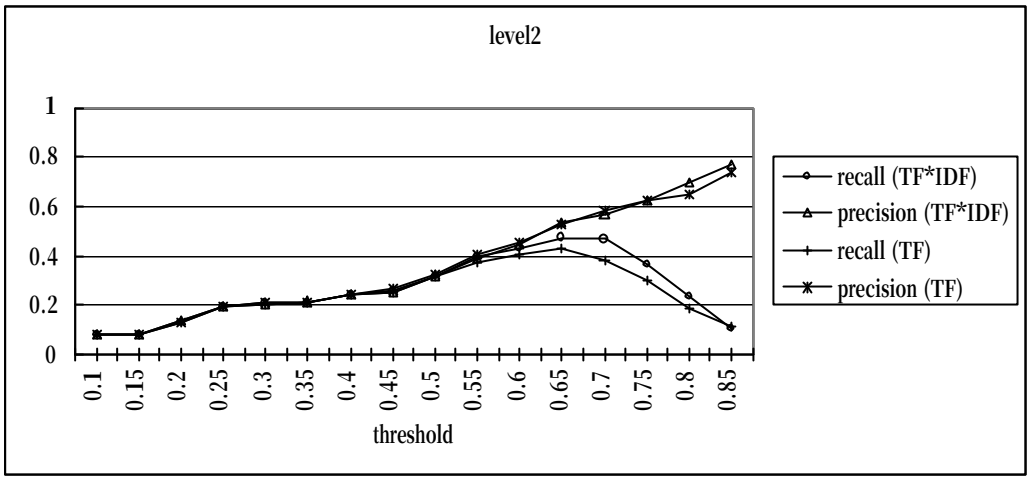

Figure 4 Clustering recall and precision at levels two of Cilin's semantic hierarchy

A partial data of Figure 4

	0.55	0.6	0.65	0.7	0.75	0.8	0.85
recall (TF*IDF)	0.394763	0.429003	0.475327	0.467271	0.367573	0.234642	0.108761
precision (TF*IDF)	0.389884	0.446903	0.53567	0.568421	0.624738	0.697761	0.77027
recall (TF)	0.372608	0.406848	0.431017	0.380665	0.300101	0.188318	0.114804
precision (TF)	0.403708	0.455128	0.527964	0.585859	0.626072	0.650838	0.734694
F-Score (TF*IDF)	0.392324	0.437953	0.505499	0.517846	0.496156	0.466202	0.439516
F-Score (TF)	0.388158	0.430988	0.479491	0.483262	0.463087	0.419578	0.424749

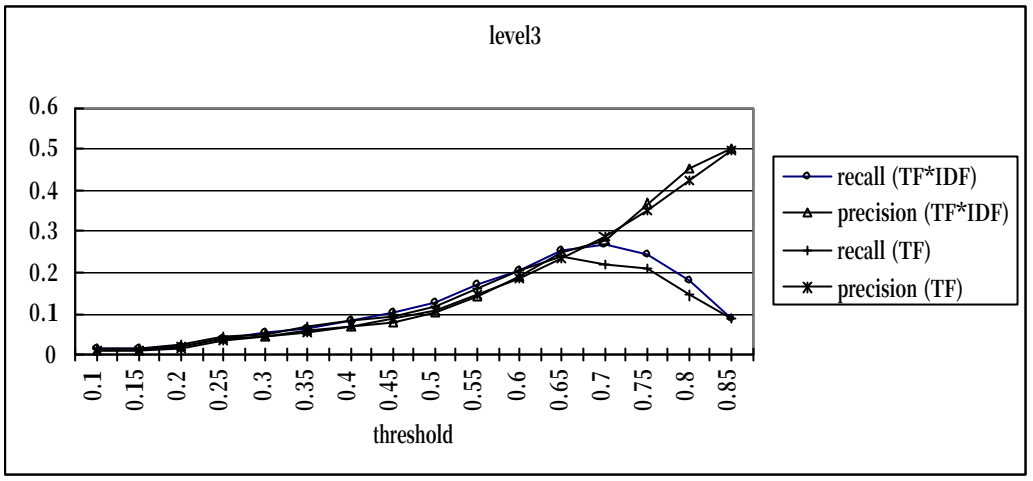

Figure 5 Clustering recall and precision at levels three of Cilin's semantic hierarchy

A partial data of Figure 5

	0.55	0.6	0.65	0.7	0.75	0.8	0.85
recall (TF*IDF)	0.169654	0.205496	0.252091	0.270012	0.243728	0.181601	0.087216
precision (TF*IDF)	0.142255	0.190265	0.249061	0.278195	0.366876	0.451493	0.5
recall (TF)	0.16129	0.205496	0.237754	0.221027	0.20908	0.144564	0.088411
precision (TF)	0.146241	0.183761	0.236018	0.285354	0.349914	0.424581	0.496599
F-Score (TF*IDF)	0.155955	0.197881	0.250576	0.274104	0.305302	0.316547	0.293608
F-Score (TF)	0.153766	0.194629	0.236886	0.253191	0.279497	0.284573	0.292505

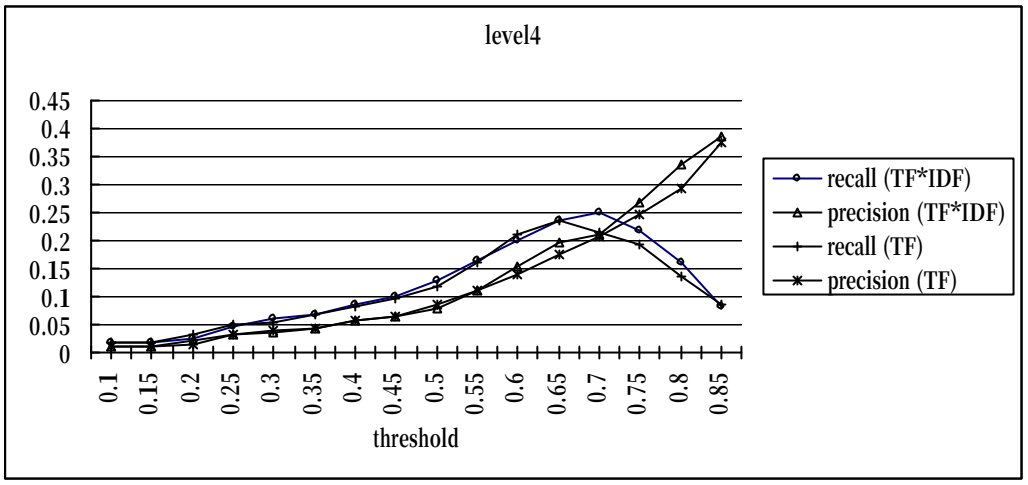

Figure 6 Clustering recall and precision at levels four of Cilin's semantic hierarchy

A partial data of Figure 6

	0.55	0.6	0.65	0.7	0.75	0.8	0.85
recall (TF*IDF)	0.164134	0.200608	0.237082	0.25076	0.218845	0.159574	0.083587
precision (TF*IDF)	0.110643	0.153761	0.195244	0.21203	0.268344	0.335821	0.385135
recall (TF)	0.159574	0.211246	0.237082	0.214286	0.194529	0.136778	0.086626
precision (TF)	0.111226	0.141026	0.174497	0.208333	0.246998	0.293296	0.37415
F-Score (TF*IDF)	0.137389	0.177185	0.216163	0.231395	0.243595	0.247698	0.234361
F-Score (TF)	0.1354	0.176136	0.20579	0.21131	0.220764	0.215037	0.230388

Table 1

GroupId	Word	average	sd
1	首先,第二,第一	0.391749	0.324147
2	狀態,大概,狀況,情形	0.400941	0.307061
3	十分,非常,特別	0.45054	0.305209
4	穩定,一定,固定	0.314694	0.262506
5	大量,太多,那麼,很多,許多	0.409695	0.256464
6	數量,多少,人數	0.379825	0.253895
7	具備,所有,具有,擁有	0.562212	0.24363
8	思想,考量,考慮,思考	0.544195	0.229534
9	大家,人們,民間	0.494836	0.214664
10	檢討,檢查,方案	0.34861	0.20496
11	類似,如同,好像,一樣,一般,接近,似乎	0.295764	0.203913
12	可以,良好,不錯,理想	0.35752	0.198352
13	明白,知道,理解,看出,發現,清楚,了解,意識,掌握	0.584519	0.195064
14	或許,也許,可能,是否	0.631396	0.186908
15	可以,肯定,同意	0.424513	0.186726
16	相同,同時,同樣,一致,一樣	0.432104	0.184799
17	以後,將來,之後,後來	0.525829	0.18295
18	體會,經驗,感受	0.513465	0.179421
19	科技,科學,統計	0.514208	0.173151
20	運動,走向,活動	0.472324	0.17241
21	不易,科學,正確	0.417904	0.17094
22	這樣子,這樣,如此,這麼	0.53811	0.170594
23	自動化,成為,變成	0.586634	0.167871
24	需要,要求,需求	0.584609	0.165922
25	在一起,共同,一起	0.559024	0.16552
26	上課,教學,教授	0.368009	0.164008
27	標準,專業,正式,規範	0.379336	0.163315
28	適合,相當,適當	0.301937	0.163089
29	以前,過去,之前	0.510083	0.161643
30	負責人,院長,主任,家長,領導,經理,主管,校長	0.54561	0.16104
31	透過,通過,經過	0.497587	0.1587
32	自然,當然,本來	0.450807	0.157253
33	基本,基礎,根本	0.303086	0.148239
34	組成,形成,構成,組織	0.558954	0.14715
35	方面,方向,走向	0.425739	0.146068
36	作為,表現,行為	0.500081	0.142414
37	城市,香港,都市	0.523753	0.142381
38	發現,發覺,感受,感覺,感到,覺得	0.587801	0.142334
39	實在,十分,真正	0.438013	0.140809

40	危險,事情,現實,事實,機關,活動,實際,行政,新聞	0.393595	0.139102
41	使用,分享,利用,採用,應用,運用	0.588138	0.132851
42	以為,認為,看看	0.684297	0.130785
43	裡面,期間,之間	0.343657	0.125277
44	博物館,媒體,中心	0.607148	0.124048
45	確定,決定,規定	0.566784	0.123687
46	維持,支持,保持	0.680776	0.123566
47	做法,辦法,方法,方式,措施,藝術,作法	0.619291	0.122881
48	根本,為主,基本,關鍵,重要,重點,主要	0.295473	0.122236
49	視為,當成,當作,當做,作為	0.747904	0.121471
50	可是,只是,但是,然而,不過	0.77375	0.121439
51	地區,地方,所在,社區,區域	0.591823	0.120499
52	消失,失去,沒有	0.632536	0.120176
53	非常,一定,特別,特殊	0.219618	0.119577
54	當中,中央,中心	0.303831	0.11493
55	成就,成功,完成	0.482776	0.112033
56	美國,中國,日本,大陸,台灣,國際,國家,我國,伊拉克,新加坡	0.614028	0.110559
57	其中,內部,裡面	0.498764	0.107996
58	告訴,報導,報告	0.425274	0.107627
59	開放,著手,開始,出發	0.677721	0.106543
60	有的,一般,部分	0.480778	0.100469
61	學會,社團,團體,協會,組織	0.660711	0.100259
62	增加,加上,豐富	0.480646	0.0995899
63	小朋友,兒童,小孩子,小孩,孩子	0.740326	0.0986638
64	一般,通常,平常	0.317419	0.0984778
65	完整,完全,整體	0.265376	0.0977068
66	工業,貿易,行業,企業,商業,交通	0.552969	0.0961678
67	計畫,計劃,設計,規劃	0.717254	0.0956848
68	規劃,設計,計劃,計畫	0.717254	0.0956848
69	系統,雙方,上面,世界	0.419015	0.0940401
70	人員,東西,個人,人口,人士,人物,份子,人類	0.473114	0.0934926
71	支援,幫忙,幫助,補助,協助,支持	0.645829	0.0932524
72	只有,只是,不過	0.698804	0.0926325
73	練習,作業,答案	0.399523	0.0902104
74	處理,安排,因應,從事	0.647893	0.0880783
75	呈現,展現,表現	0.628407	0.0859708
76	生態,動物,生物,植物	0.650257	0.0853656
77	太太,小姐,女性,女兒,女人,婦女,女孩	0.670354	0.0851898
78	前往,過去,走到	0.468615	0.0849678

79	吸引,引發,引起	0.749471	0.0834028
80	回家,回到,回來,回去	0.752641	0.0789966
81	現在,目前,今天	0.729664	0.0781769
82	圖書館,教室,房間,辦公室	0.687431	0.0758984
83	主張,主持,堅持	0.507651	0.0751646
84	委員會,機構,小組,單位,部門,組織	0.682376	0.0736292
85	系統,系列,體系	0.592294	0.0724394
86	期望,想要,願意,希望	0.750708	0.072335
87	高興,喜歡,快樂	0.71814	0.0691603
88	取得,爭取,得到,獲得	0.772821	0.0680632
89	見到,看見,看到,看看	0.789036	0.067304
90	明顯,肯定,明白	0.601009	0.0658739
91	不能,不要,不可	0.716424	0.0655924
92	高中,學院,研究所,大學,學校	0.731575	0.0652677
93	設置,舉辦,設立	0.717764	0.0621858
94	模式,程式,形式	0.669031	0.0601233
95	投入,進入,參加,加入	0.699484	0.0595996
96	作用,力量,功能,意義	0.763711	0.0577985
97	男子,先生,男人	0.632963	0.0562316
98	訓練,練習,教練	0.39034	0.0558793
99	確實,真正,實在	0.712853	0.0544218
100	學期,時期,階段,時代	0.681406	0.0539913
101	預算,計算,統計	0.377922	0.0531331
102	成果,成就,成績	0.646048	0.0524472
103	標準,規則,規範,原則	0.684752	0.050314
104	心態,觀念,概念,理念,思想,心理	0.7056	0.0490335
105	教師,教練,博士,教授,老師	0.709024	0.0489227
106	想法,意思,思想	0.69697	0.0466416
107	怎麼樣,如何,怎麼	0.482305	0.0457521
108	作業,工作,業務	0.741531	0.043475
109	年輕人,青年,青少年,少年	0.716108	0.0425544
110	體制,結構,架構,組織	0.67785	0.0397466
111	自己,本身,自我	0.57837	0.0380475
112	到底,算是,終於	0.111716	0.035438
113	環境,氣氛,條件	0.656953	0.0284353
114	缺乏,不足,緊張	0.594345	0.0275148
115	現場,市場,場所	0.470312	0.0273891
116	人家,兄弟,個人	0.405597	0.023535
117	全部,所有,一切	0.650397	0.0225594

Table2

Role index	
Role ID	Role
R1	agent
R2	apposition
R3	benefactor
R4	causer
R5	CHINESE
R6	companion
R7	comparison
R8	complement
R9	condition
R10	conjunction
R11	degree
R12	deontics
R13	DUMMY
R14	DUMMY1
R15	DUMMY2
R16	duration
R17	epistemics
R18	evaluation
R19	exclusion
R20	experiencer
R21	frequency
R22	goal
R23	Head[GP]
R24	Head[NP]
R25	Head[PP]
R26	Head[S]
R27	Head[VP]
R28	imperative
R29	instrument
R30	interjection
R31	location
R32	manner

Role index	
Role ID	Role
R33	negation
R34	particle
R35	possessor
R36	predication
R37	property
R38	quantifier
R39	quantity
R40	range
R41	reason
R42	receipient
R43	source
R44	standard
R45	target
R46	theme
R47	time
R48	topic