

Zhou qiaoli: A divide-and-conquer strategy for semantic dependency parsing

Qiaoli Zhou **Ling Zhang** **Fei Liu** **Dongfeng Cai** **Guiping Zhang**

Knowledge Engineering
Research Center Shenyang Aerospace University
No.37 Daoyi South Avenue
Shenyang, Liaoning, China

Zhou_qiao_li@hotmail.com 710138892@qq.com fei_l2011@163.com caidf@vip.163.com zgp@ge-soft.com

Abstract

We describe our SemEval2012 shared Task 5 system in this paper. The system includes three cascaded components: the tagging semantic role phrase, the identification of semantic role phrase, phrase and frame semantic dependency parsing. In this paper, semantic role phrase is tagged automatically based on rules, and takes Conditional Random Fields (CRFs) as the statistical identification model of semantic role phrase. A projective graph-based parser is used as our semantic dependency parser. Finally, we gain Labeled Attachment Score (LAS) of 61.84%, which ranked the first position. At present, we gain the LAS of 62.08%, which is 0.24% higher than that ranked the first position in the task 5.

1 System Architecture

To solve the problem of low accuracy of long distance dependency parsing, this paper proposes a divide-and-conquer strategy for semantic dependency parsing. Firstly, Semantic Role (SR) phrase in a sentence are identified; next, SR phrase can be replaced by their head or SR of head. Therefore, the original sentence is divided into two kinds of parts, which can be parsed separately. The first kind is SR phrase parsing; the second kind is

parsing the sentence in which the SR phrases are replaced by their head or SR of head. Finally, the paper takes graph-based parser as the semantic dependency parser for all parts. They are described in Section 2 and Section 4. Their experimental results are shown in Section 5. Section 6 gives our conclusion and future work.

2 SR Phrase Tagging and Frame

To identify SR phrase, SR phrase of train corpus are tagged. SR phrase is tagged automatically based on rules in this paper. A phrase of the sentence is called Semantic Role phrase (SR phrase) when the parent of only one word of this phrase is out of this phrase. The word with the parent out of the phrase is called Head of Phrase (HP). The shortest SR phrase is one word, while the longest SR phrase is a part of the sentence. In this paper, the new sequence in which phrases are replaced by their head or SR of head is defined as the frame. In this paper, firstly, SR phrases of the sentence are identified; secondly, the whole sentence is divided into SR phrases and frame; thirdly, SR phrase and frame semantic dependency are parsed; finally, the dependency parsing results of all components are combined into the dependency parsing result of the whole sentence.

SR of HP is used as the type of this phrase. Only parts of types of SR phrases are tagged. In this paper, the tagged SR phrases are divided into two

types: Main Semantic Role (MSR) phrase and Preposition Semantic Role (PSR) phrase.

2.1 MSR Phrase Tagging

In this paper, MSR phrase includes: OfPart, agent, basis, concerning, content, contrast, cost, existent, experiencer, isa, partner, patient, possession, possessor, relevant, scope and whole. MSR phrase tagging rules are shown in figure1&2.

```

Input: wi: word index (ID) in a given sentence.
      N: the number of words.
      Mi: MSR list.
      Vi: POS tags list
Output: the last word ID of MSR phrase
Function: Findmainsemanticword(wi): return word
         ID when wi of semantic belongs to Mi.
         Otherwise return 0.
Function: FindPOSword(wi): return true when wi
         of POS tagging not belongs to Vi. Oth-
         erwise return 0.
Function Findlastword(wi)
  For i←1 to N do begin
    If (Findmainsemanticword(wi)&&
        FindPOSword(wi))
      {
        return wi;
      }
    else {
      i++;
    }
  end
  return 0;

```

Figure1: Tagging Rule of the Last Word of MSR Phrase

Figure 1 shows the rule for identification of the last word of MSR phrase. If the SR of the current word is MSR and its POS is not VV, VE, VC or VA, it is the last word of phrase.

As shown in the figure 2, the first word of phrase is found based on the last word of phrase. The child with the longest distance from the last word of phrase is used as the current word, and if the current word has no child, it is the first word of phrase; otherwise, the child of the current word is found recursively. If the first word of phrase POS is preposition and punctuation, and its parent is the last word, the word following the first word serves as the first word of phrase.

```

Input: Lword: the last word ID of MSR phrase.
Output: Fword: the first word ID of MSR phrase.
Function: Findmaxlenchild (w): return child ID
         with the longest distance from w when w
         has child. Otherwise returns 0.
Fuction: FindPOSword(w): return POS of w.
Fuction: Findparent(w): return parent ID of w.
Function Findfirstword(Lword)
  If(Findmaxlenchild (Lword)= =0)
  {
    return Lword;
  }
  Else {
    Fword=Findmaxlenchildword(Lword);
    If(findPOSword(Fword)= =P||
        findPOSword(Fword)= =PU)
    {
      If (findparent(Fword)= =Lword)
        Return Fword +1;
    }
    Findfirstword(Fword);
  }

```

Figure2: Tagging Rule of the First Word of MSR Phrase

```

29 而 而 CC CC _ 30 aux-depend _ _
30 是 是 VC VC _ 58 s-succession _ _
31 借鉴 借鉴 VV VV _ 54 s-succession_ _
32 发达 发达 JJ JJ _ 33 d-attribute _ _
33 国家 国家 NN NN _ 37 s-coordinate _ _
34 和 和 CC CC _ 37 aux-depend _ _
35 深圳 深圳 NR NR _ 37 d-member _ _
36 等 等 ETC ETC _ 35 aux-depend _ _
37 特区 特区 NN NN _ 40 d-genetive _ _
38 的 的 DEG DEG _ 37 aux-depend _ _
39 经验 经验 NN NN _ 40 s-coordinate _ _
40 教训 教训 NN NN _ 31 content _ _

```

Figure3: Example of the Tagging MSR Phrase

As shown in the figure 3, the first column is word ID and the seventh column is parent ID of word. SR of ID40 is content, so ID40 is the last word of phrase. Its children include ID39 and ID37, thus ID37 with the longest distance from ID40 is the current word. The child of ID37 is ID33, the child of ID33 is ID32, ID32 has no child, and ID32 is the first word of SR phrase.

The tagged result in the above figure 3 is as follows: 而/CC 是/VC 借鉴/VV content[发达/JJ 国家/NN 和/CC 深圳/NR 等/ETC 特区/NN 的/DEG 经验/NN 教训/NN]

After phrases are tagged, a new sequence generated by replacing the phrase with HP is called MSR frame.

MSR frame: 而/CC 是/VC 借鉴/VV 教训/NN

Example of sentences with nested phrases:

据/P 初步/JJ 统计/NN , /PU 目前/NT exist-ent[在/P 中国/NR 境内/NN 承包/VV content[工程/NN] 的/DEC 国外/NN 承包商/NN] 已/AD 有/VE 一百三十七/CD 家/M

After phrases are tagged, a new sequence generated by replacing the phrase with HP is called MSR frame.

MSR frame: 据/P 初步/JJ 统计/NN , /PU 目前/NT 承包商/NN 已/AD 有/VE 一百三十七/CD 家/M

2.2 PSR Phrase Tagging

In this paper, SR phrase containing preposition is defined as PSR phrase. If the POS tags of the current word is Preposition (P), the first word and the last word of PSR phrase are found based on the current word. PSR phrase tagging rule as figure 4 & 5.

```

Input: Pword: the word ID that word POS tags is P.
Output: Fword: the first word ID of PSR phrase.
Function: Findmaxlenchildword(w): return word ID
with the longest distance from w when w
has child. Otherwise returns 0.
Function Findfirstword(Pword)
  If(Findmaxlenchildword(Pword)= =0)
  {
    return Pword;
  }
  Else {
    return Fword=
    Findmaxlenchildword(Pword);
  }

```

Figure 4: Tagging Rule of the First Word of PSR Phrase

As shown in the figure 4, the child with the longest distance from the current word is the first word of phrase. If the prep has no child, then it is PSR phrase.

As shown in the figure 5, firstly, the parent of the prep is found; next, the parent is taken as the current word, and the child with the longest distance from the current word is found recursively. If no child is found, the current word is the last word of PSR phrase. If preposition of SR is root or parent of preposition is root, and preposition is PSR.

If ID of preposition is larger than ID of parent of preposition, and preposition is PSR.

```

Input: Pword: the word ID that word POS tags is P.
Output: Lword: the last word ID of PSR phrase.
Function: Findmaxchild (w): return word ID that
length is max with w when w has child.
Otherwise return 0.
Function: Findparent (w): return word ID when w of
parent is not root. Otherwise return 0.
Function: Findroot(w): return 1 when w of semantic
role is root. Other wise return 0.
Function Findlastword(Pword)
Var cword: parent ID
  If(Findparentword(Pword)= =0||
    findroot(Pword)= =1) {
    return Pword;
  }
  else { cword=Findparent (Pword) )
    If(Pword>cword){
      return Pword;
    }
    else {
      if(Findmaxchild (cword)= =0) {
        return cword;
      }
      else{
        Lword=
        Findmaxchild (cword);
        Findlastword(Lword);
      }
    }
  }
}

```

Figure5: Tagging Rule of the Last Word of PSR Phrase

```

1 外商 外商 NN NN _ 2 j-agent _ _
2 投资 投资 NN NN _ 3 r-patient _ _
3 企业 企业 NN NN _ 11 agent _ _
4 在 在 PP _ 5 prep-depend _ first word
5 改善 改善 VV VV _ 11 duration _ head_
6 中国 中国 NR NR _ 8 d-genitive _ _
7 出口 出口 NN NN _ 8 r-patient _ _
8 商品 商品 NN NN _ 9 d-host _ _
9 结构 结构 NN NN _ 5 patient _ _
10 中 中 LC LC _ 5 aux-depend _ last word_
11 发挥 发挥 VV VV _ 0 ROOT _ _
12 了 了 AS AS _ 11 aspect _ _
13 显著 显著 JJ JJ _ 14 d-attribute _ _
14 作用 作用 NN NN _ 11 content _ _
15 。 。 PU PU _ 11 PU _ _

```

Figure6: Example of the Tagging PSR Phrase

As shown in the figure6, ID4 is prep, and it has no child, so the first word is ID4. The parent of

ID4 is ID5, the child with the longest distance from ID5 is ID10, and ID10 with no child is the last word of phrase.

The tagged result in the above figure 6 is as follows: 外商/NN 投资/NN 企业/NN duration[在/P 改善/VV 中国/NR 出口/NN 商品/NN 中/LC] 发挥/VV 了/AS 显著/JJ 作用/NN 。/PU

The position of HP in PSR phrase is not fixed. After phrases are tagged, a new sequence generated by replacing the phrase with SR of HP is called PSR frame.

PSR frame: 外商/NN 投资/NN 企业/NN duration/duration 发挥/VV 了/AS 显著/JJ 作用/NN 。/PU

Examples of sentences with nested phrases:

s-cause[由于/P 裕隆/NR s-purpose[为/P 因应/VV Y 2 K/NT] 而/MSP 决定/VV 更新/VV 整/DT 个/M 电脑/NN 架构/NN],/PU 因此/AD 资讯/NN 部门/NN 可/VV 谓/VV 人仰马翻/VV 。/PU

PSR frame: s-cause/s-cause ,/PU 因此/AD 资讯/NN 部门/NN 可/VV 谓/VV 人仰马翻/VV 。/PU

2.3 SR Phrase Tagging Performance

If the parent of only one word of the tagged phrase is out of this phrase, this phrase is tagged correctly. If each word in the generated frame has one parent (i.e. words out of the phrase are dependent on HP instead of other words of the phrase), the frame is correct.

	Phrase	Frame
MSR	99.99%	100%
PSR	99.98%	99.70%

Table 1. Tagging Performance (P-score)

As shown in the table 1, tagging results were of very high accuracy. The wrong results were not contained in phrase and frame train corpus of dependency parsing.

3 SR Phrase Identification

In this paper, we divide SR phrase into two classes: Max SR phrase and Base SR phrase. Max SR phrase refers to SR phrase is not included in any other SR phrase in a sentence. Base SR phrase refers to SR phrase does not include any other SR phrase in a SR phrase. Therefore, MSR phrase is divided into two classes: Max MSR (MMSR)

phrase and Base MSR (BMSR) phrase. PSR phrase was divided into two classes: Max PSR (MPSR) phrase and Base PSR (BPSR) phrase.

3.1 MMSR Phrase Identification based on Cascaded Conditional Random Fields

Reference (Qiaoli Zhou, 2010) is selected as our approach of MMSR phrase identification. The MMSR identifying process is conceptually very simple. The MMSR identification first performs identifying BMSR phrase, and converts the identified phrase to head. It then performs identifying for the updated sequence and converts the newly recognized phrases into head. The identification repeats this process until the whole sequence has no phrase, and the top-level phrase are the MMSR phrases. A common approach to the phrase identification problem is to convert the problem into a sequence tagging task by using the “BIEO” (B for beginning, I for inside, E for ending, and O for outside) representation. If the phrase has one word, the tag is E. This representation enables us to use the linear chain CRF model to perform identifying, since the task is simply assigning appropriate labels to sequence.

There are two differences between our feature set and Qiaoli (2010)’s:

- 1) We use dependency direction of word as identification feature, while Qiaoli (2010) did not use.
- 2) We do not use scoring algorithm which is used by Qiaoli (2010).

Direction Unigrams	$D_{-3}, D_{-2}, D_{-1}, D_0, D_{+1}, D_{+2}, D_{+3}$
Direction Bigrams	$D_{-2}D_{-1}, D_{-1}D_0, D_0D_{+1}, D_{+1}D_{+2}$
Word & Direction	W_0D_0

Table 2. Feature Templates of MMSR Phrase

Table 2 is additional new feature templates based on Qiaoli (2010). W represents a word, and D represents dependency direction of the word. With this approach, nested MSR phrases are identified, and the top-level MSR phrase is the MMSR that we obtained.

corpus	P	R	F
dev	81.41%	75.40%	78.29%
test	81.23%	73.04%	76.92%

Table 3. MMSR Identification Performance

3.2 BMSR Phrase Identification based on CRFs

We use the tag set “BIEO” the same as that used for MMSR identification.

Word Unigrams	$W_{-3}, W_{-2}, W_{-1}, W_0, W_{+1}, W_{+2}, W_{+3}$
Word Bigrams	$W_{-3}W_{-2}, W_{-2}W_{-1}, W_{-1}W_0, W_0W_{+1}, W_{+1}W_{+2}, W_{+2}W_{+3}$
POS Unigrams	$P_{-3}, P_{-2}, P_{-1}, P_0, P_{+1}, P_{+2}, P_{+3}$
POS Bigrams	$P_{-3}P_{-2}, P_{-2}P_{-1}, P_{-1}P_0, P_0P_{+1}, P_{+1}P_{+2}, P_{+2}P_{+3}$
Word_X	X_0
Word_Y	Y_0
Word_D	D_0
Word_S	$S_{-3}, S_{-2}, S_{-1}, S_0, S_{+1}, S_{+2}, S_{+3}$
Word & POS	$W_{-1}P_{-1}, W_0P_0, W_{+1}P_{+1}$
Word & Word_X	$W_{-3}X_0$
Word & Word_D	$W_0D_0, W_{-3}W_{-2}D_0, W_{-2}W_{-1}D_0, W_{-1}W_0D_0, W_0W_{+1}D_0, W_{+1}W_{+2}D_0, W_{+2}W_{+3}D_0$
Word & Word_S	$W_{-1}S_{-1}, W_0S_0, W_{+1}S_{+1}, W_{+2}S_{+2}$
Word_X & Word_Y	X_0Y_0
POS & Word_D	$P_0D_0, P_{-3}P_{-2}D_0, P_{-2}P_{-1}D_0, P_{-1}P_0D_0, P_0P_{+1}D_0, P_{+1}P_{+2}D_0, P_{+2}P_{+3}D_0$
POS & Word_S	$P_{-1}S_{-1}, P_{-2}S_{-2}, P_{-3}S_{-3}, P_0S_0, P_{+1}S_{+1}, P_{+2}S_{+2}, P_{+3}S_{+3}$
Word_D & Word_S	$D_{-1}S_{-1}, D_{-2}S_{-2}, D_{-3}S_{-3}, D_0S_0, D_{+1}S_{+1}, D_{+2}S_{+2}, D_{+3}S_{+3}$
Word & POS & Word_D	$W_{-1}P_{-1}D_0, W_0P_0D_0, W_{+1}P_{+1}D_0$
Word & POS & Word_D & Word_S	$W_{-3}P_{-3}D_{-3}S_{-3}, W_{-2}P_{-2}D_{-2}S_{-2}, W_{-1}P_{-1}D_{-1}S_{-1}, W_0P_0D_0S_0, W_{+1}P_{+1}D_{+1}S_{+1}, W_{+2}P_{+2}D_{+2}S_{+2}, W_{+3}P_{+3}D_{+3}S_{+3}$

Table 4. Feature Templates of BMSR Phrase

In table 4, “W” represents a word, “P” represents the part-of-speech of the word, “X” represents the fourth word following the current word, “Y” represents the fifth word following the current word, “D” represents the dependency direction of the current word, and “S” represents the paired punctuation feature. “S” consists of “RLIO” (R for the right punctuation, L for the left punctuation, I for the part between the paired punctuation and O for outside).

corpus	P	R	F
dev	79.32%	80.65%	79.98%
test	79.22%	79.96%	79.59%

Table 5. BMSR Identification Performance (F-score)

3.3 MPSR Phrase Identification Based on Collection

Reference (Dongfeng, 2011) is selected as our approach of MPSR phrase identification. The position of HP in PSR phrase is not fixed. Not only PSR phrase is identified, but also PSR phrase type is identified.

There are two major differences between our feature set and Dongfeng (2011)’s:

- 1) We take the PSR phrase type (the SR of HP) as tag.
- 2) We use “S-type” represents that the PSR phrase is the single preposition. “Type” represents SR of the preposition.

For example: 工作者/NN location [在/P 甘肃/NR 金川/NR] 发现/VV

O W	POS	Dongfeng (2011) Tag	Our Tag
* 工作者	NN	O	O
* 在	P	O	O
在甘肃	NR	I	I
在金川	NR	E	Location-E
在发现	VV	N	N

Table 6. Example of PSR Phrase Tag Set

In table 6, Dongfeng(2011) takes ‘E’ as the tag of last word of PSR phrase, but we take ‘**Location-E**’ as the tag of last word of PSR phrase (Location is type of PSR phrase).

With this approach, nested PSR phrases are identified, and the top-level PSR phrase is the MPSR that we obtained.

corpus	MPSR phrase	MPSR phrase & type
dev	84.00%	54.23%
test	83.78%	51.60%

Table 7. MPSR Identification Performance (F-score)

3.4 Combined Identification of MSR Phrase and PSR Phrase

Identification process: MSR phrase and PSR phrase are respectively identified in one sentence, and the results are combined in accordance with this rule: if phrases are nested, only the top-level phrase is tagged; if phrases are same, only the PSR

phrase is tagged; if phrases are overlapped, only PSR phrase is tagged.

There are two combinations in this paper:

- 1) MMSR phrase and MPSR phrase combined result is defined as MMMP phrase. For example as follow (‘[]’ represents MMSR, ‘{ }’ represents MPSR):

Example A: [建筑/NN] 是/VC [开发/VV 浦东/NR 的/DEC 一/CD 项/M 主要/JJ 经济/NN 活动/NN], /PU 这些/DT 年/M 有/VE [数百/CD 家/M 建筑/NN 公司/NN 、 /PU 四千余/CD 个/M 建筑/NN 工地/NN] 遍布/VV location{ 在/P 这 /DT 片/M 热土/NN 上/LC } 。 /PU

MMMP frame: [建筑/NN] 是/VC 活动/NN , /PU 这些/DT 年/M 有/VE 工地/NN 遍布/VV location/location 。 /PU

- 2) BMSR phrase and MPSR phrase combined result is defined as BMMP phrase.

Example B: [建筑/NN] 是/VC 开发/VV [浦东 /NR] 的/DEC 一/CD 项/M 主要/JJ 经济/NN 活动/NN , /PU 这些/DT 年/M 有/VE [数百/CD 家 /M 建筑/NN 公司/NN 、 /PU 四千余/CD 个/M 建筑/NN 工地/NN] 遍布/VV location{ 在/P 这 /DT 片/M 热土/NN 上/LC } 。 /PU

BMMP frame: 建筑/NN 是/VC 开发/VV 浦东 /NR 的/DEC 一/CD 项/M 主要/JJ 经济/NN 活动 /NN , /PU 这些/DT 年/M 有/VE 工地/NN 遍布 /VV location/location 。 /PU

corpus	phrase	P	R	F
dev	BMMP	79.48%	81.60%	80.53%
	MMMP	80.00%	76.79%	78.36%
test	BMMP	80.14%	82.48%	81.30%
	MMMP	80.19%	78.53%	79.35%

Table 8. Combination Phrase Identification Performance

3.5 Phrase and Frame Length Distribution

We count phrases, frame and Original Sentence (OS) length distribution in training set and dev set.

	BMMP	MMMP	MMSR	BMSR	OS
[0,5)	80.07%	71.36%	75.36%	85.74%	9.07%
[5,10)	16.15%	21.63%	18.93%	12.33%	8.30%
[10,20)	3.35%	6.13%	5.05%	1.80%	17.23%
20≤	0.43%	0.88%	0.66%	0.13%	65.40%

Table 9. Length Distribution of Phrases and OS

Table 9 shows, about 95% of phrases have less than 10 words, but about 65% of OS has more than 20 words.

	BMMP	MMMP	MMSR	BMSR	OS
[0,5)	16.00%	18.70%	16.43%	14.36%	9.07%
[5,10)	18.87%	24.91%	19.41%	14.11%	8.30%
[10,20)	34.26%	35.42%	33.94%	30.68%	17.23%
20≤	30.87%	20.97%	30.22%	40.85%	65.40%

Table 10. Length Distribution of Frames and OS

Table 10 shows, about 70% of frames have less than 20 words, especially 80% of MMMP frame has less than 20 words, but about 65% of OS has more than 20 words.

	BMMP	MMMP	BMSR	MMSR	OS
phrase	3.07	3.83	2.53	3.44	30.07
frame	16.00	13.21	19.16	15.79	30.07

Table 11. Average Length

We count phrases, frame and Original Sentence (OS) Average Length (AL) in training set and dev set. Table 11 shows phrase of AL accounted for 10% of OS of AL, and frame of AL accounted for 50% of OS of AL. The AL shows that the semantic dependency parsing unit length of OS is greatly reduced after dividing an original sentence into SR phrases and frame.

As shown in tables 9, 10 and 11, the length distribution indicates that the divide-and-conquer strategy reduces the complexity of sentences significantly.

4 Semantic Dependency Parsing

Graph-based parser is selected as our basic semantic dependency parser. It views the semantic dependency parsing as problem of finding maximum spanning trees (McDonald, 2006) in directed graphs. In this paper, phrase and frame semantic dependency parsing result was obtained by Graph-based parser. Training set of phrase comes from phrases, and training set of frame comes from frames.

5 Experiments

5.1 Direction of Identification

Dependency direction serves as feature of SR phrase identification, so we need to identify dependency direction of word. We use tag set is {B, F}, B represents backward dependence, F represents forward dependence. The root's dependency direction in sentence is B. Dependency direction identification p-score has reached 94.87%.

Word Unigrams	$W_{-4}, W_{-3}, W_{-2}, W_{-1}, W_0, W_{+1}, W_{+2}, W_{+3}, W_{+4}$
Word Bigrams	$W_{-3}W_{-2}, W_{-2}W_{-1}, W_{-1}W_0, W_0W_{+1}, W_{+1}W_{+2}, W_{+2}W_{+3}$
Word Trigrams	$W_{-1}W_0W_{+1}$
Word Four-grams	$W_{-2}W_{-1}W_0W_{+1}, W_0W_{+1}W_{+2}W_{+3}$
Word Five-grams	$W_{-4}W_{-3}W_{-2}W_{-1}W_0, W_0W_{+1}W_{+2}W_{+3}W_{+4}$
POS Unigrams	$P_{-4}, P_{-3}, P_{-2}, P_{-1}, P_0, P_{+1}, P_{+2}, P_{+3}, P_{+4}$
POS Bigrams	$P_{-3}P_{-2}, P_{-2}P_{-1}, P_{-1}P_0, P_0P_{+1}, P_{+1}P_{+2}, P_{+2}P_{+3}$
POS Trigrams	$P_{-1}P_0P_{+1}$
POS Four-grams	$P_{-2}P_{-1}P_0P_{+1}, P_0P_{+1}P_{+2}P_{+3}$
POS Five-grams	$P_{-4}P_{-3}P_{-2}P_{-1}P_0, P_0P_{+1}P_{+2}P_{+3}P_{+4}$
Word & POS	$W_{-2}P_{-2}, W_{-1}P_{-1}, W_0P_0, W_{+1}P_{+1}, W_{+2}P_{+2}$

Table 12. Feature Templates of Dependency Direction In table12, w represents word, p represents POS.

5.2 System and Model

For a sentence for which phrases has been identified, if phrases can be identified, then the whole sentence semantic dependency parsing result is obtained by phrase parsing model and frame parsing model. Therefore, in this paper, the sentence is divided into the following types based on the phrase identification results: (1) SentMMMP indicates MMSR phrase and MPSR phrase identified in a sentence; (2) SentBMMP indicates BMSR phrase and MPSR phrase identified in a sentence; (3) SentMMSR indicates only MMSR phrase identified in a sentence; (4) SentMPSR indicates only MPSR phrase identified in a sentence; (5) SentBMSR indicates only BMSR phrase identified in a sentence; (6) SentNone indicates no phrase identified in a sentence.

Sentence type	Phrase parsing Model	Frame parsing Model
SentMMMP	MMMP phrase	MMMP frame
SentBMMP	BMMP phrase	BMMP frame
SentMMSR	MMSR phrase	MMSR frame
SentMPSR	MPSR phrase	MPSR frame
SentBMSR	BMSR phrase	BMSR frame
SentNone	Sentence model	

Table 13. Type of Sentence and Parsing Model

Table 13 shows types of sentence, and parsing models for every type of sentence. For example, parsing SentMMMP needs MMMP phrase parsing model and MMMP frame parsing model

The corpus contains the sentence type determined by the phrase identification strategy.

Strategy of phrase identification	Sentence type in the corpus
Strategy MMMP	SentMMMP, SentMMSR, SentMPSR, SentNone
Strategy BMMP	SentBMMP, SentMPSR, SentBMSR, SentNone
Strategy BMSR	SentBMSR, SentNone

Table 14. Sentence Types in the Corpus

As shown in table 14, Strategy MMMP indicates that MMMP phrase in the corpus was identified, and sentences in the corpus were divided into SentMMMP, SentMMSR, SentMPSR and SentNone. Strategy BMMP indicates that BMMP phrase in the corpus was identified, and sentences in the corpus were divided into SentBMMP, SentBMSR, SentMPSR and SentNone. Strategy BMSR indicates that BMSR phrase in the corpus was identified, and sentences in the corpus were divided into SentBMSR and SentNone.

5.3 Comparative Experiments

In this paper, we carry out comparative experiments of parsing for the test set by 3 systems.

- 1) System1 represents strategy MMMP in the table 14.
- 2) System2 represents strategy BMMP in the table 14.
- 3) System3 represents strategy BMSR in the table 14.

	Dev	Test
G-parser	62.31%	61.68%
System1(MMMP)	61.98%	61.84%
System2(BMMP)	62.7%	62.08%
System3(BMSR)	62.22%	61.15%

Table 15. Comparative Experiments

As shown in the table 15, system2 result is more accurate than system1, because BMMP phrase identification is more accurate than MMMP as shown in the table 8. Although, BMSR phrase identification is more accurate than MMMP phrase as shown in the table 5 & 8, system 3 result is less accurate than system1. Compared with BMSR iden-

tification, MMMP identification reduces the complexity of sentences significantly, because the table 11 shows that the AL of MMMP frame is about 30% less than that of BMSR frame. G-parser is graph-based parser (Wangxiang Che, 2008).

6 Conclusion and Future Work

To solve the problem of low accuracy of long distance dependency parsing, this paper proposes a divide-and-conquer strategy for semantic dependency parsing. We present our SemEval2012 shared Task 5 system which is composed of three cascaded components: the tagging of SR phrase, the identification of Semantic-role- phrase and semantic dependency parsing.

Divide-and-conquer strategy is influenced by two factors: one is identifying the type of phrase will greatly reduce the sentence complexity; the other is phrase identifying precision results in cascaded errors. The topic of this evaluation is semantic dependency parsing, and word and POS contain less semantic information. If we can make semantic label on words, then it will be more helpful for semantic dependency parsing. In the future, we will study how to solve the long distance dependency parsing problem.

Acknowledgments

The authors would like to thank the reviewers for their helpful comments. This work was supported by National Natural Science Foundation of China (NSFC) via grant 61073123 and Natural Science Foundation of Liaoning province via grant 20102174.

References

- Dongfeng Cai, Ling Zhang, Qiaoli Zhou and Yue Zhao. *A Collocation Based Approach for Prepositional Phrase Identification*. IEEE NLPKE, 2011.
- McDonald, Ryan. 2006. *Discriminative Learning and Spanning Tree Algorithms for Dependency Parsing*. Ph.D. thesis, University of Pennsylvania.
- Guiping Zhang, Wenjing Lang, Qiaoli Zhou and Dongfeng Cai. 2010. *Identification of Maximal-Length Noun Phrases Based on Maximal-Length Preposition Phrases in Chinese*, 2010 International Conference on Asian Language Processing, pages 65-68.
- Qiaoli Zhou, Wenjing Lang, Yingying Wang, Yan Wang, Dongfeng Cai. 2010. *The SAU Report for the*

1st CIPS-SIGHAN-ParsEval-2010, Proceedings of the First CIPS-SIGHAN Joint Conference on Chinese Language Processing, pp:304-311.

Wangxiang Che, Zhenghua Li, Yuxuan Hu, Yongqiang Li, Bing Qin, Ting Liu, and Sheng Li. 2008. *A cascaded syntactic and semantic dependency parsing system*. In CoNLL-2008.