Research on Hypothesizing and Sorting the Eg Candidates in Chinese Semantic Parsing

XiangFeng Wei[1], Quan Zhang[1]

[1] Institute of Acoustics, Chinese Academy of Science,
Beijing, China. 100080
weixiangfeng@tsinghua.org.cn, zhq@mail.ioa.ac.cn

Abstract. This paper introduced a semantic parsing model constructed above a symbolic system of concepts for understanding natural language. Based on the model a sentence can be mapped into the corresponding semantic category. In order to obtain the semantic category of a sentence (SC), it is necessary to get the global eigen semantic chunk (Eg) in a sentence. The Eg is like the head verb in a sentence but has its own constitution. It is more difficult to find out the Eg in a Chinese sentence than in English. This paper puts forward an approach and rules to hypothesize and sort the Eg in a Chinese sentence. The approach is named 'Hypothesis-Test'. The experiments on sentences in which multi-verb appeared showed that the approach was valid and the accuracy rate of the semantic parsing system was 87.2%.

Keywords: Semantic Parsing, Hypothesis-Test, Global Eigen Semantic Chunk (Eg), Formalized Rule

1 Introduction

The global eigen semantic chunk (abbreviated as 'Eg' in the following) is the head dominant words at the top layer of a sentence. Generally the Eg roots in the head verb. However, it is possible that several verbs would appear in the sentence, like '*have also continued*', '*seek, reveal and develop*' and '*guides*' in the example sentence: '*They have also* ***continued****, in the practice of historical activities and by making comparisons, to* ***seek, reveal*** *and* ***develop*** *the truth that* ***guides*** *their advance*'. In Chinese sentences, it is more difficult to find out the Eg because of changeless form of Chinese character, flexible parts of speech (POS), lack of case, plural and tense.

In order to get the right Eg from all possible verbs in a Chinese sentence, the approach of 'Hypothesis-Test' is adopted. When parsing a sentence, the parser firstly hypothesized the Eg according to some special concepts. Then the parser hypothesized the SC of the sentence according to the Eg. Finally, the parser tests the hypothesized SC of the sentence according to the knowledge-base in the computer. If all semantic chunks in the sentence are corresponded with the transcendental concepts in the knowledge-base, the SC is confirmed. Otherwise, the SC is rejected. Therefore, the first important step of parsing the SC of a sentence is to hypothesize the Eg from the possible verbs. It is necessary to find out the right Eg in a sentence by hypothesizing and sorting the Eg candidates.

This paper discussed how to establish the rules for hypothesizing and sorting the Eg candidates in Chinese Sentences and how to implement it by rules in computer programs. The results of experiments on the Chinese sentences selected from true articles indicated that the approach of 'Hypothesis-Test' is valid. According to the formalized rules, the accuracy rate was 87.2% in which the right Eg was sorted into the first Eg candidate.

2 Rules to Sort the ‘v’ Concept

2.1 The ‘v’ Concept, the Head Verb and the Eg

In the HNC (Hierarchical Networks of Concepts) theory [1], a hierarchical symbols network of primitive concepts is established. The symbols are associated each other through conceptual relations. Any word can be explained by the primitive concepts or the combination of them. For example, the mapped conceptual symbol from the word ‘*think*’ is ‘v80’, and the mapped conceptual symbol from the word ‘*idea*’ is ‘r80’. The close relation between the two words is uncovered by the common symbol ‘80’, as that ‘*have an idea*’ is one meaning of the word ‘*think*’. So the conceptual relationship between two words can be expressed perspicuously by the symbols of the primitive concepts.

Most concepts are distinguished by two kinds of concepts: concrete and abstract. Some concepts are between them. The concrete concepts are material and touchable, such as people and the objects in the nature. The abstract concepts are invisible and untouchable. The abstract concepts are described from five properties: dynamic(v), static(g), attribute(u), value(z), result(r). Many words contain more than one aspect. For example, the word ‘look’ contains both ‘v’ property and ‘g’ property. In the sentence ‘*The teacher told us to look at the blackboard.*’, ‘look’ represents its ‘v’ property as a verb. And in the sentence ‘*I took just one look and I was sure.*’, ‘look’ represents its ‘g’ property as a noun.

The knowledge-base in the computer stores the conceptual categories, the conceptual symbols, the SC codes, the format codes, the used frequency, and the restriction in syntax of the words used frequently. If the symbols of the conceptual categories in the knowledge-base of a word include the ‘v’ property, the word is named a ‘v’ concept word, and is also called a ‘v’ concept. The term ‘the ‘v’ concept’ represents all the words whose conceptual symbols include the ‘v’ property.

The head verb is the verb or verbs at the first layer of a sentence. Usually the head verb represents the head meaning of the sentence, but in some sentences it doesn't. For example, in the sentence ‘*Cunningham took a long hard look at his lifestyle.*’, the head verb is ‘took’, but the head meaning of the sentence lies on the noun ‘look’ instead of the verb ‘took’. To grasp the head meaning of the sentence, it is necessary to take all the head words ‘*took a long hard look at*’ as a whole semantic chunk of the sentence.

The eigen semantic chunk (EK) may include the verb, the noun, and the adjective, even the preposition. There are several constitutional modes of the EK (detailing in Section 2.2). The Eg is the EK at the top layer of a sentence. It is the eigen semantic chunk in the sentence and represents the head meaning of the sentence.

In a general way, the Eg in a sentence is hypothesized from the ‘v’ concepts in the sentence. It is convenient to find out the possible Eg candidates which contain the ‘v’ concept according to the knowledge-base. Whereas, a ‘v’ concept word may includes other conceptual properties, such as ‘g’, ‘r’ and so on. This phenomenon is named the conceptual category ambiguity of a word. The conceptual category ambiguity is an obstacle when hypothesizing the Eg from the ‘v’ concept. Fortunately, there are some useful rules to eliminate the ambiguity of the conceptual category. And the approach of ‘Hypothesis-Test’ will finally get rid of most of them in the process of parsing a sentence.

2.2 Constitutions of the EK

The eigen semantic chunk (EK) is not only constituted of the verb, but also the noun and other words. The EK includes three parts: the kernel (Ek), the front modifier and the back modifier. That means the constitutions of the EK can be written as the following expression.

$$\mathrm{EK} = \mathrm{Qu} + \mathrm{Ek} + \mathrm{Hu}\ . \qquad \textbf{(1-1)}$$

For the kernel (Ek), there are five kinds of expressions to express it. The expressions are listed as the following [2].

$$\mathrm{Ek} = \mathrm{E}\,. \tag{a}$$

$$\mathrm{Ek} = \mathrm{EQ} + \mathrm{EH}\,. \tag{b}$$

$$\mathrm{Ek} = \mathrm{EQ} + \mathrm{E}\,. \tag{c}$$

$$\mathrm{Ek} = \mathrm{E} + \mathrm{EH}\,. \tag{d}$$

$$\mathrm{Ek} = \mathrm{EQ} + \mathrm{E} + \mathrm{EH}\,. \tag{e}$$

In the kind of (a), the Ek is simply composed of one ‘v’ concept as E. In the kind of (b), EQ and EH are both ‘v’ concepts, they are coordinative components of the Ek. In the kind of (c), the focus lies in E, such as ‘*seems to do*’, the meaning is determined by ‘*do*’. In the kind of (d), the focus lies in EH, such as ‘*go shopping*’, the meaning is determined by the gerund ‘*shopping*’. In the kind of (e), the meaning is determined by E.

For Qu in Formula (1-1), it is the prefix of EK. Qu maybe constituted of auxiliary verb, such as ‘*have, has, may, will*’ and so on. The Hu is the postfix of EK as a complementary illustration, such as ‘*have stayed for ten years*’. And the Qu is divided into two kinds of component as QE and qE, the Hu is divided into two kinds of component as HE and hE. Moreover, the EQ, E and EH in the kernel (Ek) may have their own modifiers EQu, Eu and EHu.

To sum up, the whole detailed expression of the EK is showed as the formula (1-2).

$$\mathrm{EK} = \mathrm{QE} + \mathrm{qE} + \mathrm{EQu} + \mathrm{EQ} + \mathrm{Eu} + \mathrm{E} + \mathrm{EHu} + \mathrm{EH} + \mathrm{hE} + \mathrm{HE}\,. \tag{1-2}$$

The constitutions of the EK not only indicate the detailed characteristic of each part of the EK, but also provide some available signs to find out the EK in a sentence. For example, the prefix and postfix of the EK are the relative reliable proofs when hypothesizing the Ek from a ‘v’ concept word.

2.3 Examples of the Rules

According to the constitutions of the EK, if a ‘v’ concept word went with prefix (QE, qE etc.) or/and postfix (hE, HE etc.), then the probability of the ‘v’ concept to be the Eg will be increased. In Chinese, the postfix ‘了’ of the Eg often appears in a sentence to indicate that the tense of the sentence is preterite. It is more credible that if ‘EQ’ in ‘EQ+E’ is a ‘vv’ concept[1], then the probability of the ‘v’ concept of ‘E’ to be the Eg will be increased. Therefore, we summarized several key rules to hypothesize and sort the ‘v’ concepts in a sentence as the following.

1. If a ‘v’ concept went with the postfix hE, then the probability of the ‘v’ concept to be Eg must be increased.
2. If a ‘vv’ concept appears in front of the ‘v’ concept, then the probability of the ‘v’ concept to be Eg must be increased.
3. If a ‘v’ concept went with the prefix QE, then the probability of the ‘v’ concept to be Eg must be increased.
4. If high layer concepts appear in front of the ‘v’ concept, then the probability of the ‘v’ concept to be Eg must be increased.
5. If the ‘uv’ concepts appear in front of the ‘v’ concept, then the probability of the ‘v’ concept to be Eg must be increased.

On the other hand, some words (concepts) in a sentence may eliminate the possibility that a ‘v’ concept to be Eg. In Chinese sentences, the typical word is ‘的’. Some rules to reject a ‘v’ concept as the Eg in a sentence are listed as the following.

A. If the word ‘的’ follows a ‘v’ concept, then the ‘v’ concept must not be the Eg in a sentence.

[1] The ‘vv’ concept is a kind of special ‘v’ concept. It cannot be the EK solely and demands the ‘v’ concept follows it to be the kernel of the EK.

B. If a ‘v’ concept follows the word ‘的’, the ‘v’ concept word isn't ‘是’, and the ‘v’ concept word isn't at the end of a sentence, then the ‘v’ concept must not be the Eg in a sentence.

C. If a ‘v’ concept follows the ‘l9’ concept, then the ‘v’ concept may not be the Eg in a sentence.

D. If a ‘v’ concept follows the ‘h$g’ concept, then the ‘v’ concept must not be the Eg in a sentence.

In rule B, the word ‘是’ is not included, because the word ‘是’ is the sign of the SC of yes-no judgment. And the first rule about the word ‘是’ in the following rules has precedence over other rules. In rule C, the ‘l9’ concept represents the concepts of denotation, such as ‘this, that’. In rule D, the ‘h$g’ concept represents some concepts which is similar to the suffix of the noun, such as ‘-ing, -er’.

I. If the word ‘是’ appears in a sentence, then the ‘v’ concepts near it must not be the Eg in a sentence, the word ‘是’ is selected to be the most possible Eg.

II. If a ‘v’ concept word possesses only the ‘v’ property of the conceptual categories in the knowledge-base, then the probability of the ‘v’ concept word to be the Eg must be increased.

Rule II is constructed on the data of the knowledge-base. Other rules can also be summarized in the same way based on the constitutions of the EK, the knowledge-base or the knowledge of syntax.

The rules can be classified into two kinds of rules. One of them is named the affirmative rule, the other is named denial rule. In order to find out the SC of a sentence ultimately, the parser gathers all ‘v’ concepts as the Eg candidates firstly. According to the denial rules, some ‘v’ concepts are rejected as Eg candidates. And according to the affirmative rules, the Eg candidates are sorted by their possibilities.

For each Eg candidate, a SC is hypothesized according to the knowledge of an Eg candidate word. Then the parser tests the SC according to the knowledge-base. If all semantic chunks are corresponded with the knowledge about the SC in database, the SC is confirmed. Otherwise, the SC is rejected. On the condition that the SC is rejected, the parser must remount the next Eg candidate to carry out the ‘Hypothesis-Test’ of the corresponding SC.

Therefore, the final targets of all rules are to hypothesize a SC from the Eg candidates. Whether the hypothesized SC is confirmed is determined by the test of the SC according to the knowledge-base.

3 Formalization of the Rules

In some parser systems, the linguistic rules are embedded into the computer programs. The advantage in this way is that the system runs efficiently. But when the rules must be modified or the new rules are summarized, it is very difficult to apply the new rules into the system, because only the programmer is able to modify the rules embedded in the program codes. If the rules are relatively independent of the program, there will be many advantages for the system. At first, all rules are managed together in a rule database. It is convenient to update old rules or add new rules. The second, linguists and programmers can dedicate themselves to their own professional work. Of course they must cooperate sometimes to correspond with the whole system. The third, it is in favor of the modularization of the system so that the work of different models can be completed by different people. The fourth, testing the rules is easier than the system of embedded rules.

Because most rules can be expressed in the form of ‘IF-THEN’, we designed a kind of symbolic formal language to describe the rules based on ‘IF-THEN’ format, Backus Naur Form (BNF), concept symbols, and program languages. If a rule can not be described in the formal language, it will be embedded into the program. The rules to hypothesize and sort the ‘v’ concepts in a sentence are described in the formal language by linguists. Further more, the rules in the formal language will be explained and executed by the computer program.

For the convenience of the executing of the computer program, the formal language describing the rules looks like program language more than natural language. The node, function, operator, expression and the character string in quotation marks are the main parts of the formal language. For example, the following rule to hypothesize the kernel of Eg (Ek) is written in the formal language.

```
_ConceptCategory((0)).INCLUDE."vv" ,  _ObjType((1),"CC").IS."v"
=> _SETOBJTYPE( (1), "CHUNK","Ek" )
```

In formal rules, a function begins with the symbol '_'. A node is a number in a pair of parentheses, describing a word or other processing units. An operator is a word between two dots. Therefore, the above formal rule means that if the symbols of the conceptual category of the current node (such as a word) include the symbol 'vv', and the next node is a 'v' concept word, then the next node would be assumed as the Ek.

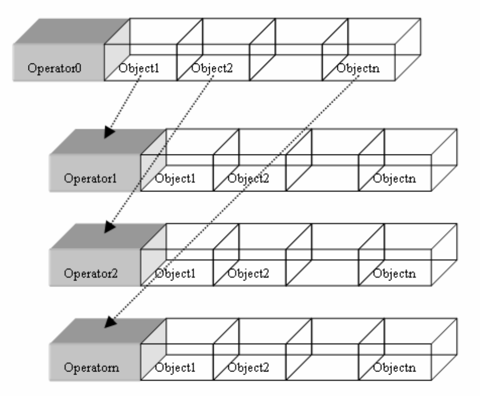

Figure 1. The Memory Structure of a Formal Rule

In the Figure 1, the memory structure of a rule is a recursive tree of multi-branch and multilayer. At the first layer of the rule, the operator is '=>'. It has two branches, one is the condition expression on the left of '=>', the other is the operation expression on the right. Each expression has its own sub-expressions or nodes. Also the sub-expressions may have their sub-expressions or nodes. For example, the above formal rule can be mapped into the following memory structure as Figure 2.

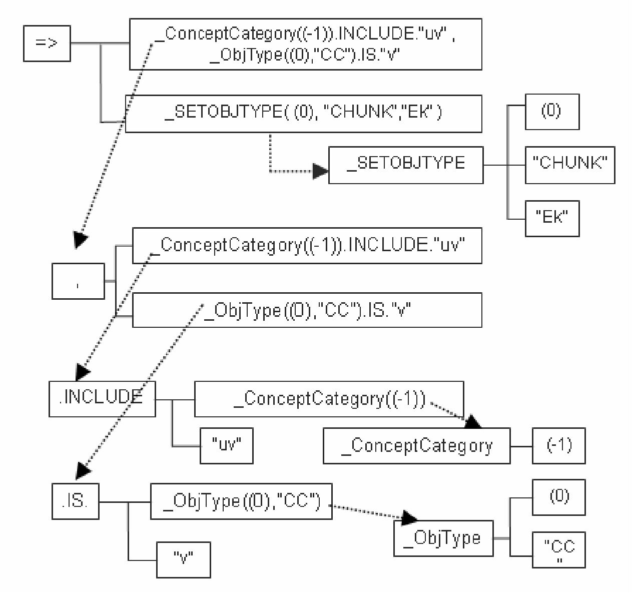
=>
_ConceptCategory((-1)).INCLUDE."uv" , _ObjType((0),"CC").IS."v"
_SETOBJTYPE((0), "CHUNK","Ek")
_SETOBJTYPE
(0)
"CHUNK"
"Ek"
,
_ConceptCategory((-1)).INCLUDE."uv"
_ObjType((0),"CC").IS."v"
.INCLUDE
_ConceptCategory((-1))
"uv"
_ConceptCategory
(-1)
.IS.
_ObjType((0),"CC")
"v"
_ObjType
(0)
"CC"

Figure 2. An Example of the Structure of a Formal Rule

As a formal rule is described in self-defined symbols and the rule can be stored in the computer like a tree structure, it is easy to explain and execute the formal rule using recursive arithmetic by the computer program.

4 Processing Multiple ‘v’ Concepts

If there is only one ‘v’ concept in a sentence, it is easy to assume the ‘v’ concept as the Eg of the sentence. However, multiple ‘v’ concepts often appear in a Chinese sentence. It will be a difficult disambiguation problem to the computer program to decide which ‘v’ concept word is the Eg of a sentence. Besides using the above rules to hypothesize Eg, it is necessary to process the relationships between ‘v’ concepts and the relationships between the possible EKs in the sentence.

If two or more ‘v’ concepts appear continuously one by one, we call them the ‘v’ block. The relationship between two ‘v’ blocks could be [3]:

(i) Ep+Er
(ii) Et+Eg
(iii) The parts of Eg
(iv) Eg+El
(v) El+Eg
(vi) E1+E2

The ‘Ep+Er’ means that the second ‘v’ concept must be the EK of a clause, such as ‘*do*’ in the sentence ‘*I think that somebody will do something*’. In Chinese sentences, the ‘Et+Eg’ means that the first ‘v’ block implies that the sentence is passive voice. ‘The parts of Eg’ means that the constitutional units are separated into two parts by other semantic chunks. The ‘El’ means that the EK from a ‘v’ block is only a local eigen semantic chunk, but not a global EK. The ‘E1+E2’ means that the sentence is a compound sentence.

The different relationships can be processed by the corresponding modules. The relationships between multiple ‘v’ blocks can be transformed into the simple relationship between two ‘v’ blocks.

Inside one ‘v’ block, the relationships between two ‘v’ concepts can also be processed as the relationships between two ‘v’ blocks. According to the denial rules, the constitutions of the EK, and the relationships between two ‘v’ blocks, the following steps of the experiment on the two consecutive ‘v’ concepts are established.

(1) Using the denial rules to find out the ‘v’ concept which can not be Eg.
(2) According to the knowledge-base, judge whether the first ‘v’ concept can be the ‘Ep’ or ‘Et’.
(3) According to the knowledge of the constitutions of the EK, judge whether the two ‘v’ concepts make up of the EK.
(4) Test the two ‘v’ concepts to judge whether one of them can be El.
(5) Test the two ‘v’ concepts to judge whether their relationship is the ‘E1+E2’.
(6) Select the first ‘v’ concepts as the Eg of the sentence.

If a ‘v’ concept has been denied to be Eg at some step, it can not be assumed as the ‘Eg’ at the latter steps. All the hypothetic EK must be tested according to the knowledge of the SC. If the test failed, the hypothesis of the EK from the ‘v’ concept wouldn’t be confirmed.

5 Experiments

We experimented on 187 sentences in which multiple ‘v’ concepts appeared. The sentences were selected directly from three raw articles. After parsing the sentences the computer program listed the Eg candidates by the above rules and steps. The experimental results indicated that there were 24 sentences in which the parser didn't hypothesize the first Eg candidate as the right Eg from multiple ‘v’ concepts. So the error rate of the system of hypothesizing the first Eg candidate as the right Eg from multiple ‘v’ concepts is 12.8%, and the accuracy rate of the system is 87.2%. More detailed data associated with the formal rules are listed in Table 1.

In Table 1, the most frequent rule which is used by the system is the rule ‘Denying the last ‘v’’. The accuracy rate of the rule ‘Denying the last ‘v’’ is 70.5%, whereas the accuracy rate of some other rules is 100%. Therefore, the rule ‘Denying the last ‘v’’ defined in the step (6) must be improved to obtain higher accuracy rate.

Table 1. The Data of Rules for Processing Multiple ‘v’ Concepts.

Rule Name	The Count of accurate Sentences	The Count of inaccurate Sentences	The Total of Sentences	Accuracy Rate	Error Rate
Denying ‘v’	3	0	187	100%	0
‘的’Denying	12	2		85.7%	14.3%
Ep+Er	15	0		100%	0
EQ+E	17	4		81%	19%
Confirming the Prefix and Postfix	39	0		100%	0
EQ+EH	4	0		100%	0
Denying the Last ‘v’	43	18		70.5%	29.5%
Confirming the Eg	30	0		100%	0
Total	163	24		none	

6 Conclusion

The aim of understanding a sentence is to obtain the SC and semantic chunks in it. The work of hypothesizing and sorting the Eg candidates in a sentence is the most fundamental and important step. This paper illuminated the rules for hypothesizing and sorting the Eg candidates in a Chinese sentence and the implementation of the rules in computer. The experimental results showed that the accuracy rate of the right Eg was up to 87.2%. The results of analyzing the SC and semantic chunks in a sentence can be applied into an intelligent web search system. Through the association of the conceptual symbols of the words in a sentence, more associated information are matched and listed. Through analyzing the eigen semantic chunk (EK), the general semantic chunk (GBK) and their semantic relationships in a sentence, the intelligent search system will be able to process the question what the user asked in natural language and obtain more reasonable accurate answers.

Acknowledgments. This work is 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. Zengyang Huang. The Theory of Hierarchical Network of Concepts. Tsinghua University Press, Beijing, China (1998)
2. Chuanjiang Miao. Studies on the Knowledge of Sentence Category in HNC Theory. Ph. D. Dissertation of the Institute of Acoustics, Chinese Academy of Sciences, Beijing, China (2001)
3. Yaohong Jin. The Research and Implementation for Dealing with the Difficulties Arose by More Verbs in Chinese Understanding. Ph. D. Dissertation of the Institute of Acoustics, Chinese Academy of Sciences, Beijing, China (2003)