Toward Discourse-guided Theta-grid Chart Parsing for Madarin Chinese -- A Preliminary Report

Koong H. C. Lin and Von-Wun Soo
Department of Computer Science, National Tsing-Hua University HsinChu,
Hsinchu, 30043, Taiwan, R.O.C.
E-Mail:soo@cs.nthu.edu.tw

Abstract

An attempt for this work is to show a way of combining word identification, syntactic processing, semantic processing, and discourse processing into a cohesive framework. We utilize thematic information as a media to integrate these processing modules. In this work, the thematic information is assumed to reside in lexicons based on the theta grid theory. For determining the main verb(s) of a sentence with Serial Verb Constructions (SVCs), we propose an algorithm which evaluates a scoring function; examples showing how Chinese texts with SVCs in the legal domain can be parsed by our parser are presented. We also show how the previously acquired anaphoric knowledge can be used to guide the chart parser.

1 Introduction

Traditional natural language processing (NLP) systems are normally composed of many standing alone modules to perform individually and sequentially the word identification, the syntactic processing, the semantic processing, and the discourse processing. However, problems such as PP-atachments, anaphora, and structural ambiguities cannot be easily resolved if these modules are not cohesively interacted with each other. Thus, some attempts were made to integrate these modules by enhancing interactions between modules, or making the boundaries between modules vague. *Preference semantics* [Wilks75] [Fass83], *case-based parser* [Martin89], *expectation-driven partial parsing* [Rau87], and *conceptual parsing* [Shank73], were paradigms of such attempts. In this work, we also aim at integrating these modules into a cohesive framework, in which thematic information plays a significant role. We also make use of the anaphoric knowledge acquired by our previous work, *G-UNIMEM* [Chen92], to "*predict anaphora*" during parsing.

A sketch of our work is illustrated in [figure 1]. The functions of each module are briefly described as follows: the *TG-Chart parser*, which accepts the input from the word identifier, and interacts with the Discourse Daemon, is the core of our work. Syntactic knowledge in grammar rules, and thematic information in lexicons, are the two major knowledge used by TG-Chart parser. The *partial word identifier* partially identifies the input sentence, and constructs a word lattice, which serves an input for the parser. The most interesting module, the *Discourse Daemon*, utilizes a set of G-Rules to make prediction for occurrences of anaphora, where G-Rules are acquired by G-UNIMEM. More precise descriptions of each module will be given in the following sections.

This work serves as a natural language front-end for our long-term research of a verdict understanding system. Thus, the corpora we use are some judicial verdict documents from the Kaohsiung district court [臺90a] [臺90b], which were written in a special official-document style. Thus, our analysis is based on such a kind of sub-language.

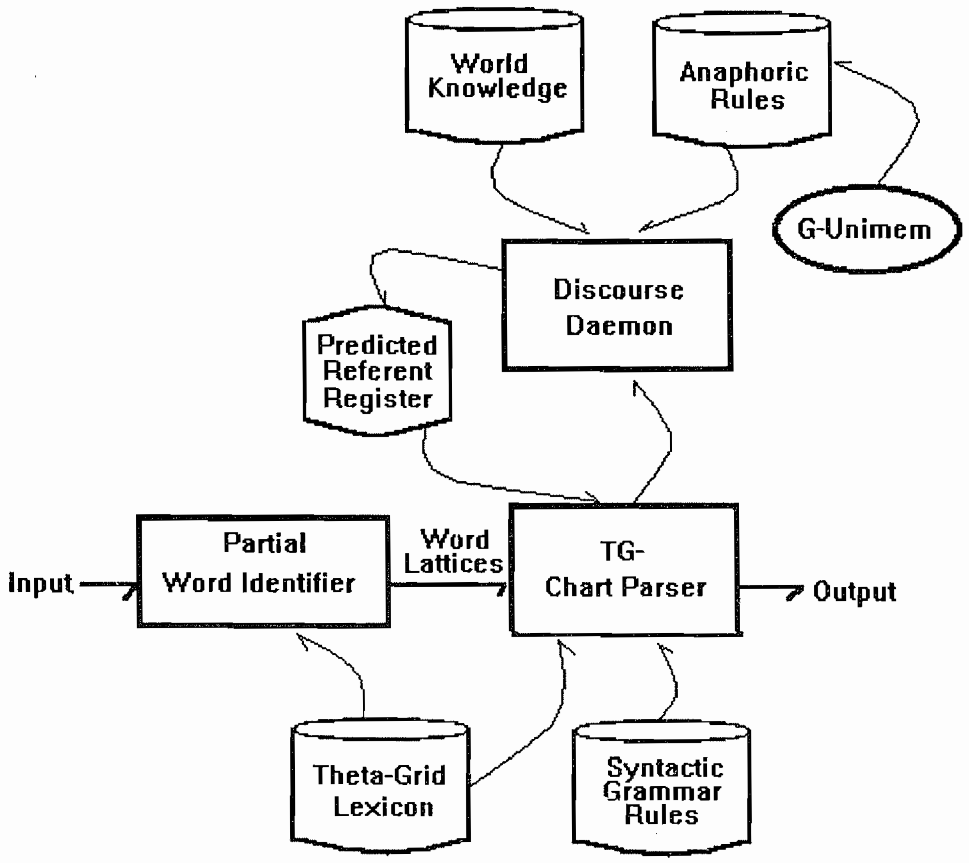


Figure 1. A schematic framework of Discourse-guided Theta-grid chart parser

2 The TG-Chart Parser

2.1 The Theta-Grid Theory and the Chart Parser

Thematic information is one of the information sources that can bridge the gap between syntactic and semantic processing phases. Tang proposed a *theta-grid theory* [湯 92] in which rich thematic information is incorporated for the analysis of human languages. The idea of theta-grid theory is as follows: we use a predicate, say, a verb, as the *center* of a "grid" and, by finding the theta-roles registered in the lexical entries of this predicate, we can construct a grid formed by this predicate and then construe the sentence (or clause) spanned by this predicate. As we know, Chinese is not sensitive to syntax; therefore, the theta-grid seems to be suitable for processing Chinese. However, to computationalize theta-grid theory, some control strategies for parsing must be included.

The well-known *chart parser* [Kay80] [Allen87], which utilizes a data structure called "*chart*" to record the partial parsing results, is suitable for our work. Since it considers all possible combinations of constituents, it is more flexible and can accept sentences with missing theta roles. Thus, we design a modified chart parser called TG-Chart parser by combining the theta-grid theory and the chart parser. Note that currently in our work, only the theta grids for *verbs* are considered. For each verb, there are two kinds of theta roles registered: the *obligatory roles*, which *must* be found for this verb to construct a legal "grid"; the *optional roles*, with their appearance being optional. Take "告訴" for example, its theta roles are registered as: *+[Th (Pd) Ag]*; thus, two *NPs* must be found in the chart for the construction of a legal grid (From *syntactic clues*, both *"Ag"* and *"Th"* are always played by *NPs* according to Liu. [Liu93].), while the appearance of a clause to serve as a "Pd" role is optional. Besides, some theta roles, like Qd, Ma, Ti,..,etc., are not registered in the lexical entries of individual verbs, since they occur commonly in grids formed by every kinds of verbs.

A brief description of our parsing algorithm is as follows:

[Step 1] Search the sentence for positions of all "possible verbs". (what we call *possible verbs* are those words with *verb*-category as one of its syntactic categories)

[Step 2] By considering all possible combinations, the chart parser groups the words into *syntactic constituents*. Syntactic knowledge is used in this step.

[Step 3] If only one verb is found in [Step 1], search the chart for constituents which can play the theta roles of this verb.

[Step 4] If more than one verb are found, more complex considerations are needed. We will discuss such a situation more detailedly in the next section.

Note that currently in our work, only simple information is encoded in the lexicon. Thus, we need a small set of syntactic grammar rules, and a syntactic chart parser to group the phrases. While the lexicon is enlarged and enriched, it seems better to drop the syntactic grammar rules, and drive our parser toward *information-based* and *unification-based*. The successful ICG parser [Chen 89] [Chen 90] is a paradigm for our further development.

2.2 Dealing with Serial Verb Constructions

Serial verb construction (SVC) is a unique construct of the Chinese language, which refers to a sentence containing two or more VPs juxtaposed without any marker indicating what the relationships are between verbs [Li81]. Many works are reported on processing different sorts of SVCs. Some of them are rule-based [Chang91], some are lexicon-driven based on Case Grammar [Yeh92] [Pun91] [Fillmore68]. Linguists classified SVCs into five types: *two and more separate events*, *pivotal construction*, *descriptive clauses*, *sentential subjects*, and *sentential objects*. This classification is the basis for their analysis. In our legal domain corpora, there are also many occurrences of SVCs. Since our parser is based on the theta grids, in case of SVCs, different verbs will *compete* in finding their own theta roles. Thus, some mechanism for arbitrating among verbs for the ownerships of each constituent in the chart must be designed. According to Yorick Wilks, *language does not always allow the formation of "100%-correct" theories* [Hirst81]; therefore, we attempt to find a more flexible method for recognizing SVCs. We propose a *scoring function* to select a "preferable" construction for the sentence with SVCs. The scoring function is defined as follows, where RWR is the abbreviation of "Ratio of Words included in some phrase with Roles assigned", OBR, "OBligatory Role", and OPR, "OPtional Role" (Note that OBR and OPR indicate those roles *registered in theta grids*.):

$$Score = \frac{\sum_{\text{every verb}} Score - Per - Verb}{\text{number of verbs}} \qquad (3.3.1)$$

$$Score - Per - Verb = \frac{[(\text{number of OBR found}) * 2 + (\text{number of OPR fcund})]}{Base} * RWR \quad (3.3.2)$$

$$RWR = \frac{\text{number of words included in some phrase with roles assigned}}{\text{number of words in the clause}} \qquad (3.3.3)$$

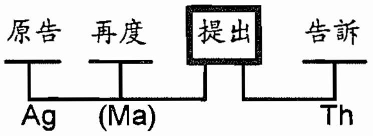
$$Base = 2 * (\text{number of OBR}) + (\text{number of OPR}) \quad (3.3.4)$$

The score is calculated as the average value of scores obtained by each verb in the sentence (as in equation 3.3.1). For each verb, the score is estimated by two factors: *first*, the ratio of theta roles found, and, *second*, the ratio of words with roles assigned , i.e., RWR. For precise calculation, see equation (3.3.2). We heuristically weigh the relative significance between obligatory roles and optional roles by *2:1*, as in (3.3.2) and (3.3.4). In some cases, the verb finds many theta roles in the clause it constructs, but the words in this clause are not all assigned roles. We consider such assignment doesn't construe the real construction of the sentence. Thus, to reflect such cases, we calculate RWR by dividing the number of words which are included in some phrase with a role assigned by the total number of words in the clause (see equation 3.3.3). Now, let's illustrate the calculation of this scoring function by the following examples:

【Example 1】"原告 再度 提出 告訴"

In [Step 1], "提出" and "告訴" are both found as "possible verbs". Here "告訴" has two syntactic categories registered in its lexical entry: verb and noun, while "提出" has only one categorty, the verb. The theta grid for "提出" is +[Th Ag], "告訴", +[Th (Pd) Ag]. So, to decide whether "告訴" is treated to a verb or a noun, there are four cases to be considered:

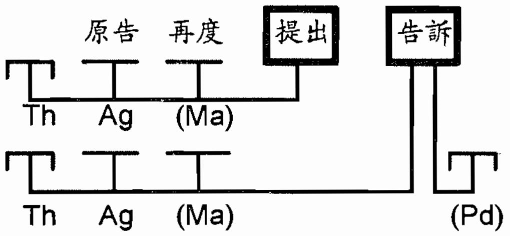
(1) "提出" is treated as a verb, while "告訴" a noun.



In the above, "提出" enveloped by a *box* means it plays a verb. When it searches for theta roles, "原告" and "告訴" are respectively found as its Ag and Th, the two obligatory theta roles registered in its lexical entry. In addition, "再度" is found as an optional role, Ma. However, Ma is not registered in the lexical entry of "提出", it contributes no credit for "提出". Now, the score is calculated as follows:

For "提出", there are two obligatory roles, so Base = 2*2 = 4. Moreover, in this sentence, "原告","再度","提出",and "告訴" are all assigned some roles; thus, RWR = 4/4 = 1. And then, Score-Per-Verb = {[(number of OBR found)*2 + (number of OPR found)]/Base} * RWR = {[2*2+0]/4}*1=1. Finally, Score = 1/1 = 1.00.

(2) "提出" and "告訴" both are treated as verbs.

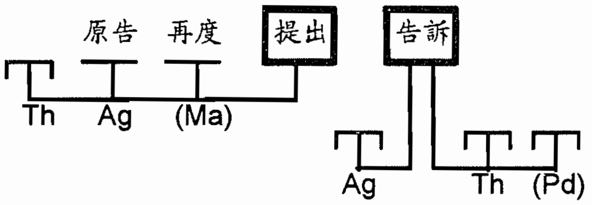


In the above figure, "提出" cannot find its Th, "告訴" cannot find its Th and Pd. Such "*cannot find*" situations are represented by the symbol "┌─┐".

For "提出", Base=4. Note that for the portion of sentence centered by "提出", "原告 再度 提出", every word is assigned a role; thus, RWR = 3/3 = 1. Score-Per-Verb = {[1*2]+0]/4}*1=0.5.

For "告訴", Base=2*2+1=5. RWR = 3/3 = 1. Score-Per-Verb = {[1*2+0]/5}*1=0.4. So, for this case, Score = (0.5+0.4)/2 = 0.45.

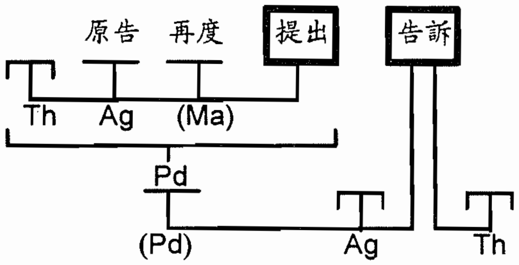
(3) "提出" and "告訴" both are treated as verbs, while "告訴" is subordinated to "提出".



For "提出", Base=4. Since "告訴" is subordinated to "提出", but the clause it forms cannot play any role for "提出", the RWR for "提出" is 3/4 = 0.75. Score-Per-Verb = {[1*2+0]/4}*0.75 =0.375.

For "告訴", it is clear that Score-Per-Verb is 0, because it cannot find any role. So, Score = (0.375+0)/2 = 0.1775

(4) "提出" and "告訴" both are treated as verbs, while "提出" is subordinated to "告訴".



For "提出", Base=4. RWR=3/3=1. Score-Per-Verb = {[1*2+0]/4}*1 = 0.5. The clause constructed by "提出" supports a Pd role for "告訴". Thus, for "告訴", RWR=4/4=1; Score-Per-Verb = {[0*2+1]/5}*1 = 0.2.

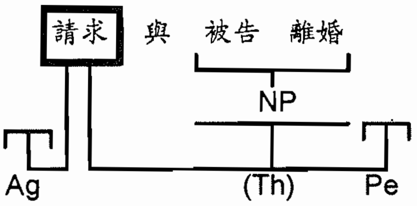
Score = (0.5+0.2)/2 = 0.35.

From the above discussions, case(1) apparently gets the highest score (1.00). So, the parsed structure in case(1) is preferable to those in the other cases. That is, in this sentence, "提出" plays as the only verb, while "告訴" plays a noun. Therefore, the right syntactic category for "告訴" in this sentence is determined.

【Example 2】"請求 與 被告 離婚"

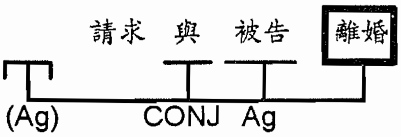
In [Step 1], "請求" and "離婚" are both found as "possible verbs". Here "請求" and "離婚" both have two syntactic categories registered in its lexical entry: the verb and the noun. The theta grid for "請求" is +[(Th) Pe Ag], "離婚" +[Ag (Ag)]. So, there are five cases to be considered:

(1) "請求" plays as the only verb, while "離婚" plays as a noun.



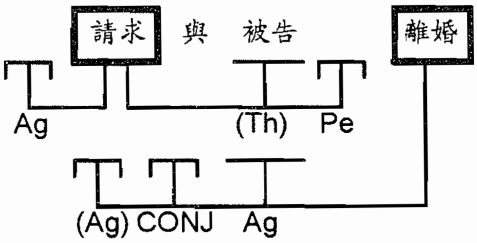
For "請求", Base=5. Since "與" can't play any role in this sentence, RWR = 3/4 = 0.75. Score-Per-Verb = {[0*2+1]/5}*0.75=0.15. So, Score = 0.15/1 = 0.15.

(2) "離婚" is treated as a verb, while "請求" a noun.



For "離婚", Base=3. Note that although "請求" is an NP, it cannot play as Ag for "離婚". It is because it doesn't satisfy the constraint for playing as Ag: an Ag must has a feature "+animate", according to Gruber's theory that an agent must be *an entity with intentionality* [Gruber76]. So, RWR = 3/4 = 0.75, and Score-Per-Verb = {[1*2+0]/3}*0.75 = 0.5. Score = 0.5/1 = <u>0.5</u>.

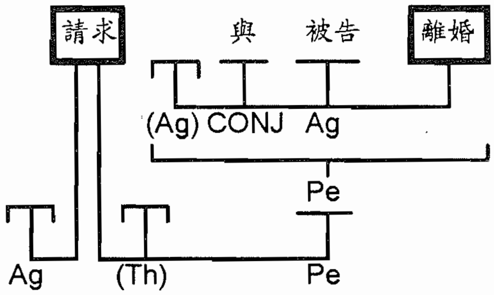
(3) "請求" and "離婚" both are treated as verbs.



For "請求", Base=5, RWR=2/3=0.67, since "與" doesn't play any role. Score-Per-Verb = {[2*0+1]/5}*0.67=0.134.

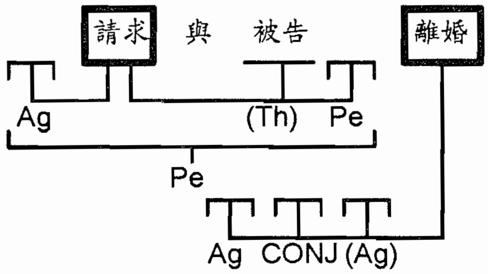
For "離婚", Base=3. RWR=3/3=1. Score-Per-Verb = {[1*2+0]/3}*1 = 0.67. Score = (0.134+0.67)/2 = <u>0.402</u>.

(4) "請求" and "離婚" both are treated as verbs,with "離婚" being subordinated to "請求"



For "請求", Base=5. RWR=4/4=1. Score-Per-Verb = {[1*2+0]/5} = 0.4.
For "離婚", Base=3. RWR=3/3=1. Score-Per-Verb = {[1*2+0]/3} = 0.67.
Score = (0.4+0.67)/2 = <u>0.535</u>.

(5) "請求" and "離婚" both are treated as verbs,with "請求" being subordinated to "離婚".



For "請求", RWR=2/3=0.67. Score-Per-Verb = {[2*0+1]/5}*.0.67 = 0.134.
For "離婚", it's clear that Score-Per-Verb = 0.
Score = (0.134+0)/2 = <u>0.067</u>.

From the above discussions, **case(4)** apparently gets the highest score (<u>0.535</u>). So, the parsed structure in case(4) is preferable to those in the other cases. That is, in this sentence, "請求" and "離婚" both are treated as verbs, while "離婚" is subordinated to "請求". The clause constructed by "離婚" is assigned the Pe role for "請求". It is one kind of Serial Verb Construction. (This kind of SVC is commonly called "*sentential objects*".)

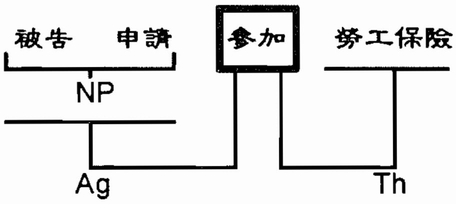
In table 1, we show the results of more sentences with SVC in the legal documents which are parsed by this scheme in our TG-Chart parser. The sample sentences are as follows:

S1: 原告 訴請 被告 給予 三十萬元
S2: 原告 請求 被告 清償 債務
S3: 被告 未 到 場 爭執
S4: 被告 於 民國七十八年 十一月二十日 突 無故 離家 出走
S5: 被告 未 返 家 與 原告 同居
S6: 原告 聲請 訊問 證人
S7: 被告 希望 原告 能 諒解
S8: 被告 申請 參加 勞工保險
S9: 原告 通知 被告 改善
S10: 原告 本人 到 場 對質
S11: 被告 否認 有 過失
S12: 原告 訴請 被告 負擔 訴訟費

Sen. No.	Verb Candidates	Verb Players	Relatioships	Highest Score	Correctness
S1	v1: 訴請 v2: 給予	v1,v2	v1>v2	1.00	Y
S2	v1: 請求 v2: 清償	v1,v2	v1>v2	1.00	Y
S3	v1: 到 v2: 爭執	v1,v2	v1=v2	1.00	Y
S4	v1: 離家 v2: 出走	v1,v2	v1=v2	1.00	Y
S5	v1: 返 v2: 同居	v1,v2	v1=v2	0.83	Y
S6	v1: 聲請 v2: 訊問	v1,v2	v1>v2	0.70	Y
S7	v1: 希望 v2: 諒解	v1,v2	v1>v2	0.84	Y
S8	v1: 申請 v2: 參加	v2		1.00	N
S9	v1: 通知 v2: 改善	v1,v2	v1>v2	0.83	Y
S10	v1: 到 v2: 對質	v1,v2	v1=v2	0.88	Y
S11	v1: 否認 v2: 有	v1,v2	v1>v2	0.75	Y
S12	v1: 訴請 v2: 負擔	v1,v2	v1>v2	1.00	Y

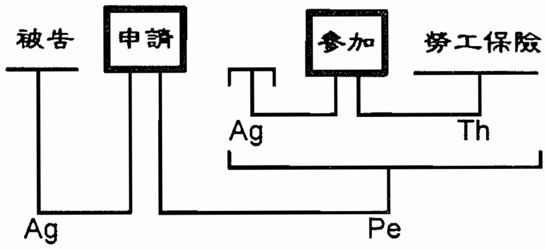
Table 1. More sample sentences with SVCs

The notation "v1>v2" means v2 is subordinated to v1, "v1=v2", no subordination relations exist between verbs.. In the above table, the result for S8 is incorrect. Analyzing this sentences, we find that it is caused by the *incorrect formation of compound nouns*. In S8, both "申請" and "參加" are the *possible verbs*, while "申請" has two syntactic categories: the verb and the noun. When "參加" is treated as the only verb, while "申請" a noun, the combination is as follows:



"被告" and "申請" incorrectly combine together and form the incorrect compound noun: "被告申請". Thus, "參加" finds "被告 申請" and "勞工保險" as its Ag and Th, respectively; and, moreover, obtains a high score: 1.00.

However, we know that the correct combination should be as follows, where "申請" and "參加" are both treated as verbs:



It obtains the *secondly* high score: 0.75, although it is the *correct* construction. Thus, a study for the *Serial Noun Constructions*, such as the work reported by Yeh et al. [Yeh91], is one of our future works.

3 The Discourse Daemon

3.1 G-Rules

The *anaphora* problem plays a significant role in natural language processing systems. The *raise-bind mechanism* [Lin86], which was based upon the *empty categories*, supports a mechanism for resolving *intra-sentential* anaphora. Many problems arise during discourse processing. However, currently in our work, we forcus on the resolution of *inter-sentential* anaphora problems only. In discourse, there may be anaphora in two consecutive sentences [Li86]. Examples of anaphoric ambiguities will be shown later. When anaphora appear in a pair of consecutive sentences, the two consecutive sentences are called *conjoined sentences*. In our work, the Discourse Daemon is just the module which resolves the anaphora problems in conjoined sentences. In Anaphora Daemon, two kinds of knowledge may be used: (1) *Anaphoric rules* generated by G-UNIMEN. (2) *Domain knowledge* However, in this paper, only anaphoric rules are reported. In the following, we concentrate on the usage of anaphoric rules.

The anaphoric rules used by Discourse Daemon are called *G-Rules*. We use G-Rules to predict and resolve the anaphora occurring in conjoined Chinese sentences. Here we show some sample G-Rules following:

```
1. [ante(agent), type(nil)]
   :- [f1(agent), anaphor(agent)]
2. [ante(theme), type(nil)]
   :- [f1(theme), anaphor(theme)]
3. [ante(agent), type(nil)]
   :- [f1(agent), f2(theme), anaphor(agent)]
4. [ante(agent), type(pronoun)]
   :- [f1(agent)]
5. [ante(theme), type(pronoun)]
   :- [f2(theme)]
6. [ante(theme), type(pronoun)]
   :- [anaphor(theme), f1(agent), f2(theme)]
```

G-Rules are written in a Prolog-like style. For rule 1, it means that if in the first sentence, one word plays the *agent* role (represented by "*f1(agent)*"), and in the second sentence, an *anaphor* occurs in the *agent position*, this anaphor will refer to the word which is in the *agent position* of the first sentence. (i.e., *the antecedent of this anaphor is the agent in the first sentence*.) In addition, the surface representation of this anaphor is *zero-pronoun*. (This is just what the notation "*type(nil)*" means.) We can use the following sentence to illustrate such rules:

[他]i 跌倒了, []i 很難過.
agent agent

In the first sentence, "他" can play the only role in this sentence as an agent; and, in the second sentence, an anaphor occurs at the agent position, represented as zero-pronoun. Thus, by rule 1, we know that this anaphor refers to "他" in the first sentence.

Similarly, we can use G-Rules to analyze another example:

[老張]i 娶了一個 [美嬌娘]j, []i 快樂極了.
agent theme agent

In the first sentence, "老張" and "美嬌娘" are treated as agent and theme respectively; and, in the second sentence, an anaphor occurs at the agent position, also represented as zero-pronoun. According to G-Rule 3, the antecedent of this anaphor is likely to be "老張", the agent of the first sentence. It is obviously a correct choice. From the above two examples, we find that G-Rules can be used to *resolve* and, besides, to *predict* the antecedents of anaphora occurring in conjoined Chinese sentences. In the next section, we will see how to utilize G-Rules in anaphora prediction.

3.2 Use G-Rules to Predict Anaphora

Let's observe an example directly, here a pair of sentences are conjoined:

原告再度提出告訴，請求與被告離婚

Due to the analyzed result in section 3.3, we know that in the first sentence, "提出" is treated as the only verb, while "原告" as its agent, and "告訴" as its theme:

[原告]i 再度提出 [告訴]j
Agent Theme

Search G-Rules for matched rules, we find that such situation in this sentence satisfies the first two conditions for rule 3:

```
[type(nil), ante(agent)]
:- [f1(agent), f2(theme), anaphor(agent)]
```

That is, there are both roles of agent and theme in the first sentence, which satisfies the f1 and f2 conditions in rule 3 respectively. Thus, if there is an anaphor which appears as zero-pronoun in the second sentence to be parsed later, the antecedent should be the agent in the first sentence, i.e., "原告". So, "原告" *is kept in a temporary register, Pred-Ref,* since it is likely to be the referent of the next sentence.

And then, when the second sentence is parsed (See the analysis in section 3.3, case(4) of example 2.), exactly as we expected, there is an anaphor at the agent position, with this agent ommitted:

[]i 請求 [Pe []i 與 [被告]j 離婚]
agent agent agent

So, "原告" kept in Pred-Ref is extracted to fulfill the agent position. And, moreover, "原告" continues propagating to the embedded clause, i.e., the Pe "與被告離婚", also fulfills the ommitted agent. (The theta role Pe means a proposition without a subject.) Therefore, we get a completely parsed sentence pair as follows:

[原告]i 再度提出 [告訴]j, [原告]i 請求 [原告]i 與 [被告]k 離婚
agent theme agent agent agent

Thus, the predictive power supported by G-Rules is exhibited.

4 Discussions

We have proposed a framework in which each component for natural language processing can be cohesively correlated. The Discourse Daemon, a module which utilizes a set of G-Rules, makes TG-Chart parser anaphora predictable. In our current status, the parser is implemented in C language. We believe it quite easy to incorporate Discourse Daemon into the parser. G-UNIMEM is well implemented, and gets accuracy rates 95.8% for resolving and 90.8% for choosing anaphora with 120 training instances. The reported accuracy supports the reliability for G-Rules.

There are some concerns in the further development of our system:

(1) The pre-defined feature set used to train G-UNIMEM could be extended, since in some cases these features do not seem to be adequate. Observe the following sentence:

[老張]i 娶了一個 [老婆]j, []j 很會做飯
agent theme agent

An anaphor occurs in the second sentence. It is obvious that the antecedent of this anaphor is "老婆", the theme in the first sentence. However, if we apply our G-Rules, "老張" will be selected due to rule 3, cause an incorrect prediction. We can easily see that such a link between this anaphor and its antecedent is caused by the relation: "老婆" is always the agent for "做飯" when we recall our previously seen cases. So, we think that this problem can be resolved either by adding such semantic features or applying world knowledg. How to integrate world knowledge is our furture work.

(2) There are two problems which might happen during the application of G-Rules: First, the G-Rules might not be 100%-accurate. Second, There might be cases where two possible predictions can be invoked by two G-Rules for the same sentence. Thus, more careful considerations are needed for the design and the application of G-Rules. In addition, how to

extend our work so that it could handle anaphora occurring in sentences that are *not simple conjoined sentences*, is also one of our concerns.

(3) In Madarin, either an NP, a PP, or a S can play a role. However, in our work, the possibility of a PP is not considered yet. It's also our future concern.

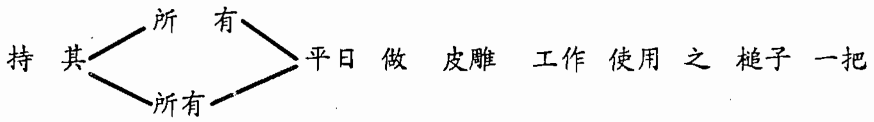
(4) Previous works always treate word identifier as a preprocessor of the subsequent parser. However, in the analysis of the corpora of the verdict documents, we found it was so complicated that it was impractical to isolate the word identifier to be a standing alone module. So, we attempt to integrate the word identifier and the parser into a cohesive one. That is, the word identifier finds coarse word boundaries first, and the parser refines the boundaries and produces the final correct word identifications.

Word lattices are often used in speech recognition and word identification [Lee91] [Carter92], since the word boundaries for the input always are not definite. Lee et al. found that the data structure, chart, used by chart parser, is suitable to represent such word lattices, and utilize a chart parser to parse the indefinite input, a word lattice, which are produced during the speech recognition module, to get a correct recognition. We find that the situation is similar to our work. So, we imitate the idea and propose a two-phase algorithm which will be developed in the future. Our algorithm is quite simple: In phase 1, for each character C in the sentence, word identifier searches the lexicon for all words that beginning by C, and keep the matched word to construct a word lattice. In phase 2, TG-Chart parser parses this word lattice and gets a correct identification.

As an example, for following sentence, it is not reasonable for a standing alone word identifier to achieve a 100%-correct identification:

"持其所有平日做皮雕工作使用之槌子一把"

In phase 1, word identifier constructs a partially identified word lattice:



In phase 2, TG-Chat parser is used to parse this lattice. We can correctly select the following identification, since the other possibilities will be filtered out during parsing:

持 其 所 有 平日 做 皮雕 工作 使用 之 槌子 一把

Thus, by combining the word identifier and the parser together, we think a better result will be obtained.

Acknowledge

We would like to thank Benjamin L. Chen, the G-UNIMEM he developed provides us useful knowledge for predicting anaphora.

References

[Allen87] James Allen, Natural Language Understanding, The Benjamin / Cummings Publishing Co. 1987.

[[Carter92] David M. Carter, Lattice-Based Word Identification in CLARE. In Proc. of 30th Annual Meeting of Association for Computational Linguistics (ACL-92).

[Chang91] Chao-Huang Chang and Gilbert K. Krulee, Prediction Ambiguity in Chinese and Its Resolution. Proc. of ICCPCOL 1991, pp.109-114.

[Chen89] 陳克健,黃居仁，訊息爲本的格位語法--一個適用於表達中文的語法模式，ROCLING II, 1989.

[Chen90] Keh-jiann Chen and Chu-Ren Huang, Information-based Case Grammar. In Proc. of COLING-90.

[Chen92] Benjamin L. Chen and Von-Wun Soo, An Acquisition Model for Both Choosing and Resolving Anaphora in Conjoined Madarin Chinese Sentences. In Proc. of COLING-92, pp. 274-279.

[Fass83] Dan Fass and Yorick Wilks, Preference Semantics, Ill-Formedness, and Metaphor. American Journal of Computational Linguistics, Vol.9, No.3-4, pp.178-187. 1983.

[Fillmore68] Fillmore, C., The Case for Case. In Universals in Linguistics Theory, ed. E. Bach and R.T. Harms. New York: Holt. (1968).

[Gruber76] Gruber J. S., Lexical Structures in Syntax and Semantics, North-Holland Publishing Company. 1976.

[Hirst81] In Graeme Hirst, Lecture Notes in Computer Science, Anaphora in Natural Language Understanding, A Survey. Springer-Verlag Berlin Heidelberg. 1981.

[Kay80] Martin Kay. Algorithm Schemata and Data Structures in Syntactic Processing. In Proc. of the Nobel Symposium on Text Processing, Gothenburg, 1980.

[Lee91] Lin-Shan Lee, Lee-Feng Chien, L.J. Lin, J. Huang, K.-J. Chen. An Efficient Natural Language Processing System Specially Designed for the Chinese Language. Computational Linguistics, 17(4), pp.347-378. 1991.

[Li86] Mei Du Li, Anaphori: Structure of Chinese, Student Book CO., Taipei, Taiwan. 1986.

[Li81] C. N. Li and S. Thompson, Mandarin Chinese: a Functional Reference Grammar, University of California Press, Berkeley. 1981.

[Lin86] Long-Ji Lin, James Huang, K.J. Chen, and Lin-Shan Lee, A Chinese Natural Language Processing System Based upon the Theory of Empty Categories. In Proc. of AAAI 1986.

[Liu93] Rey-Long Liu and Von-Wun Soo, An Empirical Study of T-hematic Knowledge Acquisition Based on Stntactic Clues and Heuristics. In Proceedings of ACL 1993.

[Martin89] Charles E. Martin, Case-Based Parsing. In C. K. Riesbeck and R. C. Schank "Inside Case-based Resoning", Lawrence Erlbaum Associates, Inc. 1989.

[Pun91] K. H. Pun, Analysis of Serial Verb Constructions in Chinese. ICCPCOL 1991, pp.170-175. 1991.

[Rau87] Lisa F. Rau, Knowledge Organization and Acess in a Conceptual Information System. Information Processing and Management. Vol.23, No.4, pp.269-283. Special Issue on Artificial Intelligence for Information Retrieval. 1987.

[Shank73] Shank, R. C., Identification of Conceptualizations Underlying Natural Language. In Computer Models of Thought and Language, ed. R. C. Shank and K. M. Colby. San Francisco: Freeman. 1973.

[Wilks75] Yorick Wilks, An Intelligent Analyzer and Understander of English. In B. J. Grosz, K. S. Jones, and B. L. Webber "Reading in Natural Language Processing". 1975.

[Yeh91] Ching-Long Yeh and Hsi-Jian Lee, Resolution of Serial Noun Constructions in Chinese. In Proc. of ROCLING IV, pp.97-110. 1991.

[Yeh92] Ching-Long Yeh and Hsi-Jian Lee, A Lexicon-Driven Analysis of Chinese Serial Verb Constructions. In Proc. of ROCLING V, pp.195-214. 1992.

[湯92] 湯廷池 (1992), 語法理論與機器翻譯: 原則參數語法, ROCLING V, pp.53-83. 1992.

[臺90a] 臺灣高雄地方法院, 臺灣高雄地方法院刑事裁判書彙編第一冊. 1990.

[臺90b] 臺灣高雄地方法院, 臺灣高雄地方法院民事裁判書彙編第一冊. 1990.