Journal of Theoretical and Applied Information Technology
28th February 2013. Vol. 48 No.3

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

K-PRUNING ALGORITHM FOR SEMANTIC RELEVANCY CALCULATING MODEL OF NATURAL LANGUAGE

YUNTONG LIU
School of Computer and Information Engineering, Anyang Normal University, Anyang 455000, Henan China

ABSTRACT

In order to process natural language more effectively, a semantic relevancy calculating model of natural language was proposed, and the *k*-pruning algorithm for solving the model was researched. In the model, the best parsing process for a sentence could be determined by the value of semantic relevancy of the sentence; the two-level semantic structure of a sentence were analyzed, and two grammar rules were used to describe the two-level semantic structure; In the process of solving the model, a state tree would be generated; the *k*-pruning algorithm could be used to delete the states with less semantic relevancy when searching the state tree, and the computational complexity could be effectively reduced and the approximate solution could be acquired. Finally experiments were finished to verify the effectiveness of the algorithm.

Keywords: *Natural Language, Semantic Relevancy, The State Tree, K-Pruning Algorithm*

1. INTRODUCTION

Due to there is no effective technical method for natural language processing, more and more researchers were researching to achieve better results by the semantic information in the sentence.

Therefore, semantics were more and more widely utilized in natural language processing. Many lexical semantic knowledge bases such as Wordnet, Framenet had been constructed as the basic resources in semantic analyzing process[1]; semantic role labeling system based on dependency tree distance was researched[2]; the head-driven phrase structure grammar had parsed sentences by the semantics of the words in the sentences[3]; the preposition disambiguation had been exploited through the semantic role resources[4]; the online semantic resources could be automatically reused through word sense disambiguation[5];the effect of word sense disambiguation could be improved according to the WordNet [6].Although the research had made considerable achievements, but no effective model by using semantics for natural language processing had not been proposed.

In order to using semantics more effectively in natural language processing, a semantic relevancy calculating model of natural language was proposed, and the *k*-pruning algorithm for solving the model was researched. The two-level semantic structures of a sentence were analyzed and the mathematical forms were proposed; the bottom-up resolution method had been used to solve the model; in the model, the best parsing process for a sentence could be determined by the value of semantic relevancy of the sentence; in the process of solving the model, a state tree would be generated, and the *k*-pruning algorithm had be used to delete the states with less semantic relevancy during searching the state tree, and the approximate solution could be acquired.

2. THE SEMANTIC RELEVANCY CALCULATING MODEL FOR A SENTENCE

Suppose each word W_i (Except for the predicate words) in a sentence (C_S) semantically modify another word W_{Gi}, the semantic relevancy between W_i and W_{Gi} could be represented by the correlation function $SR(W_i, W_{Gi})$.

Suppose there are m kinds of parsing process for the sentence C_S; in the *i* parsing process P_i: V are the predicate words, S are the subject words, and O are the object words. The semantic relevancy of the sentence for P_i can be expressed by the formula (1), as shown in Figure 1:

$$f_{Pi} = \mathbf{w} * (\boldsymbol{SR}(\mathbf{S},\mathbf{V}) + \boldsymbol{SR}(\mathbf{O},\mathbf{V})) + \sum_{i=1}^{\mathbf{n}} * \boldsymbol{SR}(\mathbf{W}_i, \mathbf{W_{Gi}}) \quad (1)$$

In formula 1, n is the number of words in C_S (not including S, V, O), w is the weight coefficient, generally w should be proportional to the length of the sentence and w> 1.

Journal of Theoretical and Applied Information Technology
28th February 2013. Vol. 48 No.3

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

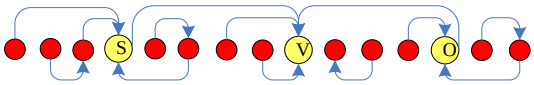

Figure 1. The Schematic Diagram For Formula 1

In the calculating process, the grammatically-partial word should be neglected.

The rule for selecting the best result: the most reasonable parsing process would meet the conditions in the formula (2):

$$P_i = argmax(f_{Pi}) \quad (2)$$

The semantic relevancy of the best parsing process would be the max in all the parsing process.

3. THE TWO-LEVEL SEMANTIC STRUCTURE AND ITS DESCRIPTION

According to the semantic structure, all the sentences could be divided into two kinds: the simple sentences and the complex sentences.

I. The simple sentence: the sentence (C_S) without subordinate sentence. It could be described by grammar G_1 (Figure 2, Table 1), the grammar G_1 was designed according to the case grammar [7][8].

Table 1
Rules For Grammar G_1

items	Content			
Notation	Noun(n)	Adjective(v_a)	Adverb(v_{ad})	verb(v)
	Pre-word (w_b)	Mid-word (w_m)	Post-word (w_e)	Other (n)
Sentence Rules $C_S\rightarrow$	LSLVLOL LVLSLOL LOLSLVL	LSLOLVL LVLOLSL LOLSLVL	LS LVL LVLSL LOLVL	LVLOL
Rules of L	$L\rightarrow\varepsilon\|A_A\|A_B\|P_D\|P_DA_B\|A_AP_D\|A_AP_DA_B$ A_B (A_A) $\rightarrow\varepsilon\|n\|v_an\|$ $A_Bn\|w_bA_B\|A_Bw_mA_B\|w_bA_Bw_eA_B$ $P_D\rightarrow\varepsilon\|v_{ad}\|P_Dv_{ad}\|w_bP_D\|P_Dw_mP_D\|w_bP_Dw_eP_D$			
Rules of SVO	$S\rightarrow n\|SLS$ $V\rightarrow v\|v_a\|V\|Vw_mV$ $O\rightarrow n\|OLO$			

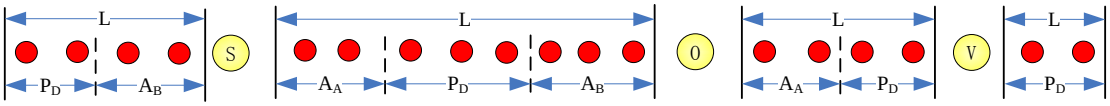

Figure 2. The Schematic Diagram For Grammar G_1

II. The complex sentences: the sentence (C_S) with subordinate sentence, the grammar G_2 could be generated by adding rule $L\rightarrow C_S$ to the grammar G_1. In G_2, a simple sentence might be any contents in a sentence, so G_2 could describe the complex sentences.

4. THE ANALYSING AND CALCULATING PROCESS FOR A SENTENCE

4.1 The bottom-up resolution algorithm for the simple sentence

In the calculating process, a simple sentence (the subordinate sentence) would be selected and resolute to L, and the resolution process might be repeated until the sentence had become a simple sentence:

Step1: (finding all the subordinate sentence): for the sentence C_S, executing the CYK algorithm[9] according to the grammar G_1, a collection of substrings could be gotten which were satisfied to the grammar G_1, suppose they were {C_{S1}, C_{S2} ..... C_{Sm}}, as shown in Figure 3;

Step2: Calculating the semantic relevancy for each subordinate sentence;

Step3: Selecting the subordinate sentence C_{Si} with the best semantic relevancy, and resoluting C_{Si} to L;

Step4: If C_S was not a simple sentence, repeating step2 and step3, or calculating the semantic relevancy for C_S.

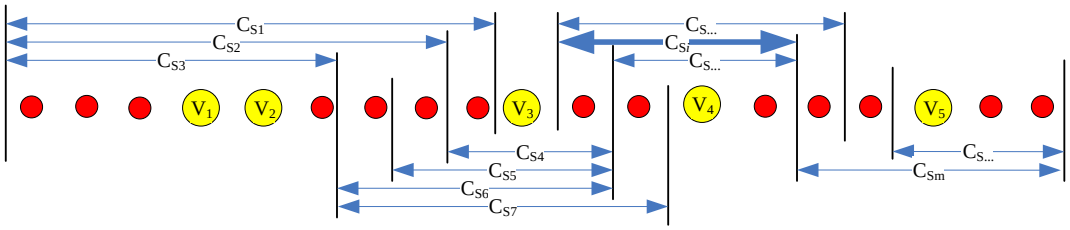

Figure 3. The Subordinate Sentences In The Resolution Process

Journal of Theoretical and Applied Information Technology
28th February 2013. Vol. 48 No.3

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

4.2 The Best Semantic Relevancy For A Simple Sentence

4.2.1 The semantic relevancy between two words

Suppose there are two words W_i and W_{Gi}, and there is a lexical semantics library organized as a tree, such as Wordnet. The semantic relevancy between W_i and W_{Gi}, could be calculated by formula 3:

$$SR(W_i, W_{Gi}) = \alpha * \text{sim}(W_i, W_{Gi}) + \beta * \text{rel}(W_i, W_{Gi}) \quad (3)$$

$\text{sim}(W_i, W_{Gi})$ is the semantic similarity between them, $\text{rel}(W_i, W_{Gi})$ is the semantic association between them, and the coefficien $\alpha+\beta=1$, the specific details had been discussed in references [10].

4.2.2 The multiple choices for SVO

In the calculating process, the location of the SVO should be determined as a prerequisite. For a simple sentence, the position of V was determined, and the position of the S and O would be multiple choices, as shown in Figure 5:

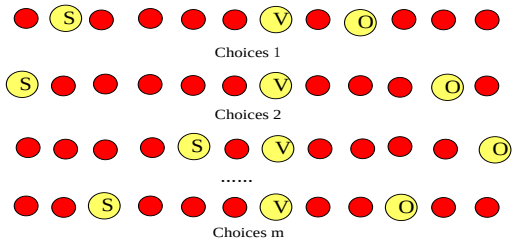

Figure 5. The Multiple Choices For SVO

4.2.2 The multiple choices for parsing L

After the position of SVO were determined, we should the determined the semantically modified word W_{Gi} for each word W_i, and the semantic similarity $SR(W_i, W_{Gi})$ between them should be calculated. But for each segment L, there would be multiple choices for parsing L, as shown in Figure 5:

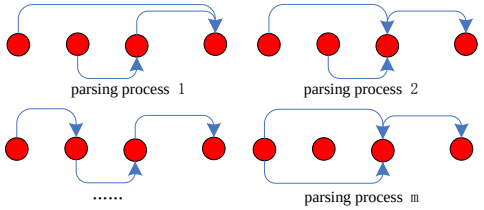

Figure 4. The Multiple Choices For Parsing L

4.3 The *k*-pruning algorithm

The *k*-pruning algorithm is the variation of the greedy algorithm. When there were multiple choices in the greedy algorithm, only one best choice would be selected and treated for the next step. However, in the *k*-pruning algorithm, the *k* best choices would be selected and treated for the next step.

4.3.1 The principle of *k*-pruning algorithm

It is can be seen from the analysis of 4.1 and 4.2:

I. There would be multiple choices in the bottom-up resolution process for the simple sentence;

II. There would be multiple choices when determining the SVO of a simple sentence;

III. There would be multiple choices when parsing the segments L.

In the process of solving the model, a state tree would be generated. The *k*-pruning algorithm could delete the states with less semantic relevancy by the pruning function when searching the state tree.

Theoretically, we could find the most reasonable parsing process by exhaustively searching for the state tree; however the computational complexity of the method would be too high for the computers.

In order to reduce the computational complexity and solve the problem, we used the ***k***-pruning algorithm in searching for the state tree; when there were multiple states, only the *k* highest possible states would be searched and the other states would be deleted. By the ***k***-pruning algorithm, an approximate result would be obtained and the computational complexity would be significantly reduced. As shown in Figure 6:

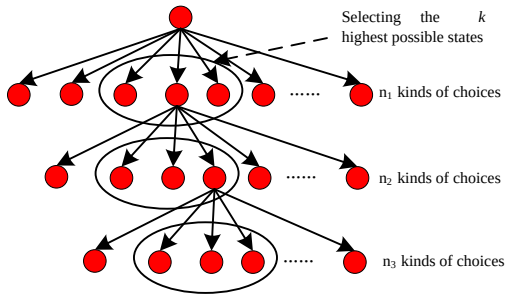

Figure 6. The K-Pruning Algorithm(K=3)

4.3.2 The pruning function

During the searching process, a pruning function was needed to select and delete the states of the state tree. We choose the average semantic relevancy as the pruning function, as shown by formula 4:

$$f_C(\mathbf{L}) = f_{Ai}(\mathbf{L}) / n \quad (4)$$

The segments L (a sentence) might be multiple parsing processes, suppose they are $\{A_1, A_2, ..., A_m\}$, n is the word number of L, f_{Ai} (L) is the semantic relevancy of L for parsing processes A_i. By the formula 1, we can get the semantic

Journal of Theoretical and Applied Information Technology
28th February 2013. Vol. 48 No.3

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

relevancy results, suppose they were $\{f_{A1}(L), f_{A2}(L)\ldots(L)\ f_{Am}(L)\ \}$. According to the results the best *k* the corresponding state could be selected and the other states were deleted.

5. EXPERIMENTAL RESULTS AND ANALYSIS

When we calculated the semantic relevancy between two words the Wordnet had been used as the lexical semantics library. The experiments were composed of two stages:

I. The experiments to determine the value of the weight coefficient w in formula 1;

II. The experiments to determine the value of the *k* in the *k*-pruning algorithm.

5.1 The Experiments To Determine W

Because w should be proportional to the length of the sentence, in the experiments we set w=0.9n, 0.8n, 0.7n, 0.6n, 0.5n, 0.4n, 0.3n (n is the number of words in the statement); and we selected 100 simple sentences for the experiments; for each sentence, the exhaustively searching method was used to get the best parsing processes, the experimental results were shown in Table 2, and the relations between w and the correct rates were shown in Figure 7:

Table 2: The Relations Between W And The Correct Rates

w=	0.9n	0.8n	0.7n	0.6n	0.5n	0.4n	0.3n
correct rates	57	64	73	78	81	74	68

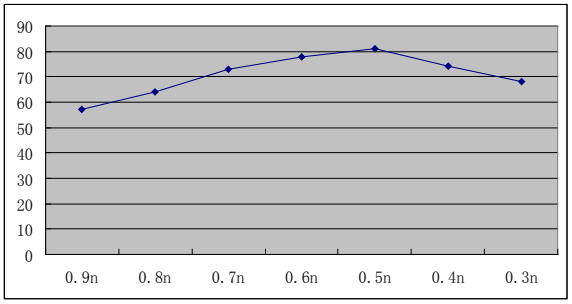

Figure 7. The relations between w and the correct rates

The experiment results showed that if w is between 0.5 and 0.6 the correct rate would be maximum.

5.2 The Experiments To Determine *K*

The value of *k* should be determined through experiments, *k* were respectively set 2,3,4,5,6 in the experiments. 100 complex sentences were selected for the experiments; for each sentence, the *k*-pruning algorithm was used to get the best parsing processes(set w=0.55), the experimental results were as follows:

5.2.1 The relations between *k* and the average time

The average times for *k* were shown in Table 3 and Figure 8 (windows xp; CPU: Xeon E5-2403,2GHz; Mem:8G).

Table 3: The Relations Between K And The Average Time

k=	2	3	4	5	6	2
average time	2.57	7.81	30.72	156.36	932.42	2.57

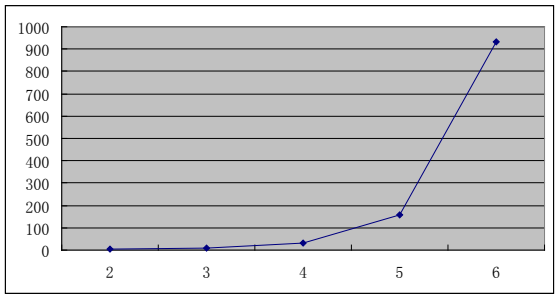

Figure 8. The Relations Between K And The Average Time

The experiment results showed that the average time T were proportional to the factorial of k: $T \approx \gamma * k!$. If $k \geq 6$, the average time would be too high to be accepted.

5.2.1 the relations between *k* and the correct rates

The relations between *k* and the correct rates were shown in Table 4 and Figure 9.

Table 4
The relations between *k* and the correct rates

k=	2	3	4	5	6	2
correct rates	34	51	68	74	77	34

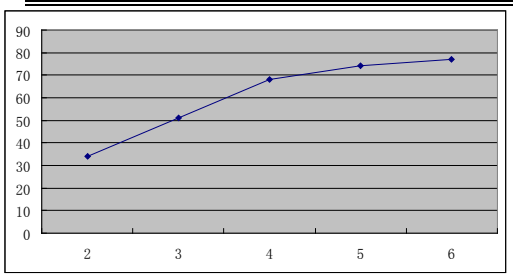

Figure 9. The Relations Between K And The Correct Rates

The experiment results showed that the correct rates increased rapidly from *k*=1 to *k*=4 and the correct rates increased slowly the correct rates might achieved a satisfactory degree when $k > 4$.

6. SUMMARIES

The reasonable parsing process for a sentence would be acquired by the *k*-pruning algorithm

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

through calculating the semantic relevancy between words in the sentence. During searching the state tree, many states with less semantic relevancy might be deleted by the pruning function, and only *k* states were selected to be calculated in the next step; the computational complexity would be significantly reduced. Finally an experiment was finished to determine the best value of *k*.

ACKNOWLEDGEMENTS

This work was supported by the National Natural Science Foundation of China (Grant No. U1204402)

REFERENCES:

[1] W. Xue, X. Xinshun, R. Zhaochun, "WordNet-based Lexical Unit Induction for FrameNet", *Journal of Computational Information Systems* Vol.8,No.3,2012,pp.1047-1054.

[2] W. Xin, S.Zhifang, "Semantic Role Labeling System Based on Dependency Tree Distance Method for Arguments Identification", *Journal of Chinese Information Processing*, Vol.26, No.2, 2012, pp.40-45.

[3] T.Ninomiya, T.Matsuzaki, Y.Miyao, "HPSG Parsing with a Supertagger", *Text, Speech and Language Technology*, Vol.43, No.6, 2011, pp.243-256.

[4] O.Tom, W.Janyce, "Exploiting Semantic Role Resources for Preposition Disambiguation", *Computational Linguistics*, Vol.35, No.2, 2008, pp.151-184.

[5] N.Imdadi, A.Syed, M.Rizvi, "Automating Reuse of Online Semantic Resources by Concept Extraction Using Word Sense Disambiguation", *Journal of Algorithms & Computational Technology*, Vol.6, No.3, 2012, pp.435-446.

[6] K.Deepesh, J.Choudhury, A.Chakrabarty, "Improvement in Word Sense Disambiguation by introducing enhancements in English WordNet Structure",*International Journal on Computer Science and Engineering*, Vol. 4, No. 7, 2012, pp.1366-1370.

[7] C. Kehjiann, H. ChuRen, "Information-based Case Grammar", COLING '90 Proceedings of the 13th conference on Computational linguistics, Vol. 2, 1990, pp. 54-59.

[8] Woolford, Ellen, "Lexical case, inherent case, and argument structure", *Linguistic Inquiry*, Vol.37, No.1, 2006, pp.111-130.

[9] L.Yongliang, H.Shuguang, L.Yongcheng, "Improved CYK algorithm based on shallow parsing", *Journal of Computer Applications*, Vol.31, No.5, 2011, pp.1335-1338.

[10] G.Suryanarayanan, S.Selvaraju, A.Irulappan, "Ontology-based relevance analysis for automatic reference tracking", *International Journal of Computer Applications in Technology*, Vol.35, No.2, 2009, pp.165-173.