An Efficient Natural Language Processing System Specially Designed for the Chinese Language

Lin-Shan Lee*
National Taiwan University

Lee-Feng Chien†
National Taiwan University

Long-Ji Lin‡
National Taiwan University

James Huang§
Cornell University

K.-J. Chen¶
Academia Sinica

In this paper an efficient natural language processing system specially designed for the Chinese language is presented. The center of the present system is a bottom-up chart parser with head-driven operation; i.e., phrases are built up by starting with their heads and adjoining constituents to the left or right of the heads instead of strictly from left to right. In this way many more unnecessary searching actions can be effectively eliminated. The present system also includes several efficient approaches such as a direction-selective chart to simplify the control of the head-driven operation; a heuristic scheduling policy and a bidirectional look-ahead approach to eliminate many unnecessary searching actions, and an improved raise-bind mechanism combined with check rules to treat the difficult problems of movement transformations and empty categories and to simplify the design of grammar rules. The present design is based on careful consideration of some special syntactic phenomena of the Chinese language, such as head-final and head-initial structures and empty categories. A prototype of the present system has been successfully implemented and extensive experiments have been performed. In the test results significant improvement in the efficiency in processing many very complicated Chinese sentences has been observed. The detailed discussion on the various approaches, the overall system design, and the experimental results will all be presented in this paper.

1. Introduction

The use of computers to process natural languages has been the research goal of many scientists and engineers for many years, and significant improvement in technologies in recent years has brought such goal closer to reality. While substantial efforts have been made to process natural languages, especially several western languages such as English, and many powerful computational models and algorithms have been pro-

* Dept. of Electrical Engineering and Dept. of Computer Science and Information Engineering, National Taiwan University
† Dept. of Computer Science and Information Engineering, National Taiwan University
‡ Dept. of Electrical Engineering, National Taiwan University
§ Dept. of Modern Linguistics, Cornell University, NJ
¶ Institute of Information Science, Academia Sinica, Taipei, Taiwan

posed and widely used (Gazdar et al. 1987), very little work has been done with the Chinese language, which more than a quarter of the world's population use as a native language. This is probably due to the fact that the structure of the Chinese language is quite different from western languages like English and, therefore, the experience in processing western languages cannot necessarily be directly applied to the Chinese language. Jiang (1985) proposed a preliminary Chinese parsing prototype system based on the METAL system, while Lin (1985) and Lin et al. (1986a, 1986b) also developed a Chinese natural language processing system with special considerations on the phenomenon of empty categories in the Chinese language. Yang (1987) presented a method using semantic constraints to reduce ambiguity in Chinese sentence analysis. H. H. Chen et al. (1988) proposed a logic programming approach considering Chomsky's Goverment-Binding theory to cope with movement transformations in Mandarin Chinese.

In the following, some special syntactic phenomena of the Chinese language that significantly affect the design of the present system are first summarized in Sections 2 and 3, and a brief description of the present system and the structure of the linguistic knowledge base is then given in Section 4. The several new approaches, including the direction-selective chart and the head-driven chart parser, the bidirectional look-ahead approach, and the heuristic scheduling policy, are described in detail in Sections 5, 6, and 7, respectively. Sections 8 and 9 then present the improved design of the raise-bind mechanism to cope with the problem of movement transformation and empty categories. Some preliminary experimental results are discussed in Section 10, and concluding remarks and future research directions are finally given in Section 11.

2. The Head-Final/Head-Initial Structures of the Chinese Language

The Chinese language has many special syntactic phenomena substantially different from western languages. Discussions about such characteristics of the Chinese language can be found in the literature (Chao 1968; Li and Thompson 1981; Huang 1982). In this paper only some of them that have significant influence on the present study will be briefly described. They are (1) head-final/head-initial structures and (2) empty categories of the Chinese language, to be respectively summarized in this and the following sections.

The notion of the head of a phrase has a very long history, which stems from the traditional grammar and plays a central role in recent syntactic analysis frameworks such as GB and GPSG (Sells 1985). The basic idea is simply that each phrase contains a certain word that is especially important in the sense that it determines many of the syntactic properties of the entire phrase; this word is called the head of the phrase.

Most Chinese phrases and sentences are head-final, e.g., head nouns in NPs are always located at the final position. For instance, some NPs (examples 1, 2, and 3) listed in Figure 1 demonstrate this situation, where the underlines indicate the heads. Comparing these Chinese phrases with their corresponding phrases in English (shown below each Chinese phrase in parentheses), the positions of the heads in English are more free. On the other hand, other Chinese phrases that are not head-final are found to be almost always head-initial, e.g., PPs (such as example 4 in Figure 1). This is somewhat different from western languages like English. Figure 2 is a list of some fundamental phrase structure rules (PSRs) for the Chinese language used in the present system. The underlines indicate the head of each PSR. Some of the categories here are from Chao's classification (Chao 1968), and the rules here are primarily based on the

1. 玩耍 的 小孩
   playing (relativizer) children
   (the children who were playing)

2. 我 那位 住 在 美國 的 好 朋友
   I the live in America (relativizer) good friend
   (the good friend of mine who lives in America)

3. 一 位 相當 漂亮的 女孩
   a (classifier) quite pretty girl
   (a quite pretty girl)

4. 他 (向 你的 朋友們)pp 借 錢
   he from your friends borrow money
   (he borrowed money (from your friends)pp)

Figure 1
Some examples of Chinese noun phrases and preposition phrases.

(1) S = bar --> S-bar PRTAG | S PRTAG
(2) S-bar --> Topic S
(3) S --> (NP) VP
(4) NP --> (XPDE) (QP) (ADJ) N |
(QP) (XPDE) (ADJ) N |
NP LOC
(5) XPDE --> S DE | NP DE | PP DE
(6) VP --> (AUX | ADV | PP | NP)* V-bar
(7) V-bar -->V QP | V (NP) (NP | PP | VP | S | S-bar)
(8) PP --> PREP NP

Notations:

Operators:

|: Or Operaror, *:Repetition Operator, (): Optional Operator, _: head

Phrasal Categories:

S=bar, S-bar, S, NP, XPDE: an Associative Phrase or a Relative/Appositive Clause,

VP, V-bar, PP, Topic, QP: Classifier and Measure Phrase.

Lexical Categories:

PRTAG: Partical Tag, N, ADJ, ADV, AUX, LOC: Localizer, DE (的) : Relativizer

Figure 2
A list of some fundamental PSRs for the Chinese language used in the present study.

theory of Huang (Huang 1982). Apparently, the head in each of the rules is located either at the initial position (head-initial) or at the final position (head-final). Such head-final/head-initial structures will be especially useful and helpful in the present study, as will be clear later in this paper.

3. The Empty Categories of the Chinese Language

In many languages the "empty category" is typically used to refer to an empty NP position that has been vacated by a transformation called "move α" (a transformational operation introduced in government binding theory (GB) that means "move something somewhere;" i.e., the NP has been moved to a different position such that an empty position is left). Such empty categories are called "traces." They indicate the empty positions left when movements occur. There is another kind of empty category that also contains vacant NP positions, but they are not traces, because they are not derived from "move α." These empty categories are called "null pronominals." Since the distance between the location of the actual NP and its corresponding empty category may be long and the grammatic relation in such sentences can be very complicated, it is usually difficult to represent such linguistic phenomenon in simple rules. In other words, it is difficult to list all such possible movements as well as null pronominals exhaustively, and to specify all the relevant constraints explicitly in the grammar. Empty categories (or empty NPs) thus become a convenient approach usually used in linguistic theories to explain these very complicated syntactic phenomena.

In Mandarin Chinese, passivization, relativization, topicalization, ba-transformation and the use of zero pronouns play major roles in Chinese sentence structures. To deal with these syntactic phenomena, the conventional approach is to collect a set of complicated grammar rules to cover all the possibilities. However, the high complexity especially resulting from the interactions among several of these transformations make such an approach infeasible. A completely different approach is, therefore, adopted in this paper, in which a specially designed raise-bind mechanism is used based upon the theory of empty categories, as will be clear later in this paper. With such a raise-bind mechanism, it will be shown that the parser will treat all these transformations in relatively simple ways. In the following, some examples of empty categories often encountered in the Chinese language are first discussed. Consider the Chinese sentences (1)–(8) listed in the following.

1. 他 打傷 了 張三
   he hurt (aspect marker) Jang-san[1]
   (He hurt Jang-san)

2. ba-transformation:
   他 把 張三 打傷 了 e
   he ba Jang-san hurt (aspect marker)
   (He hurt Jang-san)

3. passivization:
   張三 被 他 打傷 了 e
   Jang-san by him hurt (aspect marker)
   (Jang-san was hurt by him)

1 The transliteration scheme used here is based on the Mandarin Phonetic Symbols II published in Taipei, Taiwan by the Ministry of Education of the Republic of China.

4. topicalization:

那隻狗 我 沒 看過 e
|_____|
that dog I never have seen
(I have never seen that dog)

5. relativization:

e 玩耍 的 小孩 走 了
|_____|
playing (relativizer) children go (aspect marker)
(the children who were playing are gone)

6. null pronominals:

張三 設法 [S e 逃走]
|_____|
Jang-san tried escape
(Jang-san tried to escape)

7. pivot construction:

他 叫 小孩 [S e 吃飯]
|_____|
he asked children go to dinner
(He asked the children to go to dinner)

8. zero pronoun:

張三 喜歡 e
Jang-san likes
(Jang-san likes someone or something)

Sentences (2)–(8) all involve a missing subject or object (indicated by "e"). The solid lines under sentences (2)–(7) indicate the references that each missing subject or object refers to. The missing object in sentence (8), however, does not refer to any element within the sentence. In fact, it is an omitted pronoun, which refers to someone or something understood in the situation.

According to GB theory (Chomsky 1981; Huang 1982), sentence (2) is derived from sentence (1) by a transformation called "ba-transformation." The word "把 (ba)" is a patient case marker. It indicates that the NP following it is the patient of the main verb in the sentence. The transformation is performed as follows: the object, " 張三 (Jang-san)" in (1), is moved by the carrier " 把 (ba)" to the position indicated in (2), and a trace (indicated by "e") is left behind. The trace dominates no lexical material, but is "bound" to its antecedent, " 張三 (Jang-san)." This phenomenon appears very frequently in Chinese sentences. Similar situations occur in sentences (3)–(5). In sentence (3), it is believed in the theory that the object " 張三 (Jang-san)" is moved back to the subject position and a trace is left behind to transform the sentence into a passive one. In sentence (4), the object " 那隻狗 (that dog)" can be thought of as being moved to the sentence initial position to form a topic. This is also very often seen in Chinese sentences, and is called "topicalization." In sentence (5), one explanation is that originally the relative clause " 小孩玩耍 (the children were playing)" in the sentence-initial position is used to modify the subject " 小孩 (children)," but the first " 小孩 (children)" is omitted due to repetition. This is relativization. All these sentences (2)–(5) involve a movement and a trace. In the Chinese language, ba-transformation, passivization, topicalization, and relativization all can be analyzed using the movements and the traces. The basic idea is that these phenomena are very sophisticated syntactically, but

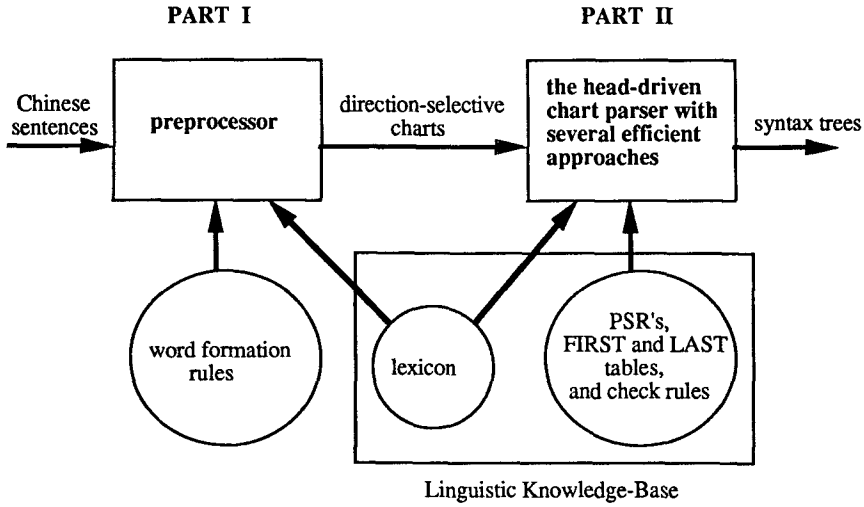


Figure 3
The block diagram of the system described in this paper.

as long as the empty NP can be inserted into the right position and the movement understood, the analysis of these phenomena will be significantly simplified, as will be shown later in this paper.

Sentences (6)–(8) are null pronominals rather than traces, because they are not derived from "move α." The notation $[\hat{s}\ldots]$ in sentences (6) and (7) denotes the presence of a clause. Null pronominals are in general free, for example, in sentence (8). But in certain constructions null pronominals are also bound, for example, in sentences (6) and (7). Sentence (7) is called a pivot construction but sentence (6) isn't; this is because in sentence (7) the object of the first verb is also the subject of the second verb, while in sentence (6) it is the subject of the first verb that is actually the subject of the second verb. Therefore in sentence (7) the null pronominal in the subject position is "bound" to the object of the first verb, but this is not the case in sentence (6). The special techniques of the raise-bind mechanism proposed in this paper to handle all such different types of empty categories will be explained in detail later in this paper.

4. The Overall System and the Linguistic Knowledge Base

Because the Chinese language has many special structures quite different from many other languages, in this paper a Chinese natural language processing system is specially designed to parse Chinese sentences more efficiently. The block diagram of the system is shown in Figure 3. The system is composed of two parts. The first part, consisting of a preprocessor and a lexicon plus word formation rules, first segments the input Chinese sentences (or a series of Chinese characters) into words by looking them up in the lexicon and applying some word formation rules. This is because, in Chinese, a word can be composed of from one to several characters without blanks on both ends to indicate the boundaries of a word; therefore, such a segmentation is necessary. Because it is impossible to collect all Chinese words into the lexicon, some word formation rules can be found to identify the words in the input sentences to help the formulation of some compound words; e.g. the determiner/measure compound words, the reduplication words, etc., such that they don't have to be stored in the

lexicon. However, because of the high degree of inherent lexical ambiguity, very often an input sentence can be segmented into several different possible word combinations and there are no simple rules to decide which combination is the correct answer. In this preprocessor, a heuristic longest word matching rule (Chen 1985) is applied to decide a most promising word combination, but errors still happen sometimes in the preprocessor and manual correction is actually needed. The preprocessor also adds relevant categorial information and other features extracted from the lexicon to each of the words. The result of the first part is represented by a data structure—a direction-selective chart (to be discussed in detail in the next section) and is transported to the second part. The second part, consisting of a parser and a linguistic knowledge base, builds up phrases on the direction-selective chart by applying the linguistic knowledge base. The parser is a head-driven chart parser, but with several special approaches developed to make the parser more efficient for the Chinese language, which will also be made clear later in this paper. The linguistic knowledge base can be broadly seen as a compilation of a four-tuple; i.e., the phrase structure rules (PSR), the FIRST and LAST parsing tables of these rules, the check rules, and the lexicon shared with the first part. If the sentence is grammatical in the sense of the grammar, a syntax tree will result as the output. Otherwise, failure will be reported. From now on, this paper will concentrate on the second part of the system, i.e., the parser and the linguistic knowledge base only, while the details of the first part can be found in other works (Ho 1984; Chen 1985). As far as the second part of the system is concerned, the input sentences are assumed to be segmented into words with categorial information and other features provided by the lexicon.

The linguistic knowledge base used in this system, as mentioned above, can be broadly seen as a compilation of a four-tuple: the phrase structure grammar (PSRs), the FIRST and LAST parsing tables for these PSRs, the check rules, and a lexicon as shown in Figure 4. The PSRs describe how sentences are built up out of phrasal categories, and how phrases are built up out of lexical categories and/or phrasal categories. All of these PSRs combined with some syntactic and semantic constraints are implemented as an ATN-like network (Woods 1970). For each probable phrasal category (constituent), the FIRST and LAST parsing tables indicate all possible lexical categories that may begin or end with the present phrasal category to guide the parser to eliminate some unnecessary searching actions in parsing, as will be described in detail in Section 6. The check rules are used in the raise-bind mechanism to handle the binding problems of empty categories and to reject illegal sentences or parsing trees, as will be described in detail in Sections 8 and 9. The lexicon is a Chinese machine dictionary, in which the allomorphs are stored together with their features and other information for syntactic and semantic analysis; e.g., category (CAT), arguments (ARG), meaning (MEA), allomorph (ALO), person, number...etc.

5. The Direction-Selective Chart and the Head-Driven Chart Parser

As discussed above, the Chinese language has prominent head-final/head-initial sentence structures, and zero pronouns are relatively freely used in Chinese sentences. Therefore, to reduce unnecessary computation in parsing Chinese sentences, a bottom-up and head-driven parsing strategy, as was used in the present study, will be more efficient than a top-down and strictly left-to-right parsing strategy. This is because a bottom-up parsing strategy can avoid inefficiency in duplicating many computations that a top-down parser often suffers from when backtracking occurs, and a head-driven parsing strategy can eliminate many unnecessary searching actions (i.e., searching actions fired by head constituents could be more promising) that often occur in a strictly

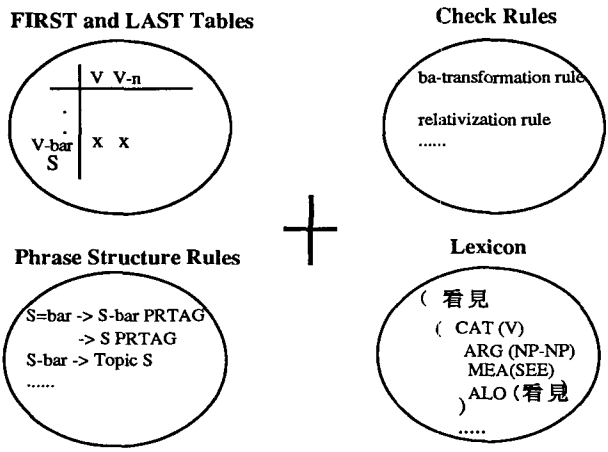


Figure 4
The linguistic knowledge base.

left-to-right parsing scheme. This will all become clearer later in this paper. Several approaches were further developed in the present parser described in this paper to better realize this concept, so that significant improvement as compared to some previous Chinese natural language processing systems (Yang 1987; Jiang 1985; H. H. Chen et al. 1988) can be observed. In the following sections, these approaches, including the direction-selective chart, the bidirectional look-ahead approach, the heuristic scheduling policy, and the raise-bind mechanism will be described in detail. Here, we first describe the direction-selective chart in this section.

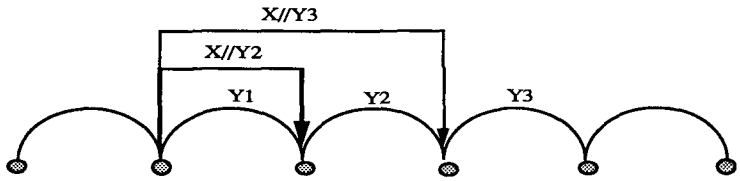
Before parsing is performed, any input word sequence has to be first represented by the direction-selective chart. Just like a conventional chart (Kay 1980; Winograd 1983), the direction-selective chart is an efficient data structure to record what has been done so far in the course of parsing to avoid duplicate computation. The special feature of the direction-selective chart is that the active edges (the incomplete constituents that need other complete constituents to their left or right to compose larger ones) are further partitioned into two disjoint groups: forward-active (F-active) and backward-active (B-active) edges to indicate different search directions as described below.

In the head-driven parser, the parsing process will begin on the heads in the input word sequence. As described in Section 2, the heads in the Chinese language are at either the initial or final position of a phrase; therefore, in a head-driven parser, the searching actions triggered by an initial head (being a complete constituent) are always looking forward (from left to right); while the actions triggered by a final head are always looking backward (from right to left). However, no bidirectional searching actions can be triggered by a single head in the course of parsing. Therefore, in this head-driven chart parser, the F-active edges are used to denote forward searching actions, and the B-active edges are used to denote the backward. The information specified on each active edge then consists of the search direction (forward or backward), in addition to normal information, such as the vertices where the edge starts, and ends, the grammar rule referred to, etc.

Two diagrams depicted in Figure 5 show the two different searching actions. Figure 5a is the forward search and Figure 5b the backward, in which each arc represents an inactive edge (a complete constituent) and each arrow line represents an active edge. The labels attached above the inactive edges denote the corresponding categories.

(a) The searching actions triggered by an initial head are always looking forward (left-to-right).

The sample grammar rule: X -> Y1 ... Yn



(b) The searching actions triggered by a final head are always looking backward (right-to-left).

The sample grammar rule: X -> Y1 ... Yn

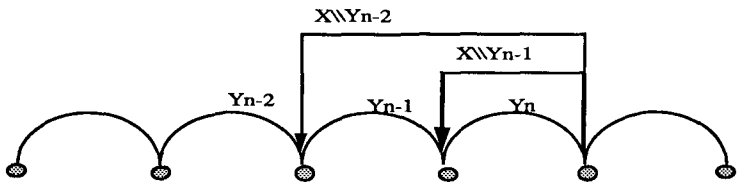


Figure 5
The searching actions in the direction-selective chart.

According to the sample grammar rules listed in the figure, the arrow points out the search direction, and a label attached above with a form X//Y indicates that it needs a right neighboring complete constituent with Y category to form an X constituent; a label with a form X\\Y indicates that it needs a left neighboring complete constituent with Y category to form an X constituent.

To compare with a similar approach, in Stock's island-driven bidirectional chart (Stock et al. 1988), the searching actions are triggered by islands (an island is a more reliable word hypothesis resulting from speech recognition) and the searching directions may be bidirectional; i.e., an active edge may search for constituents on both sides as shown in Figure 6. Also, Pareschi and Steedman (1987) had proposed another similar bidirectional chart parsing algorithm to handle operations such as functional composition for categorial grammars applications (Steedman 1985). However, in our parser, the actions triggered by the heads have directions either strictly forward or strictly backward, obviously resulting from the head-final/head-initial phenomena of the Chinese language. This makes the control of our parser much simpler and more efficient in the present problem.

6. The Bidirectional Look-Ahead Approach

Since Chinese is believed to be a syntactically ambiguous language with relatively free word order, many complicated syntactic phenomena derived from such situation will thus make it difficult for a parser to work on Chinese sentences deterministically, as was done in Marcus's famous work for English (1982). For example, in long-distance movements the distance between the location of the binding NP and its corresponding empty category may be long, and the grammatical relation in such a sentence can be

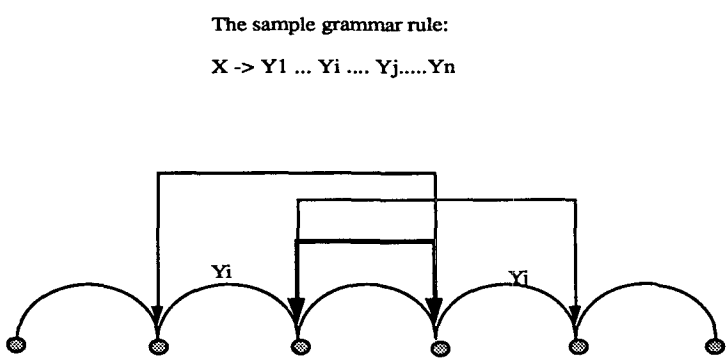


Figure 6
The searching action with Stock's bidirectional chart parser.

very complicated. It is also very often difficult for a parser to deal efficiently with the binding of the empty category deterministically. This is why substantial redundant computation efforts usually occur unavoidably in analyzing Chinese natural language. However, some of this redundant computation can be avoided in the present parser, by the approach discussed below.

An active edge built on a chart indicates a stage in the search for a constituent. It records the category of the constituent it is looking for, where the constituent should be, and the structure obtained so far in order to form a complete one. During the course of parsing because the parser usually cannot correctly predict the category and position of the constituents to be built next, many unnecessary and redundant active edges will inevitably be built and substantial searching efforts thus have to be wasted, as very often happened in many chart-based parsers. This is very inefficient and can, in fact, be significantly improved in the present system based on the following concept. Because no phrasal category is null in the grammar rules, whenever an active edge is built into the direction-selective chart, the parser can first examine the constituent, exactly located at the position the active edge is looking for, to check whether the desired category can begin with (if the active edge is F-active for an initial head) or end with (if the active edge is B-active for a final head) the category of the examined constituent, according to the description of the grammar. In this way it is possible for the parser of the present system to avoid building many unnecessary active edges by such a "bidirectional look-ahead approach" combining the special head-driven strategy developed in the present study with the concept of FIRST and LAST parsing tables (Aho and Ullman 1972) to be discussed in detail below. In the following, we shall first define these two tables and then describe how the bidirectional look-ahead approach works.

FIRST(C): If C is a category, FIRST(C) is the set of all possible lexical categories the category C can begin with. Meanwhile, a matrix tabulating such FIRST relations of all categories of a grammar is called the FIRST parsing table of the grammar. For example, Figure 7 is the FIRST parsing table for the sample grammar rules listed in Figure 8. For instance, in it FIRST(NP) = {PRON, N} because FIRST(NP) = PRON ∪ N ∪ FIRST(XPDE) = PRON ∪ N ∪ FIRST(S) = PRON ∪ N = {PRON, N}.

LAST(C): If C is a category, LAST(C) is the set of all possible lexical categories the category C can end with. Meanwhile, a matrix tabulating such LAST relations of all categories of a grammar is called the LAST parsing table of the grammar. For example, Figure 9 shows the LAST parsing table for the sample grammar rules listed in Figure 8.

	V-	V-n	PRON	N	DE	ADV
NP			X	X		
XPDE			X	X		
VP	X	X				X
V-bar	X	X				
S			X	X		
V-	X					
V-n		X				
PRON			X			
N				X		
DE					X	
ADV						X

Figure 7
The FIRST parsing table for the sample grammar shown in Figure 7, where each entry filled by an "X" indicates that the category (constituent) for the row may begin with the lexical category for the column.

(1) NP --> PRON | (XPDE) N
(2) XPDE --> S DE | NP DE
(3) VP --> (ADV)* V-bar
(4) V-bar --> V- | V-n NP
(5) S --> NP VP

Figure 8
A set of sample grammar rules to show the construction of the FIRST and LAST parsing tables.

	V-	V-n	PRON	N	DE	ADV
NP			X	X		
XPDE					X	
VP	X		X	X		
V-bar	X		X	X		
S	X		X	X		
V-	X					
V-n		X				
PRON			X			
N				X		
DE					X	
ADV						X

Figure 9
The LAST parsing table for the sample grammar shown in Figure 7, where each entry filled by an "X" indicates that the category (constituent) for the row may end with the lexical category for the column.

For instance, in it LAST(VP) = LAST(V-bar) = V- ∪ LAST(NP) = V- ∪ PRON ∪ N = {V-, PRON, N}.

In the present parser, both of these parsing tables for the Chinese grammar used have been constructed (no phrasal category is null). During parsing, when an F-active edge is waiting to be constructed (using X//Y as in Figure 5a to indicate its searching action), the parser will first look up the FIRST table to see whether the word, at the position it is looking for, has a lexical category belonging to FIRST(Y). If it does, then the active edge can be built; otherwise the active edge is redundant, because no such required constituent can be constructed. Similarly, when a B-active edge is waiting to

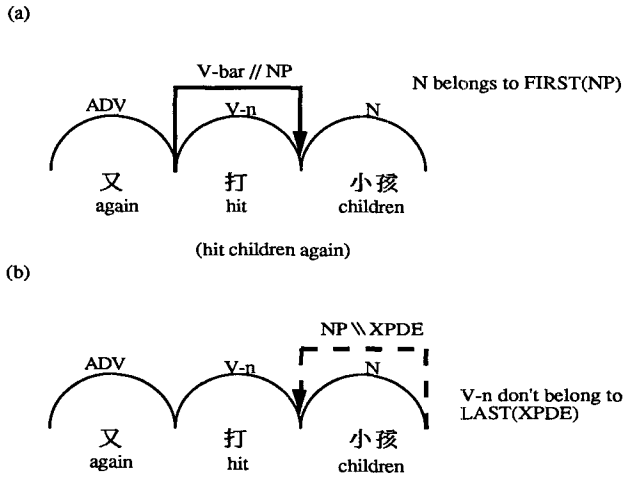


Figure 10
An example to illustrate the use of FIRST and LAST parsing tables to avoid building many unnecessary active edges on the direction-selective chart.

be constructed (using X\\Y to indicate its searching action), the parser will first look up the LAST table to see whether the word, at the position it is looking for, has a lexical category belonging to LAST(Y). If it does, then the active edge can be built; otherwise the active edge is redundant, because no such required constituent can be constructed.

For example, consider parsing a Chinese phrase, illustrated in Figure 10, with the above sample grammar and parsing tables in Figures 7–9. In Figure 10a, an F-active edge (V-bar//NP) is triggered by the initial head "打 (hit)." This indicates that a right neighboring NP constituent is needed to form a complete V-bar constituent. Fortunately, the right neighboring word " 小孩 (children)," exactly has an N category belonging to FIRST(NP); therefore, the edge can be constructed. On the other hand, in Figure 10b, a B-active edge NP\\XPDE is triggered by the final head " 小孩 (children)." This indicates that a left neighboring XPDE constituent is needed to form a complete NP constituent. However, in this case the left neighboring word "打 (hit)" doesn't have a category belonging to LAST(XPDE); therefore, the edge will not be built, because it is apparently redundant. In this way, the bidirectional look-ahead approach can, in fact, eliminate many unnecessary searching actions (or active edges) and make the parsing process more efficient. A similar approach can be found in Tomita's extended LR parser (Tomita 1986), in which the parsing table used is an extended LR parsing table, and the parsing process performed is strictly left-to-right, as compared to the two different parsing tables and two parsing directions in the present system.

7. The Heuristic Scheduling Policy

Each step in the parsing process can very often produce more than one subsequent steps. For example, a new edge built into a chart may cause an arbitrary number of edges (candidate constituents) to be built. Usually, in such situations some of them

should be processed prior to the others instead of simply performing exhaustive processing. In other words, a well defined scheduling policy is, in fact, helpful. This is why most of the chart parsers have an agenda to schedule these steps (Kay 1980). In the present system, a heuristic scheduling policy is also developed, as described in this section.

In the present system the scheduling policy is primarily based on some heuristic estimation obtained from empirical experiences, in which each candidate constituent is assigned a priority to indicate processing order. Most of the time, the assignment is described by its category. For example, a constituent with an S category will be constructed prior to a constituent with a VP category (some unnecessary VP constituents may be therefore eliminated, for example), a constituent with a VP category will be constructed prior to one with a V-bar category... etc. On the other hand, if more than one candidate constituents have the same priority, the constituent at the right-most position (located at the farthest end vertex) is then the first to be built.

To see how the above heuristic scheduling policy is integrated with the head-driven chart parser discussed here to efficiently parse an input sentence, a simple example is used in the following to show the parsing process. Suppose the input sentence is:

你	的	哥哥	又	打	小孩
you	of	brother	again	hit	children

(your brother hits children again)

and the grammar rules, the FIRST and LAST tables used are those shown in Figures 7–9, respectively. The resulting chart is shown in Figure 11a, where the numbers attached on the edges indicate their order in the course of parsing. In fact, it is easy to see that many constructions have been successfully avoided in the chart. To make the illustration simple and clear, here we shall follow the parser to analyze only the sentence fragment " 又 打 小孩 " (hit children again) as shown in Figure 11b.

Before the parsing process starts, three inactive edges are constructed in the chart to represent the sentence fragment. Then, based on the head-driven principle and the sample grammar, the word "打 (hit)," according to rule (4) in the sample grammar, is an initial head (a transitive verb) that needs a right neighboring NP to form a V-bar (that is represented by an F-active edge; i.e., edge(1) in Figure 11b); the word " 小孩 (children)," based on rule (1), is a final head (a noun) that either can be an NP by itself (this is represented by an inactive edge; i.e., edge(2) in Figure 11b) or needs a left neighboring XPDE to form an NP (this is represented by a B-active edge(*) in Figure 11b). Examining each of these three edges with either the FIRST or LAST tables, as illustrated in Figure 10b previously, edge(*) should not be built (it is a redundant edge) but edges (1) and (2) both can be potential candidates and, thus, should be built. Now, according to the heuristic scheduling policy, a V-bar edge will be built prior to an NP edge; therefore, edge (1) is the first edge to be built. However, since there is no such NP currently in the chart, no new edges can be produced after edge (1) is added, and thus edge (2), the only candidate, is then added to the chart. This edge now satisfies the request of edge (1) and, therefore, creates a V-bar inactive edge (edge (3)) as a new candidate. Meanwhile, since edge (2) isn't a head, no active edges can be triggered, so that edge (3) is the third edge to be built. Similarly, a VP (edge (4)) can then be triggered by edge (3), and, finally, a complete VP constituent (edge(5)) can be built.

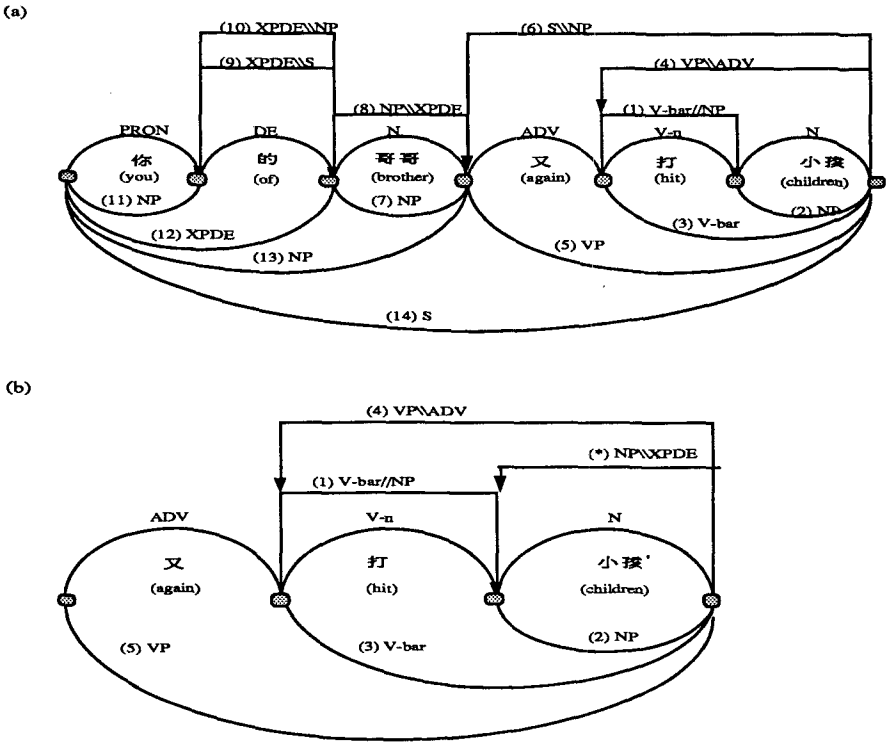


Figure 11
An example to demonstrate the parsing process.

8. The Raise-Bind Mechanism and Check Rules

The raise-bind mechanism presented here is specially developed in the present system to treat the difficult problems of movement transformations and empty categories so that the design of the grammar can be simplified. It is used to cope with the empty categories; in other words, to find the antecedent for each empty category except for those that are free (such as in sentence (8) in Section 3). During the parsing process when an NP is desired, the parser, with the aid of the raise-bind mechanism, will perform the following operations.

First, a corresponding active edge indicating the request for an NP may be built in the chart after looking up the FIRST or LAST tables. This request can be satisfied when a desired NP is actually encountered. Second, if the desired NP is not encountered and the NP position can, instead, be filled by an empty category (according to the check rules with details explained below), then an empty NP will be generated to fill the vacant position and a new edge (active or inactive) denoting this satisfaction will be built in the chart. This empty NP will then be raised up in some way along the parsing tree (implicitly represented in the chart) when the tree is growing up (recall that the parser works bottom up), until its antecedent is parsed. At this point, the parser binds the empty NP by setting it to refer to its antecedent (this is also guided by check rules as described below). Once bound, the empty NP will not be raised up any further, because an empty NP has exactly one antecedent and cannot be bound more than once.

Not every NP position can be filled by an empty category. In the Chinese language,

empty categories only appear in the subject position and direct object position, never in the indirect object position, or the prepositional object position. In our implementation, an empty NP contains three fields: (1) a field to keep the pointer to indicate its antecendent, (2) a field to keep where it came from, and (3) a field to keep the syntactic or semantic constraints on the empty NP for later checking. Rules for this kind of checking are called check rules in the present system. Most of the time, these check rules are invoked when a constituent containing unbound empty categories is built in the chart. Usually, distinct rules are used to treat different problems. For example, we can informally state the rules to treat the relativization phenomena as follows: for a noun and a relative clause to be combined into an NP, the relative clause must contain an empty NP that is unbound and marked to be coming from either the subject position or the object position of the relative clause, and then this empty NP will be bound to the (head) noun (just as in sentence (5) in Section 3; a further example will be given below). We can also state the check rules for passivization as follows: once a clause is constructed, the parser checks whether the prepositional phrase, "被 + NP" (similar to "by + NP" in English) is involved in the clause. If so, there must be an empty NP that is unbound and marked to be coming from some object position, and this empty NP will be bound to the subject of the clause (just as in sentence (3) in Section 3; a further example will be given below). The check rules for pivot construction can also be formulated as follows: in a pivot construction, the direct object will bind the empty NP coming from the subject position of the embedded clauses (just as in sentence (7) in Section 3; a further example will be given below). Apparently, check rules for other linguistic phenomena such as topicalization, ba-transformation, and so on can all be similarly developed. In fact, the binding process in the raise-bind mechanism here is rule-based rather than principle-based; that is, the whole binding process in the raise-bind mechanism is determined by the check rules and the phrase structure rules, while instead in some other principle-based parsers, for example, a parser completely based on GB theory (Wehrli 1988), it is influenced by some linguistic principles; e.g. the government binding principle in GB theory. However, the rule-based approaches may take some more cost in computation than the principle-based approaches, but in dealing with some specific problems the former approaches seem more flexible than the latter approaches. This is why in sentence (7) (pivot construction) the empty category in the subject position of the embedded clause can be bound to the NP "小孩 (children)" in the higher clause, even if it is not governed by the NP. To illustrate the operation of the above check rules, let's consider phrase (9) in the following and its parsing tree in Figure 12, in which several of such phenomena interact with one another. It will be shown that, with the present approach, this complicated problem can be solved easily.

9. 被 李四 叫 去吃饭 的 小孩
   by Li-s ask go to dinner relativizer children
   (the children who were asked by Li-s to go to dinner)

Let's follow the bottom-up parser to parse phrase (9): (1) Node S1 (a clause constituent) is constructed and e1 serves as the dummy subject (an NP). (2) Node V-bar is constructed and the dummy object e2 is inserted. Because of the empty category e1 existing in the embedded clause S_1, the check rules are invoked. According to the check rules for pivot construction, e1 is bound to e2. (3) Node S2 is constructed with an empty NP e3. S2 is a passive clause because of the PP, "by Li-s." According to the check rules for passivization, e3 binds e2. (4) Node NP is constructed. According to the check rules for relativization, e3 is bound to "children." Notice that only e3 was raised up across the node S2, because e1 and e2 had been bound before S2 was constructed.

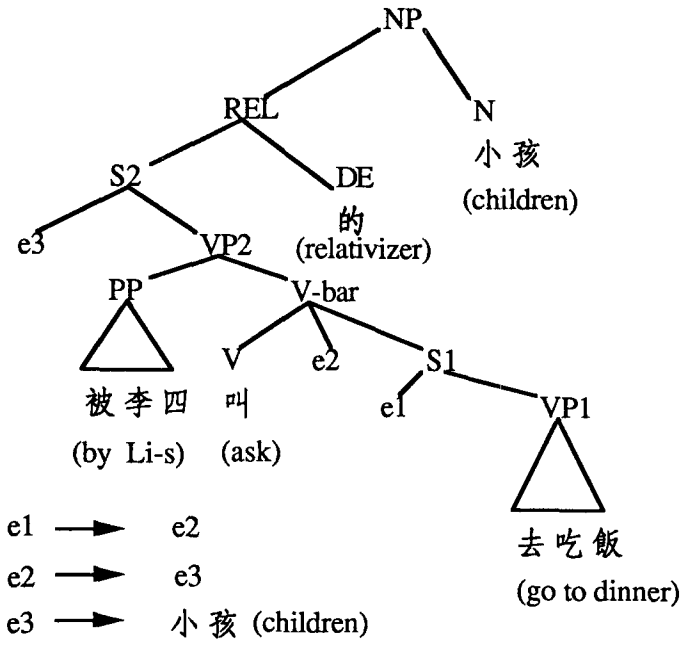


Figure 12
The parsing tree of the example (9).

Once the parsing tree in Figure 12 is completed, it is easy to answer questions such as who were asked and who went to dinner. Since e1 is the dummy subject of "go to dinner" and the binder of e1 is e2, whose binder is e3, whose binder is "children," we can conclude it is "children" who went to dinner. In the same way, we can also conclude it is "children" who were asked.

The raise-bind mechanism also serves as a filter to rule out incorrect sentences or incorrect parsing trees. For example, if no empty NP is raised up or no NP is bound within a construction involving passivization or relativization, such a construction will be rejected by the check rules. On the other hand, some unbound NP could have no antecedent. This can be determined from the check rules by looking up the attributes of its corresponding verb. If the corresponding argument of the verb can actually be omitted, then the parsing tree can be accepted; otherwise, the parsing tree will be ruled out. Of course, if this mechanism is adopted for English sentence analysis, a test must be performed to rule out sentences with other empty categories that have no binder. But such sentences are, in general, grammatical in Chinese (just as sentence (8) in Section 3).

9. Further Discussion of the Raise-Bind Mechanism

Relativization in Chinese is a long-distance movement; that is, it can sometimes move an object across several S (sentence) nodes. The noun phrase in (10) below shows an example. On the other hand, the noun phrase in (11) is ambiguous. If the head noun ("the man") binds e1, this NP means "the man whom someone likes." If the head noun binds e2, on the other hand, it means "the man who likes someone or something." To remove the ambiguity, semantic interactions are needed.

10. [S 我 叫 李四 [S 幫 我 [S 買 e]]] 的 書
I ask Li-s help me buy (relativizer) book
(the book which I asked Li-s to help me buy)

11. [S e2 喜歡 e1] 的 人
like (relativizer) the man

Considering the above situations, we can further improve the check rules as follows: for a noun and a relative clause to be combined into an NP, the parser checks the "empty-NP list" raised up from the relative clause, and

- if no empty NP is raised up, rule out the NP constituent;
- if an empty NP is raised up and marked to be coming from the subject position or object position or embedded object position (as in example (10)), set the empty NP to be bound to the head noun;
- if two empty NPs are raised up from both the subject and object positions (as in example (11)), employ semantic analysis to determine the proper binding (the present system is syntactically-based therefore such semantic analysis will be considered in the next phase research).

Like relativization, topicalization is also a long-distance movement and can be further improved in a similar way.

Another syntactic phenomena crucial to the parser is known as the complex NP Constraint (CNPC) (Radford 1981); i.e., no transformation rule can move any element out of a complex NP, where a complex NP (CNP) is an NP containing a relative clause. This CNPC can be easily encoded in the grammar in the present approach by a simple rule; i.e., no empty NPs can be raised up across a CNP node. Hence, it is impossible for the empty NP within a CNP to be bound to any element out of that CNP.

In most cases, ba-transformation and passivization move the direct objects of verbs. But the phenomenon known as "subject-to-object raising" (Radford 1981) has some differences. In such a case, the subject of an embedded clause can be moved into the subject (or ba-object) position of the higher clause by passivization (or ba-transformation). For example, sentence (13) is derived from sentence (12) by such a movement.

12. 大家 將來 會 認爲 這個錯誤 是 對的
people future will believe this mistake is correct
(People will believe in the future that this mistake is correct.)

13. 這個錯誤 將來 會 被 大家 認爲 e 是 對的
this mistake future will by people believe is correct

(This mistake will be believed to be correct by the people in the future.)

To cope with such subject-to-object raising, the rules described in the previous section for passivization can be modified as follows. The subject of a passive clause will bind the empty NP in either the object position or the subject position of an embedded clause.

The raise-bind mechanism is a computational approach to deal with the binding of empty categories. Its most attractive feature is, in fact, that it is specially designed

to be used for head-driven strategy as in the present system. In ATN (Bates 1978), the hold-list mechanism is used for a similar purpose. However, it is not very helpful in parsing Chinese because: (a) it does not really fit the head-driven operation; (b) it cannot deal with really unbound empty categories (e.g. example (8)); (c) it handles left extraposition (e.g. (2)–(4)), but not right extraposition (e.g. sentence (5)). A movement is called left (or right) extraposition, if it moves an NP to the position left (or right) of its trace. To deal with right extraposition, ATN uses another mechanisim.

In GB theory, both left extraposition and right extraposition move an NP to a position governing its trace; a null pronominal, if bound, is always bound to an NP governing the null pronominal (Chomsky 1981). So, the raise-bind mechanism combined with the check rules discussed here is sufficient to cope with all empty categories, left or right extrapositions, and traces or null pronominals, since its function is simply to raise up an empty category to be bound by an NP that governs this empty category. We believe that by means of the raise-bind mechanism it will not be difficult to implement some similar linguistic operators such as the slash concept of GPSG (Sells 1985). However, this present approach still has some constraints; for example, the check rules may take some computation cost, and multiple binding may occur if some of the check rules are not consistent in some situations.

10. Preliminary Experimental Results

In order to see how the present approaches work as compared to conventional approaches in parsing Chinese sentences, an experimental system was implemented and extensive experiments have been performed. The system is written in C language and performed on an IBM PC/AT. A total of 47 phrase structure rules were used, in which 19 rules are backward with a final head (indicated by a backward flag) and 28 rules are forward with an initial head (indicated by a forward flag). All these rules, together with the corresponding FIRST and LAST tables are listed in APPENDIX A and APPENDIX B, respectively. A total of four tests were performed for each test sentence. In test I, a conventional left-to-right parsing strategy without any look-ahead capability was used, while in Test II the left-to-right parsing strategy was equipped with a forward look-ahead approach (with the FIRST table only). In Test III, the present head-driven parsing strategy based on the direction-selective chart was used without any look-ahead capability, and finally in Test IV, the present head-driven parsing strategy based on the direction-selective chart equipped with the bidirectional look-ahead approach (with both the FIRST and LAST tables) was used. Because of the flexibility of the present direction-selective chart, all the above four tests can be easily implemented in the present system. For example, to perform Test I all one has to do is simply switch all the flags in the phrase structure rules into the forward mode, etc.

After every test sentence was parsed in each test and an output syntax tree was obtained, the total number of constituents constructed in the process of parsing was recorded. In Figure 13, the total number of constituents constructed in each of the four tests, together with the number of resulting output parsing trees for 25 typical sentence examples picked up from a total of 200 test sentences, are listed. These 25 typical sentence examples are also listed in APPENDIX C . In Figure 14, the reduction ratios of edge construction for the four tests as compared to Test I (i.e., the ratio of the number of edge constructions to that of Test I for the 25 sentence examples), together with the average reduction ratios for all the 200 test sentences, are shown. Also listed in the last row of Figure 14 is the average time necessary to process a sentence on the IBM PC/AT for each test. It can be found that, as compared to the

conventional approach in Test I, on average the total number of necessary constituents constructed for a sentence is reduced by a factor of 0.635 (or less than 2/3), or the required processing time is reduced from 8.2 sec to 3.1 sec for a sentence, through the use of the present direction-selective chart, the head-driven parsing strategy, and the bidirectional look-ahead approaches (Test IV). Also, if the direction-selective and the head-driven parsing strategy are used alone without look-ahead capability (Test III), an edge reduction ratio of 0.762 can be achieved and it takes about 6.4 sec on average to parse a sentence, which is close to the edge reduction ratio and processing speed (0.758 and 6.2 sec/sentence) for the use of the FIRST table only with conventional left-to-right parsing (Test II). Moreover, another important observation is that the edge reduction ratios are more prominent or the present approaches become more efficient when the test sentence has a higher degree of ambiguity (a larger number of parsing trees were obtained). Test sentences 20 and 22 in Figures 13 and 14 are good examples. In any case, these results have shown that the present approaches of direction-selective chart, head-driven parsing, and bidirectional look-ahead can, in fact, eliminate many unnecessary searching actions (or active edges) and can make the parsing process much more efficient than conventional left-to-right parsing strategies in parsing Chinese sentences.

On the other hand, in order to show the capabilities of the present parser especially with the raise-bind mechanism to handle the difficult problem of empty categories, parsing results of several typical sentence examples having empty categories are included in APPENDIX D in bracketed text form. These results show that the raise-bind mechanism, combined with the check rules proposed here, can certainly treat sophisticated syntactic problems of empty categories and movement transformations and simplify the design of grammar rules.

Although the lexicon implemented on the present system is relatively small (including 1,120 words) compared with dictionaries for practical applications (it is estimated that at least 80,000 words are necessary), the capabilities of this system are clearly demonstrated. It was estimated in the tests that, as a general-purpose system without any special problem domain, about 80% to 85% of the sentences in the high-school textbooks of Taiwan, can be successfully analyzed by this system, provided that all the necessary words are either already in or can be keyed into the lexicon before analysis or parsing is performed. This estimate was obtained simply because in the tests in fact a total of 241 sentences randomly selected from these textbooks were tested and correct parsing trees were obtained for 200 of them. These 200 sentences are therefore used in all the above discussions.

11. Concluding Remarks and Future Research Directions

In this paper an efficient natural language processing system specially designed for the Chinese language is presented. The present design is the result of careful consideration of some of the special syntactic phenomena of the Chinese language; for example, head-final and head-initial structures, empty categories, and movement transformations. The present system is an attractive integration of several novel approaches; e.g., the head-driven parsing strategy, the direction-selective chart, the bidirectional look-ahead approach, the heuristic scheduling policy, and the raise-bind mechanism based on check rules, etc. The head-driven parsing strategy can eliminate unnecessary searching actions, and the direction-selective chart simplifies the control of the head-driven parsing strategy and makes the parser more flexible in performing many different parsing strategies. The heuristic scheduling policy and the bidirectional look-ahead approach can, in fact, further significantly reduce the large number of searching

Test sentence	Number of edge constructions				Number of parsing trees
	Test I	Test II	Test III	Test IV	
1	120	87	97	81	1
2	226	171	161	135	1
3	199	158	162	135	1
4	266	203	189	155	2
5	234	172	189	156	2
6	113	91	89	81	1
7	275	209	204	173	3
8	132	106	107	89	1
9	140	112	113	99	1
10	144	109	126	113	3
11	107	88	85	78	1
12	209	158	178	161	3
13	444	315	287	260	4
14	277	199	195	175	1
15	743	493	592	534	12
16	358	256	293	247	3
17	334	234	213	190	2
18	115	86	97	82	1
19	177	140	143	129	2
20	1540	1112	1034	949	33
21	276	201	207	181	2
22	2583	1801	1528	1327	27
23	114	95	103	88	1
24	310	253	209	190	3
25	491	376	352	317	9

Figure 13
A table listing the total number of constituents constructed in the four tests and the number of parsing trees obtained for the 25 typical sentence examples listed in APPENDIX C.

actions and make the parsing processes more efficient, while the raise-bind mechanism and the check rules can certainly handle sophisticated problems of movement transformations and empty categories, and can simplify the design of grammar rules. Although much more work is still in progress to further improve the present system, this is definitely a very good initial attempt to efficiently process natural sentences of the Chinese language, the structure of which is significantly different from most western languages, such as English.

Although the present system has shown satisfactory initial results, some natural difficulties for the Chinese language still remain, such that significant improvement over the present system is highly desired. One of the primary difficulties is due to the lack of inflections in Chinese words. This gives multiple solutions in word category identification and causes exponential growth in the number of structures. It is, therefore, believed that an integrated syntactic and semantic analysis will eventually become an inevitable solution in the future. Because verbs are, ultimately, heads of sentences, appropriate classification of verbs may help in determining syntactic structure and, thus in grasping the semantic meaning of sentences. Although in most linguistic theories verbs are classified according to syntactic properties, and some linguistic theories, such as Lexical Functional Grammar (Sells 1985) and Case Grammar (Fillmore 1968), also provide mechanisms to explicitly represent functional or semantic role assignment of constituents, such work for the Chinese language is still relatively preliminary. Chao (1968) has only distinguished intransitive verbs from transitive verbs and Yang (1987) has made very encouraging initial efforts by classifying Chinese verbs according to

Test Sentences	Edge Reduction Ratios			
	Test I	Test II	Test III	Test IV
1	1.000	0.725	0.809	0.675
2	1.000	0.757	0.718	0.598
3	1.000	0.794	0.815	0.679
4	1.000	0.764	0.711	0.583
5	1.000	0.736	0.808	0.667
6	1.000	0.806	0.788	0.717
7	1.000	0.760	0.742	0.630
8	1.000	0.804	0.811	0.675
9	1.000	0.800	0.808	0.718
10	1.000	0.757	0.875	0.785
11	1.000	0.823	0.795	0.729
12	1.000	0.756	0.852	0.771
13	1.000	0.710	0.647	0.586
14	1.000	0.719	0.704	0.632
15	1.000	0.664	0.797	0.719
16	1.000	0.716	0.819	0.690
17	1.000	0.701	0.638	0.569
18	1.000	0.748	0.844	0.714
19	1.000	0.791	0.808	0.729
20	1.000	0.723	0.672	0.617
21	1.000	0.729	0.750	0.656
22	1.000	0.698	0.592	0.514
23	1.000	0.834	0.904	0.772
24	1.000	0.817	0.675	0.613
25	1.000	0.766	0.717	0.646
Average Ratios	1.000	0.758	0.762	0.635
Average Speed of Processing (Sec/Sentence)	8.2	6.2	6.4	3.1

Figure 14
A table showing the edge reduction ratios for the 25 typical sentence examples listed in APPENDIX C and the average reduction ratios for all the 200 test sentences for the four tests campared to Test I.

their transitivity into eight different syntactic classes and giving each verb a semantic category that can be used to decide the case frame to solve the problem of serial verb construction. Recently, a new classification scheme for Chinese verbs has been developed (K.-J. Chen et al. 1988), in which current theories of feature-based categorization are adopted. This scheme is based on the results of analyzing 16,824 Chinese verbs, with careful consideration given to difficulties in parsing Chinese sentences. In the next stage of the present study, this verb classification will be employed, and much more syntactic and semantic information will be provided by the lexicon, especially for verbs, and represented as complex feature structures (Gazdar 1988). Furthermore, several other approaches will also be included in the next stage of the present study, such as the unification concept (Sheiber 1986), and the slot and filler principle (Hellwing 1988). In other words, although there is still a very long way to go before a really convenient and efficient natural language processing system for Chinese becomes available in the future, the present system apparently serves as a successful initial step on the way.

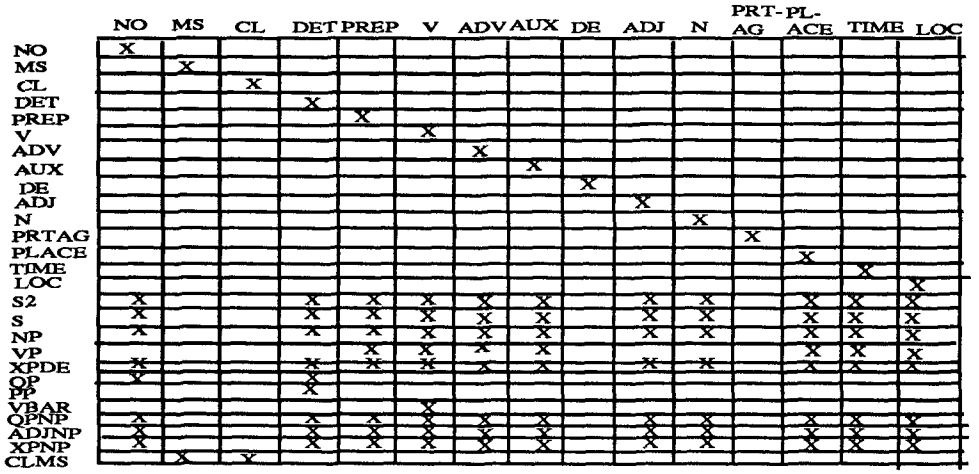
Appendix A

The phrase structure rules used in the experiments

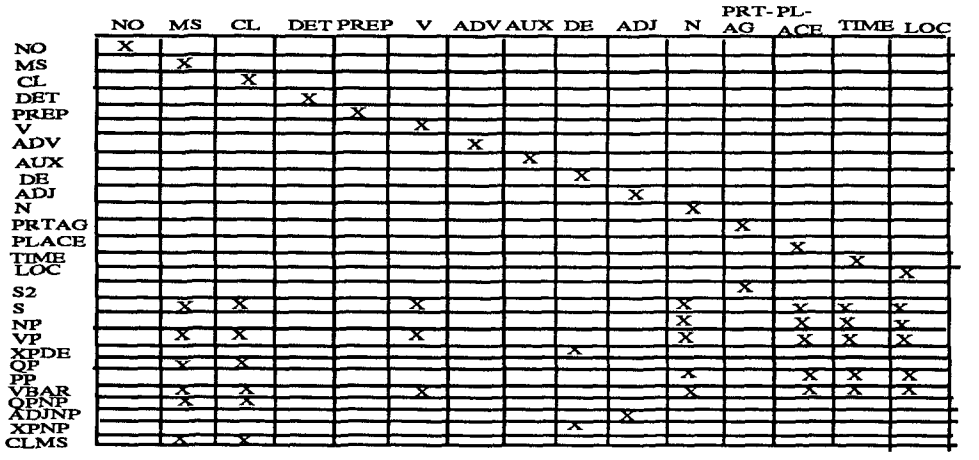
1.	b	S2	→	S PRTAG
2.	b	S	→	NP VP
3.	f	S	→	VP
4.	f	NP	→	N
5.	f	NP	→	PLACE
6.	f	NP	→	TIME
7.	b	NP	→	NP LOC
8.	f	NP	→	LOC
9.	b	NP	→	PreN1 N
10.	b	NP	→	PreN2 N
11.	b	NP	→	PreN3 N
12.	f	PreN2	→	QP
13.	b	PreN2	→	XPDE QP
14.	b	PreN1	→	XPDE ADJ
15.	b	PreN1	→	QP ADJ
16.	b	PreN1	→	ADJ
17.	b	PreN1	→	XPDE QP ADJ
18.	b	PreN1	→	QP XPDE ADJ
19.	b	PreN3	→	QP XPDE
20.	f	PreN3	→	XPDE
21.	b	XPDE	→	S DE
22.	b	XPDE	→	NP DE
23.	b	XPDE	→	PP DE
24.	f	VP	→	AUX VBAR
25.	f	VP	→	AUX ADV VBAR
26.	f	VP	→	ADV AUX VBAR
27.	f	VP	→	ADV AUX PP VBAR
28.	f	VP	→	AUX ADV PP VBAR
29.	f	VP	→	ADV VBAR
30.	f	VP	→	PP VBAR
31.	f	VP	→	VBAR
32.	f	VBAR	→	V
33.	f	VBAR	→	V QP
34.	f	VBAR	→	V PP
35.	f	VBAR	→	V VP
36.	f	VBAR	→	V S
37.	f	VBAR	→	V NP NP
38.	f	VBAR	→	V NP PP
39.	f	VBAR	→	V NP VP
40.	f	VBAR	→	V NP
41.	f	VBAR	→	V NP
42.	f	PP	→	PREP NP
43.	b	QP	→	DET NO CLMS
44.	b	QP	→	NO CLMS
45.	b	QP	→	DET CLMS
46.	f	CLMS	→	CL
47.	f	CLMS	→	MS

APPENDIX B. The FIRST and LAST tables for the sample grammar shown in APPENDIX A

(a) The FIRST table for the sample grammar shown in APPENDIX A



(b) The LAST table for the sample grammar shown in APPENDIX A



APPENDIX C. 25 example sentences to show typical experimental results

1. 我的 孩子 已經 上學 去了
   my (child) has go to school (aspect)
   (My child has just gone to school.)

2. 你的 哥哥 又 打 我的 小孩
   your brother again hit my child
   (Your brother hit my child again.)

3. 這 是 一 架 會 聽 國語 的 電腦
   this is a (classifier) can listen mandrain (relativizer) computer
   (This is a computer which can listen to Mandarin.)

4. 他 送 我 一 朵 我 很 喜歡 的 花
   he give I a (classifier) I very like (relativizer) flower
   (He gave me a flower which I liked very much.)

5. 我 喜歡 我的 畫 掛 在 牆 上
   I like my painting hang on wall (localizer)
   (I like my painting to be hung on the wall.)

6. 一 粒 種子 睡 在 泥土 裡
   a (classifier) seed sleep in earth (localizer)
   (A seed is planted in the earth.)

7. 我 不 喜歡 做 生意 的 那 個 人
   I don't like do business (relativizer) that (classifier) man
   (I don't like that fellow who is a businessman.)

8. 他 是 我的 中學 同學
   he is my high school classmate
   (He was may classmate when I was in high school.)

9. 在 郵局 後面 有 一 排 小小的 平房
   at Post Office behid there is a row little houses
   (There is a row of houses behind the Post Office.)

10. 我們 請 老師 分配 工作
    we ask teacher assign job
    (We ask the teacher to assign jobs for us.)

11. 我 住 在 那 棟 房子 裡
    I live in that (classifier) buildings (localizer)
    (I live in that buildings.)

12. 小孩 笑 他 是 大胖子
    child laugh he is fat
    (The child laughed at his fatness.)

13. 我 發現 我的 一 支 筆 掉 在 教室
    I find my a (classifier) pen lose in classroom
    (I soon discovered that I had left may pen in the classroom.)

Appendix C, cont'd

14. 在這 棵 樹 後 我們 發現 了 一 朵 白色的 花
in this (classifier) tree behind we find (aspect) a (classifier) white flower
(We found a white flower behind this tree.)

15. 我 相信 王五 一定 以為 張三 很 喜歡 那 位 漂亮的 小姐
I believe Uang-wu would think Jang-san very like that (classifier) pretty lady
(I believed that Uang-wu would think Jang-san liked that pretty lady very much.)

16. 大家 一定會 認為 這 是 一 件 錯的 事
people must think this is a (classifier) wrong thing
(People would think this is a wrong thing.)

17. 我 把 你的 一 封 信 放在 桌 上
I (ba) your a (classifier) letter put on deak (localizer)
(I left a letter of yours on the desk.)

18. 你 明天 上課 大概 上到 幾點呢
you tomorrow class about to what time
(What time will your class end tomorrow ?)

19. 他 想要 把 那 棟 房子 賣給 我
he want (ba) that (classifier) house sell to I
(He wants to sell that house to me.)

20. 他 說 他 很 喜歡 把 那 輛 綠色 車子 停 在 門口的 那
he say he very like (ba) that (classifier) green car park on door that
棵 大 樹 旁
(classifier) big tree behind
(He said he liked very much to park that green car beside the big tree near the front door.)

21. 小鳥 整天 坐 在 這 棵 樹 上 等 媽媽
little bird all day long sit on this (classifier) tree (localizer) wait for mother
(A little bird sat on the tree all day long waiting for its mother.)

22. 他 說 他 很 喜歡 把 那 輛 綠色 車子 停 在 門口 的 那
he say he very like (ba) that (classifier) green car park on door of that
棵 大樹 旁 的 一 條 小 巷子 裡
(classifier) big tree (localizer) of a (classifier) small alley (localizer)
(He says he likes very much to park that green car in a small alley which is beside the big tree near the front door.)

23. 我們 昨天 過 了 一 個 快樂的 節日
we yesterday have (aspect) a (classifier) happry holiday
(We had a very happy holiday yesterday.)

24. 他 把 一 桶 水 倒 在 那 個 小孩 身 上
he (ba) a (classifer) water pour on that (classifer) child body (localizer)
(He poured a bucket of water over that child.)

25. 他們 說 這 小孩 是 那 所 學校 的 好 學生
they say this child is that (classifier) school of good student
(They say this child is a good student of that school.)

APPENDIX D. Sample Parsing Results

1. INPUT ==> 我 去年 丟掉 的 那 隻 狗, 我 以前 以為
I last year lost (relativizer) that (classifier) dog I before assume
已經 死掉 了, 昨天 居然 被 我 找到 了.
already die (aspect marker) yesterday unexpectedly by me found (aspect marker)

(I used to assume the dog I lost last year must have already died, but it was unexpectedly found by me yesterday.)

```
(SENTENCE
    (NP <NOUN>
      (XPDE <S>
        (S
          (NP <NOUN>
            (N <PRON> ===> 我 I ))
          (VP
            (ADV ===> 去年 last year )
            (V-bar
              (V <V-n> ===> 丟掉 lost )
              (EMPTY (No.0)))))
        (DEn ===> 的 relativizer ))
      (QP <Mc>
        (DET ===> 那 that )
        (CL <Mc> ===> 隻 classifer))
      (N <NOUN> ===> 狗 dog ))
    (S
      (NP <NOUN>
        (N <PRON> ==>> 我 I ))
      (VP
        (ADV ===> 以前 before )
        (V-bar
          (V <V-s> ===> 以為 think )
          (S
            (EMPTY (No.2))
            (VP
              (ADV ===> 已經 already )
              (V-bar
                (V <V-> ===> 死掉 die)
                (Aspect ===> 了 aspect marker ))))))))
    (S=bar
      (S
        (EMPTY (No.5))
        (VP
          (NP <TIME>
            (N <Time> ===> 昨天 yesterday ))
          (ADV ===> 居然 unexpectedly )
          (PP <bei>
            (PREP <bei> ===> 被 by )
            (NP <NOUN>
```

Appendix D, cont'd

```
                (N <PRON> ===> 我 me )))
          (V-bar
             (V <V-n> ===> 找到 found )
             (EMPTY (No.4))))
    (PRTAG ===> 了 aspect marker)))

EMPTY.4 == EMPTY.5
EMPTY.0 == N: ( 狗 dog )
EMPTY.2 == NP: (我去年丟掉的那隻狗 the dog I lost last year )
EMPTY.5 == NP: (我去年丟掉的那隻狗 the dog I lost last year )
```

2. INPUT ===> 這個 錯誤 將來 會 被大家 認為 是 對的
this mistake future will by people believe is correct

(This mistake will be believed to be correct by the people in the future.)

```
(SENTENCE
   (S
      (NP <NOUN>
         (QP <Mc>
            (DET ===> 這個 this ))
         (N <NOUN> ===> 錯誤 mistake ))
      (VP
         (ADV ===>將來 future)
         (AUX ===> 會 will )
         (PP <bei>
            (PREP <bei> ===> 被 by)
            (NP <NOUN>
               (N <PRON> ===> 大家 people)))
         (V-bar
            (V <V-s> ===> 認為 believe)
            (S
               (EMPTY (No.1))
               (VP
                  (V-bar
                     (V <SHI> ===>是 is)
                        (S
                           (EMPTY (No.0)
                           (VP
                                 (V-bar
                                    (V  <ADJ> ===> 對的 correct)))
                        )))))))
EMPTY.0 == EMPTY.1
EMPTY.1 == NP:( 這個錯誤 this mistake)
```

References

Aho, A. V., and Ullman, J. D. (1972). *The Theory of Parsing, Translation, and Compling, Vol. 1.* Englewood Cliffs, NJ: Prentice-Hall.

Bates, M. (1978). "The theory and practice of augmented transition network grammars." *Natural Language Communication with Computers*, 191–259.

Chao, Y. R. (1968). *A Grammar of Spoken Chinese.* Berkeley, CA: University of California Press.

Chen, H. H., Lin, I. P., and Wu, C. P. (1988). "A logical approach to movement transformation in Mandarin Chinese." *International Journal of Pattern Recognition and Artificial Intelligence*, (2)1.

Chen, K. J., and Chang, L. L. (1988). "A classification of Chinese verbs for language parsing." *International Conference of Computer Processing and Oriental Languages.*

Chen, J. J. (1985). "An experimental parsing system for Chinese sentences," M.S. thesis, National Taiwan University, Taipei.

Chomsky, N. (1981). *Lectures on Government and Binding.* Dordrecht: Foris.

Fillmore, C. (1968). "The case for case." In *Universals in Linguistic Theory* edited by Bach and Harms, Holt, Reinhart and Winston.

Gazdar, G., Franz, A., Osborne, K., and Evans, R. (1987). "Natural language processing in the 1980s." *CSLI*, Stanford University.

Gazdar, G. (1988). "Categorial structure." *Computational Linguistics* 14:1–19.

Hellwing, P. (1988). "Chart parsing according to the slot and filler principle." In *Proceedings, International Conference on Computational Linguistics*, 242–244.

Ho, W. H. (1984). "Automatic recognition of Chinese words." M.S. thesis, National Taiwan Institute of Technology, Taipei.

Huang, J. (1982). "Logical relations in Chinese and the theory of grammar." Doctoral dissertation, Massachusetts Institute of Technology, Cambridge, MA.

Jiang. (1985). "Chinese parsing: An initial exploration at LRC." *Computer Processing of Chinese and Oriental Language* 2(2): 127–138.

Kay, M. (1980). "Algorithm schemata and data structures in syntactic processing." Xerox Report CSL-80-12, Palo Alto, CA.

Li, C. N., and Thompson, S. A. (1981). *Mandarin Chinese.* University of California Press.

Lin, L. J. (1985). "A syntactic analysis system for Chinese sentences." M.S. thesis, National Taiwan University, Taipei, Taiwan.

Lin, L.-J.; Huang, J.; Chen, K.-J.; and Lee, L.-S. (1986). "SASC: A syntactic analysis system for Chinese sentences." In *Proceedings, International Conference on Chinese Computing.* Singapore.

Lin, L.-J.; Chen, K.-J.; Huang, J.; Lee, L.-S. (1986). "A Chinese natural language processing system based upon the theory of empty categories." In *Proceedings, Fifth National Conference on Artificial Intelligence* (AAAI). Philadelphia, PA.

Marcus, M. P. (1982). *A Theory of Syntactic Recognition for Natural Language.* Cambridge, MA: The MIT Press.

Pareschi R., and Steedman, M. (1987). "A lazy way to chart-parse with categorial grammars." In *Proceedings, 25th Annual Meeting of the Association for Computational Linguistics.*

Radford, A. (1981). *Transformational Syntax: A Student's Guide to Chomsky's Extended Standard Theory.* Cambridge, U.K.: Cambridge University Press.

Sells, P. (1985). "Lecture on contemporary syntactic theories: An introduction to government-binding theory, generalized phrase structure grammar, and lexical-functional grammar." CSLI.

Sheiber, S. M. (1986). *An Introduction to Unification-Based Approaches to Grammar.* Chicago: University of Chicago Press.

Steedman, M. (1985). "Dependency and coordination in the grammar of Dutch and English." *Language.* **61**, 523–568.

Stock, O., Falcone, R., and Insinnamo, P. (1988). "Island parsing and bidirectional charts." In *Proceedings, International Conference on Computational Linguistics.*

Tomita, M. (1986). *Efficient Parsing for Natural Language: A Fast Algorithm for Practical Systems.* Boston: Kluwer.

Wehrli, E. (1988). "Parsing with a GB-grammar." In *Natural Language Parsing and Linguistic Theories*, edited by U. Reyle and C. Rohrer, Dordrecht, Boston: D. Reidel, distributed by Kluwer, 177–201.

Winograd, T. (1983). *Language as a Cognitive Process. Vol. 1: Syntax.* Reading, MA: Addison-Wesley.

Woods, W. (1970). "Transition network grammar for natural language analysis." *CACM* **13**(10), 591–606.

Yang, Y. (1987). "Semantic analysis in Chinese sentence analysis." In *Proceedings, International Joint Conference on Artificial Intelligence* (AAAI). Milano, Italy.