SEMANTIC ANALYSIS OF THE
THAI LANGUAGE

by

VACHIRA TERAYANONT

Bachelor of Science
Chulalongkorn University
Bangkok, Thailand
1979

Master of Science
University of Missouri-Rolla
Rolla, Missouri
1982

Submitted to the Faculty of the
Graduate College of the
Oklahoma State University
in partial fulfillment of
the requirements for
the Degree of
DOCTOR OF PHILOSOPHY
July, 1988

SEMANTIC ANALYSIS OF THE THAI LANGUAGE

Thesis Approved:

G. E. Hedrick
Thesis Advisor

Marilyn G. Kletke

John J. Dewey, Jr.

M J Folk

Norman N. Durham
Dean of the Graduate College

PREFACE

A model for parsing a Thai sentence is developed. The theory of the Distributed Word Expert Parser serves as the basis for this work. The knowledge of each word is stored in a word expert. The parser is a model of coroutine control and intercommunication. The central parsing process is to understand word role in a particular context.

I wish to express my sincere gratitude to all the people who assisted me in this work during my stay at Oklahoma State University. In particular, I am especially grateful to my major adviser, Dr. George E. Hedrick, who not only aided my study from the beginning, but also gave generously of his time and encouragement to guide my research.

I am also thankful to the other committee members, Dr. Michael J. Folk, Dr. John J. Deveny, and Dr. Marilyn G. Kletke, for their interest in this project.

Grateful acknowledgment is made to Mr. Vorasak Toommanon who devoted his time to help me with some illustrations written in Thai. Special thanks are due to Dr. Charnchai Siriwat for his kind help in providing material about the Thai grammar.

I wish to express my appreciation to the department of Computing and Information Sciences for the financial support I received during the course of this work.

My deepest gratitude goes to my mother, Dr. Voranuj Terayanont, for her endless love and unfailing encouragement. I dedicate this study to her.

TABLE OF CONTENTS

Chapter Page

LIST OF TABLES

Table Page

LIST OF FIGURES

CHAPTER I

INTRODUCTION

Statement of the Problem

Historically, communication between people and computers has been achieved by using highly structured computer-imposed techniques. This mechanism, however, requires some level of training and experience for computer users. During the past decade, the range of computer users has grown from highly-experienced programmers to those who do not have any knowledge about programming. Thus, to make the computer easier for a non-programmer to use, a new means of communication between humans and machines should be developed. One of the proposed ideas is to have human interaction with computers in their everyday use languages.

These common languages that people use to communicate with each other are called "natural languages". Thus, a natural language (NL) system can be defined as one that allows a person to interact with a computer by using a common language.

The most obvious advantage of an NL system is that people need not to be trained in a programming language to use a computer. Gevarter (1983) has listed several applications for computer-based NL systems. In the information access system, NL is useful for information retrieval, question answering, and computer-aided instruction. To interact with intelligent programs, people could benefit in using NL as an interface either to an expert system or to a decision support system.

One of the questions that has been raised in the field of a natural language understanding (NLU) system is "Is a model of human language comprehension attainable?" (Moyne, 1985). So far, this question is remain unanswered. In a recent study, Postal and Langendoen (1984) claimed that natural languages are neither finite nor infinite enumerable sets. The collection of sentences in a natural language is a "megacollection" with higher cardinality than any enumerable set. From the observation of different approaches that have been proposed up to present, Moyne (1985) concludes that human language comprehension cannot be modeled entirely by any one of those approaches. However, by stating that, Moyne does not imply the failure of any attempts. He suggests that human language comprehension involves all of those approaches.

In building an NLU system, one of the first major decisions to be made is how to express and organize linguistic and conceptual information for the parser. Most models of parsing to this point have made the assumption that systems of rewriting rules are good media for expressing information about sentence level parsing. Although the implementation styles of these models differ markedly, they share the view that rules of language understanding are captured best by rules which span large sentence constituents. This assumption analyzes the language by imposing structure at the sentence level and treating the words of the language as tokens. These tokens participate in comprehension by virtue of their inclusion in sentence and concept level rules.

Another problem in an NL system is dealing with word ambiguity. Traditionally, the approach is to write one rule for each usage of the word. Each such rule must contain enough contextual probes so that it executes at exactly the appropriate moments. This approach leads to highly redundant context descriptions in each rule. In addition, some uniform interpreter that is capable of rule arbitration must exist to select the "most appropriate" rule.

Rieger and Small (1981) made an observation that each word of a natural language is an object with an often rich information structure attached to it. For each word in the language, we know its contextual uses, its morphology, and its idiomatic uses. To solve problems of the traditional systems mentioned above, Rieger and Small propose to include all such information in a system of rules that make reference to the world. This system is called "Distributed Word Expert Natural Language Parsing".

In the model, each word expert has its own information of all the possible contextual interpretations of the word it represents. When placed in a run-time context, a word expert should be capable of issuing an orderly progression of inquiries about its context, then making a set of decisions based on the response to it queries. This process leads to a selection of the best interpretation of its word in the sentence. One of the advantages of a word expert system is that the only units the parser requires in memory are the word expert modules formed from a sentence. This feature requires less space in the memory of the computer than the earlier systems. Another advantage is that it allows for the modular growth of language information. Since each word expert is a center of information, one easily can comprehend the structure. There are also other advantages that Rieger and Small claim as well; for example, the characteristic of the theory suggests some approaches to language acquisition. Since each word expert is a self-contained unit, when a new usage of the word is perceived, rather than writing absolute rules to describe the new context, it is possible to grow a new branch within the word expert. This branch only needs to capture one relative difference between the existing usages and the new usages.

Rieger and Small (1981) have used their model to analyze English text. They did not mention the use of their model in the field of foreign languages. Mengel

(1984) developed a parser for the German language based on the theory of distributed word expert parsing. Even though her system does not cover the whole grammatical structure of German, it works well enough to show that this theory can be applied very well with one non-English language.

This word expert system would be very helpful in developing semantic analysis for the Thai language. At present, in Thailand, computer users must communicate with the machine in an English-command language. It requires two levels of training: learning English and learning the operating commands. The users could master the use of a computer faster by communicating with the machine in their native language; i.e., Thai. Communication in natural language would also reduce the cost of training personnel to use a computer.

Thus, to enable a better communication with computers, a system to process and understand the Thai language is desirable.

Objective and Scope

This study is considered as a preliminary research in constructing a Thai language understanding system. Hence, Thai computer users can communicate with machines in their own language.

Only a subset of the Thai language is considered in constructing the parsing system. This subset is defined in Chapter III. Based on the theory of the Distributed Word Expert Parser, a set of operations that involve in analyzing words in a Thai sentence is given. The system generates a general representation of an input sentence. This representation is applicable to any system that may aid the users to communicate with computers in Thai.

Specific objectives for this study include:

1. development of a system that can process a single Thai sentence and generate a reasonable semantic representation of the input sentence.
2. development of a parser module to analyze Thai words in a sentence based on the concept of the word expert parser.

Since the Thai alphabet is different than the English alphabet, a system that represents each Thai word must be developed. The design and development of such a system is beyond the scope of this study.

Definitions

For clarity, the following definitions are given:

Natural Language Understanding System - a computing device and programs which can perform tasks, similar to those done by people, and which requires "intelligence and reasoning" in understanding language.

Natural Language Processing System - a general purpose language processor which builds a formal representation of the input.

Parse - To analyze a sentence by applying the rules of a natural language system.

Ambiguous Sentence - A sentence with more than one valid grammatical parse.

Syntax Analysis - The analysis of the sentence according to its grammatical structure.

Semantic Analysis - The analysis of the sentence according to the meaning of words.

Concept - A data object created by and during the parse to represent comprehension result.

Sense - A distinct usage of a word. It may refer only to the meaning of the word.

Organization of the Study

The results of this study are represented in five major parts: 1) the statement of the problem, objective and scope of the study, 2) a literature review in the area of natural language processing systems, 3) the grammatical structure of Thai language, 4) the word expert parser,and 5) an evaluation of the system with conclusions of the study.

CHAPTER II

LITERATURE REVIEW

Introduction

In developing a natural language processing (NLP) system, several theories have been proposed. For the earlier attempts, most of the theories are characterized as syntax-based analyses. The models first try to parse the input sentences then analyze them according to the syntax of those specified languages. Next, a semantic representation will be applied to provide a meaning of the input sentence. Several researchers point out that in the process of language understanding of human beings, both of the syntactical components and the meaning of words must be taken together. Therefore, later theories will focus on the subject of how to map the linguistic objects to the non-linguistic objects.

This study reviews some of the well known theories of NLP system starting from the earliest work of Chomsky until the present. The implemented systems of these theories are also discussed. Finally, systems of natural language processing for non-English languages are observed.

Transformational Grammar

The first attempt to construct a theory for natural language was done by Chomsky (Charniak and Wilks, 1978). He introduced a theory of transformational grammars (TG). This theory distinguishes grammatical from non-grammatical sentences. A set of rules which generate all the grammatical sentences is produced. These rules are called "transformation rules". The grammatical sentences are generated by starting with a subset of sentences then using transformation rules to generate the rest of them.

Sentences are simply strings of words that have internal structures. To illustrate an internal structure, Charniak and Wilks (1978) use the following sentence as an example.

The big boy laughs.

An internal structure that represents this sentence is shown in figure 1. The sentence is divided and subdivided into individual words. A diagram like this is called a "phrase marker". The individual words at the bottom are called "terminal nodes".

To formulate transformation rules, the Charniak and Wilks system starts with a small set of phrase markers and generates the complete set of phrase markers for all grammatical sentences. In technical terms, a phrase marker is called "deep structure" and the result of the process is "surface structure". Figure 2 shows a diagram of this process. When generating surface structures from

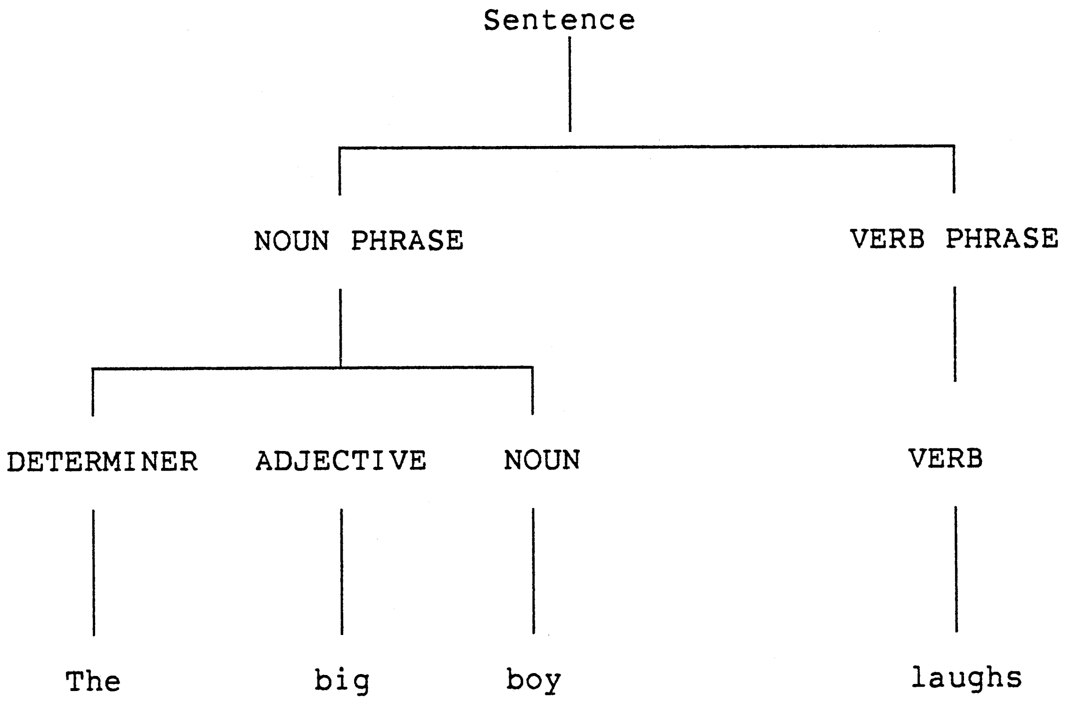

Figure 1. Internal Structure of a Sentence
(Charniak and Wilks, 1978, p. 28)

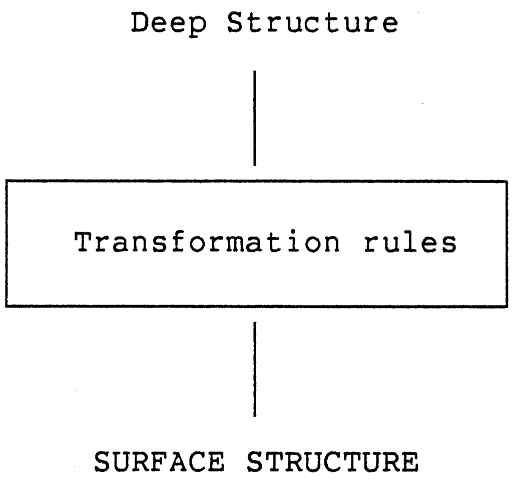

Figure 2. Process of Applying Transformation Rules

(Charniak and Wilks, 1978, p. 32)

deep structures, transformational grammars describe the relationship between deep structures and surface structures.

A syntactic component of TG has two parts:

1) a base component which generates the deep structure phrase markers, and
2) a transformation component which generates the phrase markers of all the other sentences of the language.

A TG system uses transformational rules to determine all possible parses of a sentence. For an ambiguous sentence, a sentence would have more than one valid grammatical parse. However, the model of TG could not select the "correct parse" since it relies only on syntax (Wilks, 1975). TGs are useful for computer generation of natural language but cannot be used directly for analyzing sentences (Charniak and Wilks, 1978).

Among the followers of TG theory, Katz and Fodor are the first who made a serious attempt to involve semantic into theory of TG (Charniak and Wilks, 1978). Their theory interprets a grammatical description to produce readings for the sentence. Each reading produced corresponds to a different meaning of the sentence. The theory has two components. The first component is a dictionary in which the meanings of individual words are listed together with restrictions on how words can combine with others meaningfully. The second component is a set of rules which defines how the meaning of a sentence may be built from its component words by using the information in the dictionary. However, Katz and Fodor did not set out a complete theory but only described the shape of their ideas.

Augmented Transition Network

Another well known system that is an implementation of a TG is the augmented transition network (ATN). An ATN is a form of an augmented pushdown automaton. It is a directed graph with labeled states and arcs. The labels on the arcs may be state names as well as terminal symbols. An ATN model builds up a partial structural description of the sentence as it proceeds from state to state through the network. The pieces of this partial description are held in registers. It has the ability to perform arbitrary computational tests and actions associated with the state transitions (Woods, 1970). Advantages for using ATN as a model for natural language include: 1) clarity of presentation, 2) generative power, 3) efficiency of representation, 4) the ability to capture linguistic regularities, and 5) efficiency of operations. However, ATN are tied closely to their application which makes them be nonportable and nonextensible. An improvement is developed in another system called Cascade ATN's (Wood, 1980). It allows more semantic analysis during parsing but it still is tied closely to its application.

The ATN syntactic parser is employed in LUNAR which is a system of natural language interface to moon rocks data base (Woods, 1973). In addition to a heuristic search implementation of the ATN parser, Woods also introduced a very general notion of quantification based on the predicate calculus, and he used sophisticated techniques to translate questions into data base queries (Waltz, 1982). The program parses sentence sent on to the semantic program for translating into a query. The semantic analyzer gathers information from verbs and their cases, nouns, and noun modifiers to build the data base query. This program has a capacity to handle 90% of the questions posed to LUNAR by geologists. However, the system is based on a "closed world" viewpoint of 3,500 words in the vocabulary for moon rock data base. Utterances are limited to strict data base inquiries. The system proved to be non-portable and non-extensible; it is no longer in use (Gevarter, 1983).

Another system that follows the idea of the ATN parser is called ROBOT/INTELLECT (Gevarter, 1983). It is one of the first natural language database query systems to be available commercially. The system handles a large vocabulary by building an inverted file of data element names indicating the data domains in which each name occurs. A dictionary of common English words is also included. The system is considered to be portable since the user can adapt the model to a new data base in approximately one week. Its limitation is that it does not consider context except to disambiguate pronouns.

SHRDLU

Winograd's system, called SHRDLU, is one of the systems that is based on syntax (Winograd, 1972). The system runs as a dialogue between a human operator and the machine. It displays on a screen pictures of a closed world of closed blocks and pyramids, a box that objects can be put into, and an arm that can move the objects. SHRDLU displays the responses to the operator by writing on the same screen. Winograd starts the analysis of an input sentence by determining a certain interpretation of a sentence. If the system cannot make syntactical or semantical sense of the sentence, SHRDLU will back up and try a different parse. If there is no semantic objection, then the parser will continue.

This system is one of the first systems to deal simultaneously with many sophisticated issues of NLP: parsing, semantics, references to previous discourse, knowledge representation, and problem solving. SHRDLU views the world as a logical solving universe (Gevarter, 1983). SHRDLU has a capacity to solve a broad set of problems. It interprets declarative sentences as data base updates, interrogative sentences as data base searches, and imperative sentences as specifications for goal. The system will first form a plan and then execute that plan. This process involves data base search and update as well as sentence generation (Waltz, 1982).

SHRDLU assumes the world is logical, simple, small, and closed. It also assumes that it knows everything about the world. The user must also be familiar with the system to use it successfully. SHRDLU is a non-portable and non-extensible prototype. It is no longer in use (Gevarter, 1983).

Conceptual Dependency

In the above approach to linguistic theory, syntax and semantic parts of language are analyzed separately. Many researchers agree on the same point that people understand sentences with respect to both the linguistic and situational contexts in which those sentences are spoken. From these observations, a new different theory is proposed to integrate both semantic and syntax parts of language as a whole unit in building an NLU system. This theory is called "conceptual dependency" (Schank, 1972). In this theory, Schank attempts to represent the conceptual base that underlies all natural languages. In other words, the aim of this theory is to explain how a linguistic object is mapped to a non-linguistic object. Semantic structures are claimed to be of the same formal nature as syntactic structures. The notion of "deep structure" which separates syntax from semantics and a distinction between "transformation" and "semantic representation rules" become a single system of rules which relates semantic structure and surface structure. A grammar does not generate a set of surface structures; rather, it generates a set of derivations.

Five characteristics of Schank's theory are listed as follows (Charniak and Wilks, 1978):

1) It is conceptually based;
2) The conceptual base consists of a formal structure;
3) It makes predictions based on the conceptual structure;
4) Its understanding is not limited to sentences;
5) It has formal rules to map natural language utterances to the conceptual base.

There are two distinct levels of analysis: the sentential level and the conceptual level. On the sentential level, the utterances of a given language are encoded within a syntactic structure of that language. On the conceptual level, the basic construction is the conceptualization. A conceptualization consists of concepts and certain formal relations that exist between these concepts. The concept can either be a nominal, an action, or a modifier.

Relationships between each of conceptual categories are called "dependencies". Dependency relations are established upon the basis that a concept alone or in combination with the other(s) can be understood. There are two types of dependency relations between two concepts: a dependent and a governor. It is the fact that a governor need not have a dependent but a dependent must have a governor.

The conceptual base is represented by a linked network of concepts and dependencies between concepts. It is called a conceptual dependency network. Figure 3 shows examples of conceptual dependency networks corresponding to two grammatically distinct sentences. Figure 3a is a network for a sentence, "John gave Mary a bicycle." It can be read from this structure as, "John transferred a possession of the bicycle from himself to Mary." In figure 3b, the sentence, "Mary got a bicycle from John.", has the similar representation except that Mary is listed as an agent. This is because, in figure 3b, Mary is the person who caused the action.

Schank implemented his own theory in a system called MARGIE. The program can accept simple sentences and answer questions about them, generate paraphrases of those questions, and make inferences base on the questions (Waltz, 1982). Several other computer implementations have also been developed follow Schank's idea (see Moyne, 1985). However, space limitation prevent the use of those systems.

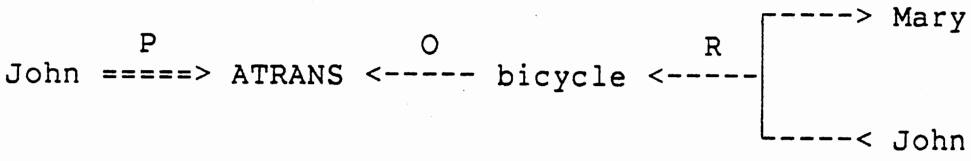

Figure 3a. Conceptual-Dependency Diagram of the Sentence, "John gave Mary a bicycle."

(Waltz, 1982, p. 12).

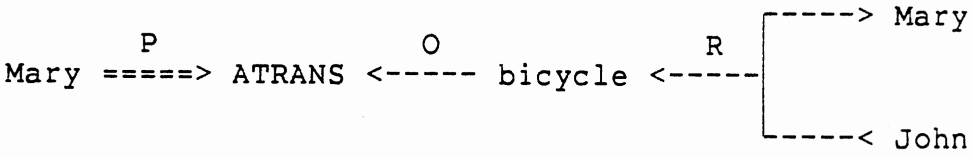

Figure 3b. Conceptual-Dependency Diagram of the Sentence, "Mary got a bicycle from John."

(Waltz, 1982, p. 12).

Preference Semantics

Another theory that accounts for a semantic representation of natural language is proposed by Wilks (1975). He developed a system that analyzes an English text to generate a French translation. This system is called "Preference Semantics". The term "preference" is used because the procedures are used to derive prefering certain structures on the basis of semantic density. The fundamental unit of this semantic representation is the template. Each template corresponds to an intuitive notion of an agent-action-object form. Templates are built from basic building blocks called formulas. These formulas correspond to senses of individual words, one formula to a word sense. In order to construct a complete text representation, namely "semantic block", templates are bound together by two kinds of higher level structures called paraplates and inference rules. Wilks shows the template connectivity of formulas corresponds to the sentence, "The black horse passed the winning post easily." This example is presented in figure 4.

Wilks's system runs on-line as a package of LISP, MLISP, and MLISP2 programs. The two latter languages are expanded LISP language that have a command structure and pattern matching capacities. Presently, a vocabulary is only 500 words but Wilks claims that it is the largest of any operating deep-structure semantic analyzer. This system is designed with clever rule and expectancy

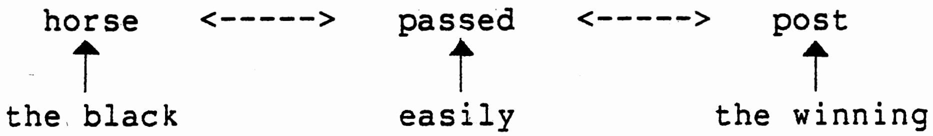

Figure 4. Example of Template Connectivity of Formulas Used in Wilks's System.

(Wilks, 1975, p. 59).

switching on a limited set of conceptual ambiguity. Unfortunately, the system does not deal with the problem of word ambiguity in a foundational sense (Rieger and Small, 1981).

Distributed Word Expert Parsing

Motivated from the earlier works of Riesback and Wilks, Rieger and Small (1981) introduced another theory called Distributed Word Expert Parsing. They use the idea of Wilks in handling multiple word senses and build mechanisms deeply into the model to deal with word ambiguity.

A word expert is a procedural entity of all the possible contextual interpretations of the word it represents. Each word contains enough information to indicate its context in a sentence and word senses to give the meaning. In analyzing the input sentence, each word expert is compiled and can ask questions of one another to contribute its meaning to the final interpretation of the sentence. The parser is organized in two levels: sentence level and the concept level. In the sentence level, workspace contains a word bin for each word. For the concept level, workspace contains a concept bin which is a repository for information about a single word of the input. The principal structure of the model is the word sense discrimination expert.

Rieger and Small illustrate how the Word Expert

Parser successfully parses the sentence, "The deep philosopher throws the peach pit into the deep pit." Initially, the parser retrieves the experts for "the", "deep", "philosopher", "throw", "s", "peach", and so forth. Then the system organizes those word experts along with data in word bins. The parsing is a left to right order in the sentence level workspace. The word expert for "the" will run first, then terminates immediately, and creates a new concept designator. This new unit is called a concept bin and participates in the concept level workspace.

Next the "deep" expert runs. Since "deep" has a number of word senses, it is unable to terminate. It will suspend the execution, noting the conditions upon which it would be resumed. These conditions are referred to as the "restart demon". The expert for "philosopher" runs, checks the control state of the parser, and contributes the fact that a new concept refers to a person who study philosophy. When this expert terminates, the expert for "deep" resumes and realizes that "deep" must describe an entity that can be viewed as a person. Therefore, the "deep" expert terminates with the fact that the person is intellectual. The expert for "throw" then runs, examines its right lexical neighbor, decides to wait for the existence of an entire concept to its right. The "s" expert runs and contributes it standard morphological information to "throw"'s data bin.

The expert for the second "the" runs and creates a new concept bin to represent the data about the nominal and description to come. The "peach" expert realizes that it could be either a noun or an adjective. It asks the expert for "pit" if the two of them can form a noun-noun pair. This causes "peach" to be in a state called "attempting pairing." "Pit" answers back that it does pair up with "peach" and enters the "ready" state. "Peach" now has its correct sense and terminates. The "pit" expert can execute quickly and terminates with "fruit pit" sense. This action causes the "throw" expert to resume and with the sense of fruit pit, the "throw expert" terminates with its word sense as an event of propelling with one's arm.

When the "into" expert runs, it opens a concept bin for the time, location, or situation and suspends itself. The third "the" expert then executes immediately and creates the expected picture concept. The word expert for "deep" will run and cannot decide among its several senses. Therefore, it will suspend and waits for the word expert of its right. "Pit"'s expert now runs and with the expectation posted by "deep", "pit" maps its contribution to a large hole in the ground. Therefore, the "deep" expert can be resumed and terminates as well as closes the concept bin that they belong to. The "into" expert can resume and marks its concept as a "location", then terminates. Now all the word experts are completed, the expert for "." runs and completes the parse.

Natural Language Processing System for Non-English Languages

As previously noted, being able to converse with computer in the users' native language brings benefits not only to the users themselves but also to their organizations. The United States has been a leader in developing English language interfaces to computers since the first language theory was developed by Chomsky. There is also a need for non-English users to communicate with the machine in their native languages. Because of the multilingual environment, many of the NLP systems have been developed in European countries. However, these systems have focused more upon theoretical aspects because the technology available in Europe is not as advanced as the American technology (Bibel, 1985).

A typical example of the European approach to NLP is the PRISE system developed in Italy. PRISE receives as input a set of semantic specifications describing the concepts which are to be expressed and outputs the appropriate Italian sentences. Based upon the theory of Conceptual Dependency, PRISE also includes the ability to generate new translation rules and to update its vocabulary. These capabilities allow flexibility in learning new words and redefining its conceptual representation rules to handle new applications.

PRISE generates its initial output in a conceptual description language (CDL) which is then translated into Italian. This translation is a two step process. The first step performs such basic operations as word selection syntactic group formation, and relationship definition. This first step also includes the process of relating the syntactic components to each other. The second step chooses the best sequence of syntactic components to generate the final Italian sentences.

PRISE, itself, is a component of a larger conversational system still under development. This system is written in PROLOG and LISP and runs on a DEC VAX 11/750 under UNIX (Adorn et al, 1984).

Also under development in Italy is an Italian interface to an airline reservation system. This reservation system uses a packet of pattern-action rules to interpret its queries. This system, implemented in LISP on a VAX 11/780, operates upon a small domain of data base (Cudazen et al, 1984).

German researchers are also very active in the NLP field. The VEB Robotron Zentrum fur Forschung und Technik has developed an interface called NLI/AIDOS for the AIDOS/VS Information Retrieval System. The system attempts to convert natural language information into the equivalent semantic representation. The global structure of the NLI/AIDOS interface consists of four main components:

a. a lexical-morphological analysis (LMA);

b. a word-class-controlled functional analysis (WFA) translating the natural language queries into expressions of a semantic intermediate language (SIL) after the preparatory LMA-step;

c. a lexicon (LX) containing the grammatical-semantic background information;

d. a transformation module (TM) transforming SIL expressions into AIDOS Query Language (AQL) expressions.

Instead of using a set of grammatical rules, the linguistic processor of this system concentrates upon the classifications of words. Each word class invokes a corresponding function whenever a word of this class appears in a sentence. This process makes extensive use of three stacks: STACK, OP-STACK, and QL-STACK. During the analysis, these stacks grow and shrink dynamically.

A transformation module translates the natural language query into the intermediate semantic representation. This module contains a central control section that uses a set of transformation rules to translate the query's linguistic deep structure into the appropriate application expression.

Like so many of the NLP systems, NLI/AIDOS is implemented in LISP on a medium scale system (Helbig, 1984).

Another German project is the SYSAN project to interpret queries for a data base. The goal of the project is to create an interface that the user can tailor to his needs. SYSAN outputs a table of the natural language formulations. SYSAN consists of a lexical analyzer, a syntactical analyzer (based on the ATN grammar), and a semantic analyzer. The analysis process works with the same concept as the NLI/AIDOS system. SYSAN also has the capability of processing grammatically incomplete questions (Koch, 1984).

Romanian researchers have developed the ROUND-S (ROmanian UNDerstanding System) to create a set of natural language processing tools to investigate domain-specific knowledge bases. ROUND-S has three major components: linguistic procedures, semantic representation, and a general problem solving procedure. The syntactical analysis of the model is based upon the ATN grammar. ROUND-S is the system that provides dialogues in Romanian in retrieving knowledges from the collection of programs from the National Program Library. The system is implemented in DMLISP on a PDP 11/45 (Mandutianu, 1984).

Another interesting system currently under development in Japan is an English-Japanese machine translation. This translation system uses Montague Grammar to generate an intermediate representation of meaningful semantic relations in a functional logical form. This logical form is then converted to a conceptual phrase structure form which is associated with Japanese language.

Chapter Summary

In this chapter, different approaches to processing natural language text are discussed. The first theory is based on the syntax of the language; it failed to analyze an ambiguous sentence appropriately. Since human beings understand the language by comprehending both the grammar and meaning of words in a sentence, several attempts have been conducted to involve semantic representation into the grammatical rules. However, the syntax and semantic parts of language still are analyzed separately. There is one approach that integrates both the syntax and semantic parts as a whole unit in building a natural language understanding system. This theory is called "conceptual dependency" which is the basic idea of "distributed word expert processing" which is the main theory applied to the Thai language in this study.

Most natural language understanding systems for non-English languages are developed based on the theory of conceptual dependency. This is a strong, unproved indication to show that in order to have a machine understand the language as closely as human beings do, systems should involve both the syntactic and semantic parts of the language.

CHAPTER III

THAI GRAMMAR

Introduction

Natural languages can be categorized into the following four major groups: inflectional languages, agglutinative languages, polysynthetic languages, and isolating languages. A word in an inflectional language is constructed by adding a prefix or a suffix to a root and rearranging the structure of word. Table 1 shows a comparison of words from different languages that have the same meaning and are built from the same root.

Agglutinative languages, in a similar way, build words by adding prefixes, infixes, or suffixes to roots but the roots remain the same. For instance, in an American-Indian language, the word "cheta" means to build a fire, when the infix "wa" is added to this word, it becomes "chewata" which means "I build a fire." Another example is the word "sev" in Turkish language which means "love", when the postfix "dirmak" is added, it becomes "sevdirmak" which means "try to be loved by someone".

Polysynthetic languages combine words to form new words just as the agglutinative languages do. However, the key difference is that the compound root words are

TABLE I

COMPARISON OF WORDS IN INFLECTIONAL LANGUAGES

Words language	7	8
Latin	Septo	Octo
Italian	Setto	Otto
Sanskrit	Sapta	Ashta
Bali	Satta	Attha
Spanish	Siete	Ocho
Portuguese	Seta	Aito
French	Sept	Huit
English	Seven	Eight

Source: Lamduan, Somchai. Thai Grammar. Odion Store Publisher, Bangkok, Thailand, 1983.

changed when combined in the polysynthetic languages.

The isolative languages, such as Thai, do not combine words to form new words. Characteristically, the conversational vocabulary consists of one syllable words. Sentences are formed from separate words grouped together to convey a thought. Unlike the English language, Thai sentences are written by putting words together without any space between words in the same sentence. Spaces indicate the end of sentences.

Thai written characters are different than those in the Roman alphabets totally. Detail about these Thai written symbols is presented. Also included in this section are word classes, phrase structures, and sentence structures of the Thai language.

Written Symbols

In contrast to some written languages, but in similar fashion to English, Thai written symbols represent the sound of the verbal language directly. These written symbols consist of consonants, vowels, and tonal marks. Thai has forty-four consonants; two are not used in modern writing, being considered obsolete and six more never begin words. The consonants in alphabetical order are shown in figure 5. The consonants also may be divided into the three tonal groups: the low tone, the medium tone, and the high tone. The consonants are arranged tonally as shown in figure 6.

ก	ข	ฃ	ค
ฅ	ฆ	ง	จ
ฉ	ช	ซ	ฌ
ญ	ฎ	ฏ	ฐ
ฑ	ฒ	ณ	ด
ต	ถ	ท	ธ
น	บ	ป	ผ
ฝ	พ	ฟ	ภ
ม	ย	ร	ล
ว	ศ	ษ	ส
ห	ฬ	อ	ฮ

Figure 5. Thai Written Symbols

Medium	High	Low
ก		
	ข	
		ค ฆ
		ง
จ		
	ฉ	ช ฌ
	ศ ษ ส	ซ
		ญ ย
ฎ ด		
ฏ ต		
	ฐ ถ	ฑ ฒ ท ธ
		ณ น
บ		
ป		
	ผ	พ ภ
	ฝ	ฟ
		ม
		ร
		ล ฬ
		ว
	ห	ฮ
อ		

Figure 6. Thai Written Symbols Tonally Arranged

Twenty-one written characters used as vowels in the Thai language are shown in figure 7. Each vowel sound can be represented by vowel marks or vowel combinations. Figure 8 shows a list of vowel sounds with 'อ' added to indicate whether each vowel is placed before, after, above, or below a consonant. Sometimes a hyphen is added to represent a following consonant that must be present. To represent tonal accent in the Thai language, there are four marks, namely, the first tonal mark ('), the second tonal mark (้), the third tonal mark (๊), and the fourth tonal mark (+).

In the Thai language, there are five accent tones: the level tone, the grave tone, the dropped tone, the acute tone, and the rising tone. Medium-tone and low-tone consonants without any tonal marks are pronounced with a level tone. High-tone consonants with no tonal marks are pronounced with a rising tone. High-tone and medium-tone consonants with the first tonal marks are pronounced with a grave tone. Low-tone consonants with the first tonal marks are pronounced with a dropped tone. High-tone and medium-tone consonants with the second tonal marks are pronounced with a dropped tone. Low-tone consonants with the second tonal marks are pronounced with an acute tone. Medium-tone consonants with the third tonal marks are pronounced with an acute tone. Medium-tone consonants with the fourth tonal marks are pronounced with a rising tone.

Symbols	Name
-ะ	วิสรรชนีย์
-า	ลากข้าง
-ิ	พินทุ์อิ
-ุ	ตีนเหยียด
-ู	ตีนคู้
เ-	ไม้หน้า
โ-	ไม้โอ
ไ-	ไม้มลาย
ใ-	ไม้ม้วน
'-	ฝนทอง
"-	ฟันหนู
-ํ	หยาดน้ำค้าง
็	ไม้ไต่คู้
ั-	ไม้หันอากาศ
ฤ	ตัวรึ
ฤๅ	ตัวรือ
ฦ	ตัวลึ
ฦๅ	ตัวลือ
อ	ตัวออ
ย	ตัวยอ
ว	ตัววอ

Figure 7. Thai Vowels

ออ อว- อะ อั-

อา อำ อิ อี

อึ อื อุ อู

เอะ เอ เอาะ เอา

เอิ- เอียะ เอีย เอียว

เอือะ เอือ เออะ เออ

เอย เอว แอะ แอ

โอะ โอ ใอ ไอ

โอะ โอ ใอ ไอ

Figure 8. Thai Vowel Sounds

Word Classes

As previously mentioned, words in a sentence are combined without any blanks to separate them. Thus, to identify a word in a Thai sentence, the only criterion is to find a minimum free form of written symbols in a meaningful sentence. Similar to the English language, words in the Thai language can be classified into seven classes by using the following three criteria.

1. Usage of words

   1.1 Noun which is a name of a person, place, or thing.

   1.2 Verb which is used to express action or state of being.

   1.2 Interjection which is a word expresses strong emotion or passion.

2. Functions of words in the sentence

   2.1 Pronoun which is used to represent a noun.

   2.2 Modifier which is a qualifying adjective and/or a modifying adverb.

   2.3 Conjunction which is used to join words or clauses.

3. Position of words

   There is only type of word in this category, i.e. preposition. A preposition is used before a noun or a pronoun to show the relation to some other word(s) in the sentence.

Figure 9 demonstrates a word class comparison between Thai and English language. However, one word may be used as a noun, as a verb, or as a modifier in sentences. To identify the class of the word, the position of word in a sentence must be observed. For example; if a word is preceding a verb, it is considered as a noun; if a word is preceded by a noun, it is considered as an adjective; if a word is preceded by a subject, and it is followed by an object, it is a verb.

Phrases

Phrase is a word or a group of words that is used as part of a sentence. Phrases can be classified into five types: noun phrase, verb phrase, auxiliary phrase, phrase of location, and phrase of time.

A noun phrase is a noun, a pronoun, a noun and its modifier, or a pronoun and its modifier that can be present as one of the following parts in the sentence.

1. Subject Part (S)
2. Direct Object (D)
3. Indirect Object (ID)
4. Single Noun (N)

A verb phrase is a verb or a verb and its modifier that is present as a verb part in a sentence.

An auxiliary phrase is a modifier or a group of modifiers that is present to stress the emotional meaning of the sentence.

นาม	noun
นามสามัญ	Common Noun
คำชื่อเฉพาะ	Proper Noun
คำนามรวมหมู่	Collective Noun
คำนามธรรม	Material Noun
ลักษณนาม	Descriptive Noun
สรรพนาม	Pronoun
บุรุษสรรพนาม	Personal Pronoun
สรรพนามใช้ชี้ระยะ	Demonstrative Pronoun
สรรพนามใช้ถาม	Interrogative Pronoun
สรรพนามใช้ชี้ซ้ำ หรือแบ่ง หรือรวมคำตามในประโยค	Distributive Pronoun
สรรพนามเชื่อมประโยค	Possessive Pronoun
กริยา	Verb
กริยาไม่ต้องมีกรรม	Intransitive Verb
กริยามีกรรม	Transitive Verb
คำช่วยกริยา	Auxiliary Verb
วิเศษณ์	Modifier
คำวิเศษณ์บอกลักษณะ	Descriptive Adjective
คำวิเศษณ์บอกเวลา	Adverb of Time
คำวิเศษณ์บอกสถานที่	Adverb of Place
คำวิเศษณ์บอกปริมาณหรือจำนวน	Adverb of Number
คำวิเศษณ์บอกความชี้เฉพาะ	Demonstrative Adjective
คำวิเศษณ์บอกความไม่ชี้เฉพาะ	Non-demonstrative Adjective
คำวิเศษณ์แสดงคำถาม	Interrogative Adjective
คำวิเศษณ์แสดงคำขานและรับ	Adverb of Approval
คำวิเศษณ์แสดงความปฏิเสธ	Adverb of Negation
บุพบท	Preposition
สันธาน	Conjunction
คำอุทาน	Interjection
คำอุทานบอกอาการ	Interjection of Action
คำอุทานเสริมบท	Interjection of Expression

Figure 9. Word Class Comparison Between Thai and English Language.

(Lamduan, 1983, p. 116)

A phrase of location is a noun phrase that is preceded by one or more prepositions. This phrase will be the modifier part of the sentence that tell the location of the event.

A phrase of time is a time word or a time word and its modifier that is present to indicate the time of the event.

In the Thai language, there are 2 major phrase structures: noun phrase structure and verb phrase structure.

Noun Phrase Structure: The following are the four major components in the noun phrase structure.

1. Major noun (M)
2. Intransitive verb part (Int)
3. Quantitative part (Q)
4. Indicative part (Ind)

These four components can be arranged into twelve different noun phrase structures. Examples of noun phrase structures in Thai follow. (The English translations are given in a parenthesis). The reader may notice that the position of these four components is different than the English structure. For instance, an intransitive verb part appears after a major noun that it modifies.

1. M

: หนังสือ

: (book)

2. M + Int

: รถคันใหญ่

: (big car)

3. M + Q

: ส้มสามใบ

: (three oranges)

4. M + Ind

: ผู้ชายคนนี้

: (this man)

5. M + Int + Q

: เสื้อใหม่หลายตัว

: (many new shirts)

6. M + Q + Int

: น้ำหอมสองขวดเล็ก

: (two little bottles of perfume)

7. M + Int + Ind

: บ้านเก่าหลังโน้น

: (that old house)

8. M + Q + Ind

: บ้านสามหลังนี้

: (these three houses)

9. M + Ind + Q

: กระเป๋านั่นอีกใบ

: (there is one more bag)

10. M + Int + Q + Ind

: รองเท้าใหม่สองคู่นี่

: (there are two new pairs of shoes)

11. M + Int + Ind + Q

: รถคันเล็กนี่คันเดียว

: (this little car)

12. M + Q + Int + Ind

: กุหลาบสองดอกโตโน่น

: (those two roses)

Verb phrase structure: There are also four major components in a verb phrase structure:

1. Main verb (V)
2. Helping verb preceding main verb (H1)
3. Helping verb following main verb (H2)
4. Modifier (M)

Functionally, helping verbs in Thai are equivalent to the linking verbs and tenses in English. The four components can be arranged into ten different verb phrase structures. The following examples are shown by using the same concept as the examples for noun phrase structures.

1. V

: สะอาด

: (clean)

2. V + H2

: ยังเปิดอยู่

: (still open)

3. V + M

: เย็นจัง

: (very cold)

4. V + H2 + M

: เล่นอยู่กับเด็ก

: (play with a child)

5. V + M + H2

: จะเปิดพรุ่งนี้

: (will open tomorrow)

6. Hl + V

: ควรจะพักก่อน

: (should rest)

7. Hl + V + H2

: คงยังเปียกอยู่

: (is still wet)

8. Hl + V + M

: อยากนั่งจัง

: (want to sit)

9. Hl + V + H2 + M

: จะไปแล้วเหมือนกัน

: (is also going to leave)

10. Hl + V + M + H2

: เคยพูดกันแล้ว

: (has already talked)

Sentence

Similar to several other languages, the sentence is the basic structure of standard written Thai. The most common single criterion is that a sentence must express a complete thought. Thai sentences are built on simple sentence patterns. Following are four of the most frequently used patterns.

1. A sentence with two words in the order of noun comes before verb; e.g.

   : ฝนตก

   (It rains.)

2. A sentence with two words in the order of verb comes before noun; e.g.

   : หิวน้ำ

   (I am thirsty.)

3. A sentence with three words in the order of noun + verb + noun; e.g.

   : คนเคาะประตู

   (Somebody knocks at the door.)

4. A sentence with four words in the order of noun + verb + noun + noun; e.g.

   : แม่ให้ของขวัญตำรวจ

   (Mother gives a gift to the police.)

Words can have variety of functions within the sentence. Seven basic functions of words are:

a. Subject (S) of the sentence which is a noun or a pronoun that acts upon a verb. In pattern 1, 3, and 4, nouns at the beginning of the sentences are subjects.

b. Direct object (D) which is a noun or a pronoun that comes right after a verb in a sentence. In pattern 2, the noun after the verb is a direct object. In pattern 4, the first noun after verb is a direct object.

c. Indirect object (ID) which is the last noun in sentence pattern 4. Indirect object will appear after a direct object.

d. Single noun (N) which is a noun that can appear by itself in a sentence without any verb.

e. Intransitive verb (IT) which a verb that does not need a direct object to act upon. For instance, in sentence pattern 1, that verb is an intransitive verb.

f. Transitive verb (T) which is a verb that needs to be followed by a direct object.

g. Multitransitive verb (MT) which is a verb that requires both direct object and an indirect object.

In Thai simple sentences, these seven types of words can be arranged into exactly twelve different sentence structures. The following examples are shown by using the same concept as the examples for phrase structure.

.1. IT

: สายแล้ว

(It is very late.)

2. S + IT

: ฝนตก

(It rains.)

3. IT + S

: เหนื่อยไหมคุณ

(Are you tired?)

4. T + D

: หิวน้ำ

(I am thirsty.)

5. S + T + D

: คนเคาะประตู

(Someone knocks at the door.)

6. D + S + T

: ผ้านี่เธอเคยใช้ไหม

(Have you ever used this piece of cloth?)

7. MT + D + ID

: อยากถามคะแนนอาจารย์

(I want to ask the grade from the teacher.)

8. S + MT + D + ID

: แม่จะแจกเหรียญบาทเด็กๆ

(Mother is going to give pennies to the children.)

9. D + S + MT + ID

: เหรียญบาทแม่จะแจกเด็กๆ

(These pennies are for mother to give to the children.)

10. ID + S + MT + D

: เด็กๆแม่จะแจกเหรียญบาท

(Children! mother will give you some pennies.)

11. N

: แม่จ๋า

(Mommy.)

12. N + N

: ปากกานี่ของใคร

(Whose pen is this?)

Chapter Summary

In the Thai language, sentences are formed from separate words grouped together to convey a complete thought. Words are put together without any space between them in the same sentence. Thai written symbols and word classes are presented in this chapter. Finally, the phrase structures and simple sentence structures are demonstrated. These sentence structures are used as a subset of the Thai language involved in the development of the project.

CHAPTER IV

THE PARSER MODEL

Design Considerations

As mentioned earlier, the 12 simple sentence structures shown in Chapter III are considered as the basic structure in constructing the parser model in this study. The parser checks the first word in the sentence. If it is a noun, it may contributes itself as a subject of the sentence, a direct object, or an indirect object. Therefore, the parser must check the next word. If the second word is a verb, then the first word is a subject of the sentence. The function of the verb is then considered next. An intransitive verb does not need an object. While a transitive verb does require a direct object and a multitransitive verb needs to have both a direct object and an indirect object. These considerations must be included in the parser model in order to generate an appropriate interpretation of the sentence. When the system processes a transitive verb, there must be a mechanism that tells the system to expect the next word to contribute its sense as a direct object of that transitive verb. The same concept must be applied for a multitransitive verb also.

If the first word of the sentence is a noun and it is followed by another noun, then the second word acts as a subject of the sentence. The function of the first word depends upon the third word. The system must expect that the third word is a verb. If it is a transitive verb, the first word is a direct object. If it is a multitransitive verb, then the first word can be either a direct object or an indirect object. And another object of the sentence is expected at the end of the sentence.

Words in a Thai sentence are written with no space to separate them. A minimum free form of written symbols is considered as a word. Occasionally, a word can be formed by combining two separate free form of symbols. This idea must be taken care of in the system. Since the sense of a new word is different than considering the meaning of those two words separately.

System Overview

The system processes, as input, a single Thai sentence and produces, as output, a conceptual representation of the sentence. The process scheme is shown in figure 10. When a new sentence arrives, it is subjected to a morphological analysis to identify all the possible words. These words and their associated word sense discrimination experts (word experts) are gathered into the parser's workspace. A word expert contains the word-specific linguistic information that directs the

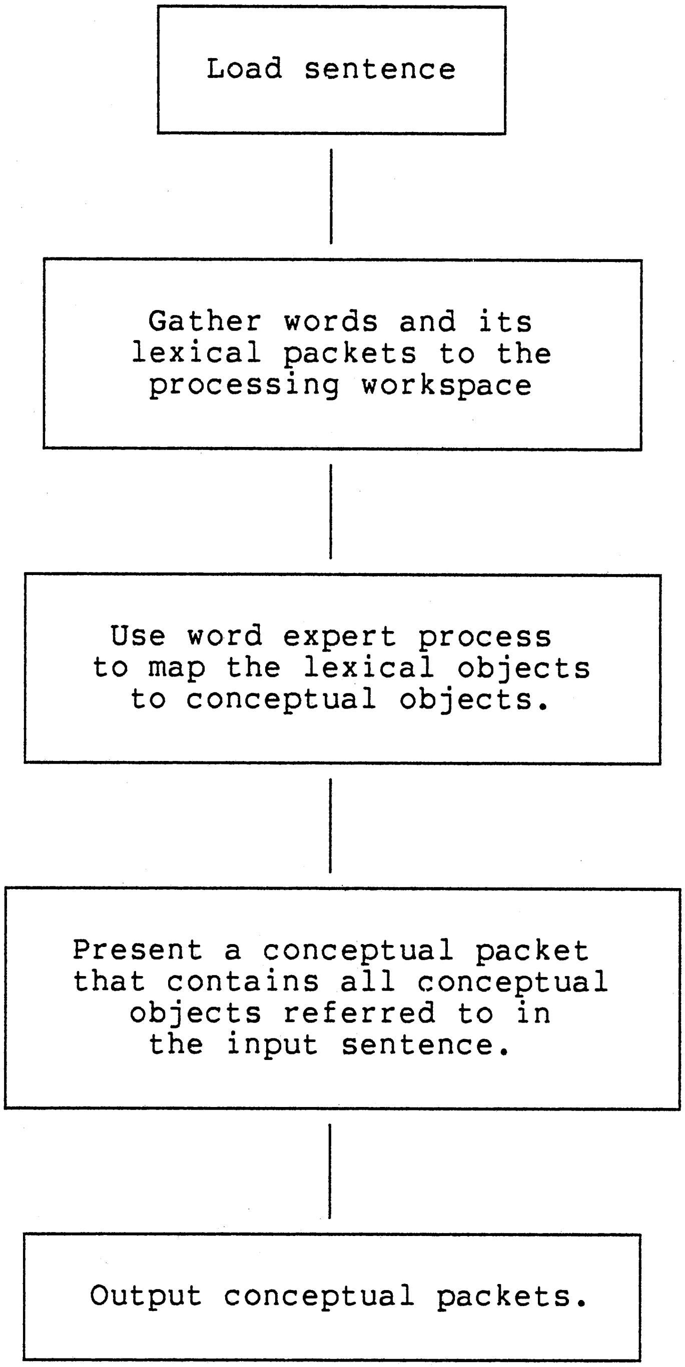
Load sentence
Gather words and its lexical packets to the processing workspace
Use word expert process to map the lexical objects to conceptual objects.
Present a conceptual packet that contains all conceptual objects referred to in the input sentence.
Output conceptual packets.

Figure 10. Process Control Scheme.

execution of the parser.

In the Thai language, words are read from left to right. Thus, the word experts are organized in the workspace such that the system can analyze words in the sentence in the same manner as people read.

The system analyzes the sentence one word at a time by considering the information contained at that word expert. If there is enough information to conclude the concept of that word, the system adds the concept to a conceptual packet and control is passed to the next unprocessed word expert. If there is not enough information, the system suspends its processing at that word expert and consults other experts for more information. After the system is able to diagnose the concept of the suspended expert, the control resumes its process at that expert, adds the new concept to the conceptual packet, and terminates the analysis at the expert. When the system completes the analysis for all words in the sentence, it produces a conceptual representation of the input sentence.

In this study, the main focus is on how to use the information stored at each word expert to parse the sentence. The word expert parser is like a model of coroutine control and intercommunication. The central parsing process is to understand word sense or its role in a particular context.

To communicate among word experts, the information of the control process environment is posted in a central tableau. The table is like a bulletin board where each word expert can obtain the information about its neighbors and the entire process. This table is called a "control state descriptions". The following sections provide more detail on the control state description and some features of the word expert parser.

Concepts

The data objects created by and during the parse to represent comprehension results are called "concepts". A concept typically is created by an expert either to represent a partially or completely finished diagnosis of a word and its context. Each concept in the parser has a type which corresponds to the kind of information found in its conceptual representation. A picture concept involves physical and abstract objects. A time_setting concept is constructed when an adverb of time is found in the sentence. The same idea is applied to an adverb of place, an adverb of number, and an adverb of negation, so that a concept type of place_setting, number_setting, and negation_setting is built, accordingly.

Control State Descriptions

To aid in communication among word experts, a collection of control state information about the processing by the system is gathered in a table. The entries found in this table are the process state of each word expert, the status of each conceptual object, and a description of the state of the entire parser. Table 2 demonstrates valid values for each entry in the control state description table. The information in the table is available to any expert that can make use of information about its own processing or the states of processing of the other experts. Thus, each word expert knows precisely the progress of its neighbors and the state of convergence of the entire process.

Information about the entire process is referred to as process_state in the table. If the system just starts to analyze the input sentence, the value of process_state is NEW_SENTENCE. If any word expert starts to construct a new conceptual object, the type of a new concept is posted in the table.

For each conceptual object constructed during the process, the state of the concept is marked with the signal that the concept is either OPEN or CLOSED. Then for each word expert, its processing state can be marked as one of the three values: inactive, suspended, or terminated.

Word Expert Components

An expert can be pictured as a decision graph in which each node is either an action node or a question node. Some nodes in the graph are designated as entry

TABLE II

VALID FIELD VALUES FOR CONTROL STATE DESCRIPTION TABLE

Field	Value
Process_State:	NEW_SENTENCE PICTURE_SETTING TIME_SETTING PLACE_SETTING NUMBER_SETTING NEGATION_SETTING EVENT_CONSTRUCTION
Concept_State:	OPEN CLOSED
Word_Expert:	INACTIVE SUSPENDED TERMINATED

points. These entry points are places that the process can start using the expert or places that the process can be resumed after the expert suspends itself to wait for more information. An action node generally constructs or adds more information to a concept, posts a signal in control process tableau, suspends itself, and branches to another node. A question node posts a specific question and branches to one of the nodes on the basis of the answer.

Terminal nodes of the graph are the distinct usages of the word. Thus, traversal of a graph converges on a single contextual usage of a word.

When building a word expert, the distinct usages of a word can be identified with a little time and perhaps help from the dictionary. As a simple illustration, several usages each for the words "คน " and "สาย " appears in figure 11.

At each node in the decision graph, a set of operations has to be performed such that the process can decide which node is to be executed next. The following list defines appropriate behaviors that an expert can perform:

1. Asks that the next word of the sentence be read;
2. Consults another word expert for more information;
3. Report new information to the model;
4. Report information in response to a request from another word expert;

Some Word Senses of "คน"

1. A general term for a human being is "คน".

2. To stir a liquid is to "คน".

3. A quantitative unit for human being is "คน".

Some Word Senses of "สาย"

1. A line (especially in the sense of a channel, route, as in "telephone line").

2. A classifier for rivers, canals, roads; for ornamental chains, necklaces; for wires, cables, and for other line-like objects.

3. The late morning is referred to as "สาย".

4. To be late (in the morning).

5. To be too late (with respect to accomplishing some purpose).

Figure 11. Example Contextual Word Usages

5. Builds a new concept or contributes to an existing concept;
6. Suspends itself, set up the condition upon future reawakening;
7. Terminates after having completed its diagnosis of its word's sense or its word's role in the sentence.

To achieve these actions, each word expert needs three major components: a declarative header, a start node, and a body. In an expert's header, it is a description of the expert's behavior in case there is an inter-expert constraint forwarded to it. The interexpert constraint is for the case that the sense discrimination by the previous expert requires that the next expert must provide the knowledge that maps to a specific sense or conceptual category, then the constraint will be forwarded to the next expert. For example, a transitive verb must expect a following noun to contribute as a direct object of that verb. From the specified condition, the description in the header provides the address at which the system expects to continue execution; therefore, unnecessary processing or incorrect reasoning can be avoided.

If there is no constraint passed to the expert, then the normal starting point is defined as a start node.

Inside the expert's body, there is a sequence of nodes together with their set of operations. Each node has a type which is designated by a letter following the node name. A node type can be one of the following four types: Q (question), A (action), S (suspend), and T (terminal).

In addition, each question node is specified as one of the following four choices: MC (multiple choice), C (conditional), and Y/N (yes/no). Again these letters follow the node type to indicate type of question performed at that node.

In appendix A, a list of operations that can be performed at each node is presented. The operations performed at each node depend on the node type.

An action node can branch into next specified node, construct a new concept, report additional information to the previously constructed concept, post signals to the control tableau, and peek at the concept of the word to its right. A word expert may peek at the name of the word expert to its right. Sometimes the currently active expert can be combined with its neighbor to form a new different meaning word.

Conditional and yes/no questions perform simple lookup for the information in the control state description, then branch the execution to another node according to the answer. These operations are helpful. Since the word expert may need some information about the parser state before it can conclude to an appropriate sense.

While multiple choice questions select the most reasonable choice of potentially answers. With this operation, the word expert checks the nature of another previously executed word expert. Then it makes a decision based on the answer to contribute its word sense.

A suspend node sets up the condition that the expert suspends itself and specifies the operations that are needed to be performed upon its reawakening.

A terminal node gives the final result of the execution at the word expert.

Sample of Analysis

To illustrate the idea of word expert performance, the word expert parser is shown in table 3 through the example sentence "คนเคาะประตู" (taken from the example of the sentence structures in chapter III which has a meaning of "Someone knocks at the door."). A step of the parser's execution is presented in the left hand column. Explanation of each execution step is given in the right hand column. Execution of this sentence is an interesting sample to demonstrate how the system solves a problem of word sense ambiguity. Since the first word of the sentence "คน" has more than one sense (previously shown in figure 11), it must suspend itself and wait for the analysis of the next word to its right before it can succeed at determining its sense.

Chapter Summary

The outline of the Thai language parsing system is presented in this chapter. The model is based on the concept of considering the contextual information of each word in a sentence. Thus, the main component of the system is the word expert which contains the word-specific linguistic information that directs the execution of the parser. To aid in communication among word experts, a control state description table is needed. Each word expert can obtain the information about its neighbors and the entire process.

The structure of a word expert also is included in this chapter. Finally, the sample of analysis is given to illustrate the idea of how the parser works.

TABLE III

SAMPLE OF ANALYSIS FOR THE SENTENCE, "คนเคาะประตู"

EVAL: (คนเคาะประตู) PROCESS_STATE: NEW_SENTENCE	The system initializes the workspace by retrieving (from the database) the word experts for the input sentence. Each word expert is placed on the executing queue. The flow of control in the model is controlled by word experts themselves.
EXECUTING 'คน' EXPERT: WORD 1 ====> AT NODE: N0 ====> AT NODE: N1 EXPERT SUSPENDED: WORD 1	The 'คน' expert is started executing. After a short time, it needs information from the word expert to its right. 'คน' temporarily suspends execution. 'คน' posts no expectations and makes no constraints. Therefore, the next expert has full freedom.
EXECUTING 'เคาะ' EXPERT: WORD 2 ====> AT NODE: N0 ====> AT NODE: N1 PROCESS_STATE: EVENT_CONSTRUCTION EXPERT TERMINATED: WORD 2	'เคาะ' now runs and contributes its word sense of "KNOCK_AT_OBJECT" to the conceptual packet. It also puts constraint on the word 'คน' to act as the subject of the sentence.

EXECUTING 'คน' EXPERT: WORD 1 ====> AT NODE: N1 ====> AT NODE: N4 EXPERT TERMINATED: WORD 1	Resumed by the termination of the word to its right and triggered by the constraint, the 'คน' expert concludes its sense to be "HUMAN_BEING" and terminates.
EXECUTING 'ประตู' EXPERT: WORD 3 ====> AT NODE: N0 ====> AT NODE: N1 EXPERT TERMINATED: WORD 3	Since 'ประตู' has a unique sense, it runs and contributes the sense of "DOOR" which acts as an object to the word 'เคาะ' and terminates.
%SCAN CONCEPT (*DESCRIP*(KNOCK_AT_ OBJECT) WORD 2) (*DESCRIP*(HUMAN_BEING) WORD 1) (*DESCRIP*(DOOR)WORD3)	Now every word in the input sentence has been analyzed. The model gathers all the senses that each word contributes to the conceptual packet. These senses, combined all together, describe an event of a human being knocks at the door.

==

CHAPTER V

EVALUATION, SUMMARY, AND SUGGESTED FUTURE WORK

Evaluation

Material presented in this study covers only the theoretical part of the system. Therefore, it is not possible to evaluate the system's performance. The goal of the research described is to develop a model that analyzes, as input, a Thai sentence. Although the model shown is based on simple principles and subject to certain limitations, it is expected to be sufficiently useful in practical applications.

One of the limitations of this system is the assumption that a morphological analysis of the input sentence has been done by some other modules which are not mentioned here. One can develop it easily by using one of the standard pattern matching techniques.

Even though the exact semantic representation of the input sentence is not presented, one may find it is a good idea since the system can be used as a natural language interpreter to any application.

Words rather than rules are the basic units of knowledge in this system. Much of a word expert's knowledge is encoded in a branching discrimination structure. Thus, adding new information about a word involves only the addition of a new branch. This new branch would be placed in the expert at the point where the contextual clues for disambiguating the new usage differ from the existing known usages.

Summary

The model that analyzes an input of the Thai sentence is described here. The language processing components of the system are shown. The control environment is characterized by a collection of generator-like coroutines, called word experts, which cooperatively arrive at a conceptual interpretation of an input sentence. Many forms of information are available to these experts in performing their task, including control state information and information of the world.

Suggested Future Work

As one would expect, much work has to be done on the implementation of the system presented here. The model can be tested only in conjunction with some computer modules of a particular kind of interaction. For example, computer modules to process the input sentence initially are needed. The input sentence must be analyzed morphologically to identify all words in it.

The testing of this model can be started with a small set of vocabulary. In order to effectively analyze a given linguistic input, it is necessary to make prediction for what the output might look like and then compare the actual output to the expected one.

If a large set of vocabulary is involved in the system, an appropriate design of the lexicon has to be considered to provide efficient performance of the system

Extensions to the ideas in this study may include adding the analyzing idioms and more complex sentence structures.

SELECTED BIBLIOGRAPHY

Adorni, G., A. Cappelli, G. Ferrari, L. Moretti, and I. Prodanod. "Syntax and Semantics for Natural Language Processing." Proceedings of the International Conference on Artificial Intelligence: Methodology, Systems, Applications. Verna, Bulgaria, Sept. 17-20, 1984.

Adorni, G., M. Di Manzo, and F. Giunchiglia. "Adaptive Natural Language Generation." Artificial Intelligence and Information-Control Systems of Robots. Ivan Plander, ed. Elsevier Science Publishers B.V. (North-Holland), 1984.

Angelova, Galia; Elena Paskaleva; and Raduslav Pavlov. "On Experimental Linguistic Processors for Man-Computers Dialoques in Bulgarian." Artificial Intelligence: Methodology, Systems, Applications, W. Bibel and B. Petkoff, editors, Elsevier Science Publishers B.V. (North-Holland), 1985.

Arens, Yigal, David Chin, and Robert Wilensky. "Talking to UNIX in English: An Overview of UC." Communications of the ACM, 27, 6 (June 1984), 574-593.

Bibel, Wolfgang. "Artificial Intelligence in Europe," Artificial Intelligence: Methodology, Systems, Applications, W. Bibel and B. Petkoff, editors, Elsevier Science Publishers B.V. (North-Holland), 1985.

Brill, D., J. Burger, M. Crilley, R. Gates, I. Kameny, and J. Weiner. "EUFID: The End User Friendly Interface to Data Management Systems." Proceedings of the 4th International Conference on Very Large Data Bases, West Berlin, German, Sept. 13-15, 1978.

Charniak, E. and Wilks, Y. Computational Semantics, North-Holland Publishing Co., New York, 1978.

Cudazzo, Raffaele, Leonardo Lermo, and Claudia Randi. "Interpretation of Natural Language Queries via Pattern-Action Rules." Artificial Intelligence and Information-Control Systems of Robots. Ivan Plander, ed., Elsevier Science Publishers B.V. (North-Holland), 1984.

Fum, Danilo. "Inferential Reasoning in Natural Language Processing." *Artificial Intelligence and Information-Control Systems of Robots*. Ivan Plander, ed., Elsevier Science Publishers B.V. (North-Holland), 1984.

Gevarter, William B. *An Overview of Computer-Based Natural Language Processing*. NASA Technical Memorandum 85635, April 1983.

Harris, Larry R. "User Oriented Data Base Query with the ROBOT Natural Language Query System." *Proceedings of 3rd International Conference on Very Large Data Bases*, Tokyo, Japan, Oct. 6-8, 1977.

Helbig, Hermann. "Natural Language Access to the Data Base of the AIDOS/VS Information Retrieval System." *Artificial Intelligence and Information-Control Systems of Robots*. Ivan Plander, ed., Elsevier Science Publishers B.V. (North-Holland), 1984.

Hendrix, Gary G., Earl D. Sacerdoti, Daniel Sagolowicz, and Jonathan Slocum. "Developing a Natural Language Interface to Complex Data." *ACM Transactions on Database Systems*, 3, 2 (June 1978), 105-147.

Koch, Dietrich. "German Language Questioning of Relational Database." *Artificial Intelligence and Information-Control Systems of Robots*. Ivan Plander, ed., Elsevier Science Publishers B.V. (North-Holland), 1984.

Kolodner, Janet L. "Indexing and Retrieval Strategies for Natural Language Fact Retrieval." *ACM Transactions on Database Systems*, 8, 3 (Sept, 1983), 434-464.

Lamduan, Somchai. *Thai Grammar*. Odion Store Publisher, Bangkok, Thailand, 1983. (in Thai).

Langengoen, D. T. and Postal, P. M. *The Vastness of Natural Languages*, Oxford: Blackwells, 1984.

McDonald, David D, and Pustejovsky, James D. "Description-Directed Natural Language Generation." *Proceedings of 9th International Joint Conference on Artificial Intelligence*, Los Angeles, California, August 18-23, 1985.

Mandutianu, Sanda. "Round-S: An Experiment with Knowledge Driven Semantics in Natural Language Understanding." *Artificial Intelligence and Information-Control*

Systems of Robots. Ivan Plander, ed., Elsevier Science Publishers B.V. (North-Holland), 1984.

Menzel, Wolfgang. "Data Optimization in Natural Language Based Systems." *Artificial Intelligence and Information-Control Systems of Robots*. Ivan Plander, ed., Elsevier Science Publishers B.V. (North-Holland), 1984.

Meyers, Amnon. "VOX - An Extensible Natural Language Processor." *Proceedings of 9th International Joint Conference on Artificial Intelligence*, Los Angeles, California, August 18-23, 1985.

Moyne, John A. *Understanding Language: Man of Machine*, Plenum Press, New York, 1985.

Nenova, Irina. "On an Implementation of the ATNL Language." *Artificial Intelligence: Methodology, Systems, Applications*, W. Bibel and B. Petkoff, editors, Elsevier Science Publishers B.V. (North-Holland), 1985.

Panevova, Jarmila. "Natural Language Interface to an Expert System." *Artificial Intelligence and Information-Control Systems of Robots*. Ivan Plander, ed., Elsevier Science Publishers B.V. (North-Holland), 1984.

Rieger, Chuck and Small, Steve. "Toward a Theory of Distributed Word Expert Natural Language Parsing." *IEEE Transactions on Systems, Man, and Cybernatics*, *SMC-11, 1 (Jan. 1981), 43-51*.

Rieger, Chuck and Small, Steve. "Parsing and Comprehending with Word Experts (A Theory and Its Realization)." *Strategies for Natural Language Processing*. Wendy G. Lehnert and Martin H. Ringle, eds. Lawrence Erlbaum Associates, Inc.m Hillsdale, New Jersey, 1982, 89-147.

Schank, Roger C. "Conceptual Dependency: A Theory of Natural Language Understanding." *Cognitive Psychology*, 3 (1972), 552-631.

Small, Steve L. "Word Expert Parsing." in *Proceedings of 17th Annual Meeting Association for Computational Linguistic*, 1979.

Waltz, D. L. "The State of the Art in Natural Language Understanding." *Strategies for Natural Language Processing*. Wendy G. Lehnert and Martin H. Ringle, eds., Lawrence Erlbaum Associates, Inc., Hillsdale, New Jersey, 1982, 3-32.

Wilks, Yorick. "An Intelligent Analyzer and Understander of English." *Communications of the ACM*, 18, 5 (May 1975a), 264-274.

Wilks, Yorick. "A Preferential, Pattern-Seeking, Semantics for Natural Language Inference." *Artificial Intelligence*, 6 (1975b), 53-74.

Winograd, T. *Procedures as a Representation for Data in a Computer Program for Understanding Natural Language*., MIT, 1971.

Winograd, T. *Understanding Natural Language*, New York: Academic Press, 1972.

Woods, W. A. "Transition Network Grammars for Natural Language Analysis." *Communications of the ACM*, 13, 10 (Oct. 1970), 591-606.

Woods, W. A. "An Experimental Parsing System for Transition Network Grammars. in *Natural Language Processing*, ed. R. Rustin. Amsterdam: North-Holland, 1973.

Woods, W. A. "Cascaded ATN Grammars." *American Journal of Computational Linguistics*, 6, 1 (Jan.-Mar. 1980), 1-12.

APPENDIX

LIST OF WORD EXPERT OPERATIONS

Action Node:

NEXT (n) : Branch the process control to node n

NEW_CONCEPT (T) : Construct a new concept with type T, post this new concept in the control state description table, set the field value of the NEW_CONCEPT to OPEN.

OLD_CONCEPT (X) : Refer to the previously constructed concept, the value of X can be either rw, which means concept of its right word, or lw, which means concept of its left word.

```
PEEK  (rw)     :  Getting the name of the expert at its
  (name1  n1)     right neighbor without executing that
  (name2  n2)     particular expert. The process control
  ( ....... )     branches to one of node ni on the basis
  ( *     nk)     of word expert name. If none is seem
                  applicable, the control branches to nk.
```

Question Node:

```
MC  CONCEPT_NAME     :  Make a reference to the concept
  (concept1  n1)        specified as CONCEPT_NAME and
  (concept2  n2)        check the nature of that concept
  ( .......... )        with possibilities listed in a
  ( *        nk)        menu. The expert branches on
                        this result.

C  FIELD_NAME        :  Probe the control state description
  (value1  n1)          table and check the value of
  (value2  n2)          specified FIELD_NAME, the expert
  ( ........ )          branches to a specified node based
  ( *      nk)          on the value of FIELD_NAME.

Y/N  FIELD_NAME   VALUE  :  Check the value of specified
  (n1)              FIELD_NAME in the control state
  (n2)              description table. If they are the
                    same then the expert branches to node
                    n1, otherwise the expert branches to
                    n2.
```

Suspend Node:

condition : Set up the condition for the expert to suspend itself.

RESUME (action) : Specifies the action that is needed to be done upon the expert's reawakening.

NEXT n : Branch to a specified node n.

Terminal Node:

CONCEPT_TYPE CONCEPT : Set up the final construction of concept currently in active.

VITA

Vachira Terayanont

Candidate for the Degree of

Doctor of Philosophy

Thesis: SEMANTIC ANALYSIS OF THE THAI LANGUAGE

Major Field: Computing and Information Science

Biographical:

Personal Data: Born in Thailand, March 25, 1958, the daughter of Dr. Sawang and Dr. Voranuj Terayanont.

Education: Graduated from Triam Udom Suksa, Bangkok, Thailand, in 1975; received Bachelor of Science Degree in Chemical Engineering from Chulalongkorn University, Bangkok, Thailand, in 1979; received Master of Science Degree in Engineering Management from University of Missouri at Rolla in December, 1982; completed requirements for the Doctor of Philosophy at Oklahoma State University in July, 1988.

Professional Experience: Scientist, National Energy Administration, 1979 to 1981; Teaching Assistant, Department of Computing and Information Sciences, Oklahoma State University, August, 1984, to May, 1987.