A UNIFICATION-BASED FOCUS SYSTEM FOR PROSODIC ANALYSIS

by

Lalita Narupiyakul

Submitted in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy

at

Dalhousie University
Halifax, Nova Scotia
August 2007

Your file Votre référence
ISBN: 978-0-494-31502-6
Our file Notre référence
ISBN: 978-0-494-31502-6

DALHOUSIE UNIVERSITY

To comply with the Canadian Privacy Act the National Library of Canada has requested that the following pages be removed from this copy of the thesis:

Preliminary Pages
- Examiners Signature Page (pii)
- Dalhousie Library Copyright Agreement (piii)

Appendices
- Copyright Releases (if applicable)

DEDICATION

To My Mother,

For Your Love, and Inspiration

Table of Contents

List of Tables

List of Figures

Abstract

A speaker's utterance may convey different meanings to a hearer than what the speaker intended. Such ambiguities can be resolved by emphasizing accents at different positions. In human communication, the utterances are emphasized at a focus part to distinguish the important content and reduce ambiguity in the utterance.

In our Focus-to-Emphasize Tone (FET) system, we determine how the speaker's utterances are influenced by foci and speaker's intention. The relationships of focus information, speaker's intention and prosodic phenomena are investigated to recognize the intonation patterns and annotate the sentence with prosodic marks. The thesis consists of three parts: analysis, design and implementation, and evaluation of the FET system. The first part is the FET analysis. The relationships between focus, speaker's intention and prosody are analyzed. We consider how to define the intonation patterns using the speaker's intention and find which parts of the sentence serve as the focus parts.

In the second section, the design of the FET structure and subgrammar is developed using the information of focus, speaker's intention and prosody. Our FET structure and subgrammar are unification-based formalisms and can be used with the LKB system, which is an HPSG parsing system. The FET subgrammar includes typed constraints, a set of focus words, grammar rules, typed hierarchy, and typed feature structures for focus, speaker's intention and prosodic features. We implement the FET system as a proof-of-concept system, developed using the LKB system with our FET subgrammar.

The last part is the evaluation of the FET system including (i) the perceptual evaluation of the utterances conveying focus, and (ii) the evaluation of the prosodic annotation. The perceptual evaluation is performed by a listening test, in which participants must listen to different utterances of the same sentence and select a sound utterance that make the most sense from multiple choice questions in a dialogue. The CMU communicator dataset is used in the second evaluations and the results are discussed with respect to the performance of the FET system.

List of Abbreviations and Symbols Used

AVM	Attribute-Value Matrix
BDI	Belief and Design Inference
CART	Classification and Regression Tree
CMU	Carnegie Mellon University
CMU-COM	CMU Communicator
CSLI	Center for the Study of Language and Information
CTS	Concept-to-Speech
EP	Elementary Predication
ERG	LinGO English Resource Grammar
FC	Focus Content
FCS	Focus Content Scoping
FET	Focus to Emphasize Tone
f_0	fundamental frequency
HPSG	Head-Driven Phrase Structure Grammar
Hz	Hertz
IPA	International Phonetic Alphabets
JSML	Java Speech Marked-up Language
LKB	Linguist Knowledge Building
MRS	Minimal Recursive Semantic
NLG	Natural Language Generation

NLP	Natural Language Processing
OSULL	Ohio State University Linguistics Laboratory
PAG	Prosodic Annotation and Generation
PIH	Prosodic Isomorphism Hypothesis
Praat	Phonetic Analysis A Transcription
PSOLA	Pitch Synchronize Overlap Adding
SG	Speech Generation
SL	Spoken Language
SLG	Spoken Language Generation
SLU	Spoken Language Understanding
SSML	Speech Synthesis Marked-up Language
TFS	Typed Feature Structure
ToBI	Tone and Break Indexing
TTS	Text-to-Speech

Acknowledgements

I wish to thank my supervisors: Dr. Nick Cercone, Dr. Vlado Keselj, and Dr. Booncharoen Sirinaovakul. Dr. Nick Cercone taught me many things both in and out of the books. He enjoys working together with students and helps students to develop their theses. Dr. Cercone is not a supervisor who guides or provides all knowledge to students. He listens to students ideas. He is always ready to learn an innovative knowledge from students and then he teaches students how to expand their knowledge. I also thank him to give me an opportunity to conduct research and study in Canada and for financial support during my studies. I thank to Dr. Vlado Keselj. He is a very generous person. His discipline is unbelievable. He can manage a lot of works both research works and academic duties very well. He does not only advices students but also works beside students. He understands students' problems and helps us solve problems. I thank to Dr. Booncharoen Sirinaovakul. He is the first person who taught me how to conduct research. He gave me an opportunity to study PhD in Thailand. He is very kind and has a teacher's soul. He is not a kind of a teacher who gives students any answers but he would like to see his student searching and exploring answers by themselves. I wish to thank my committee members: Dr. T. Pattabhiraman, Dr. Denis Riordan for their valuable suggestions to improve my thesis.

I would like to thank many friends and colleges for all their supports during my study. Specially I thank Hathai who started journey in Canada with me. Her supports to me are countless. I am very appreciate for everything that she did to me. I wish to thank Jiye who taught me everything about Canada and how to survive in Canada. She is so generous to me. I would like to thank Anand, Tee, Tony, Daan, Kate, Asad, Vivek, Tew, and Xiaofen for their encouragements and helps for the past five years. My thanks also go to my friends at Dalhousie University and KMUTT for their supports and happiness moments. I would like to thank P'Ped and P'Kaew for their helps and advices and for being like my sisters. I wish to thank staff members at Dalhousie University for their helps.

I am sincerely thankful to (i) Royal Golden Jubilee PhD Programm, Thailand

Research Fund, Thailand (ii) Precarn, NSERC, Canada, and (iii) Faculty of Graduate Studies Scholarship, Dalhousie University and (iv) Faculty of Computer Science, Dalhousie University for their financial supports during my PhD study.

Finally, I wish to thank the most important person in my life, my mother, who makes me a better person in everyday. She is my inspiration to study PhD. I wish to thank my brothers who encourage me and back up me for my whole life.

Chapter 1

Introduction

1.1 Thesis Statement

A speaker's intention when using a sentence dictates how a speaker utters that sentence. The concept of focus in a speech utterance is investigated, especially the relationship between focus and speaker's intention, to permit the diversity of prosodic generation. We propose the Focus to Emphasize Tone (FET) grammar which is a unification-based formalism as an approach to address this problem. The grammar is implemented in our FET system, which analyzes a sentence to identify focus components, to find the intonation patterns, and to produce tone marks as a result. The FET system is a proof-of-concept system, which is validated using a corpus from a travel reservation domain.

1.2 Motivation

A speaker's utterance may convey different meanings to a hearer. Such ambiguities can be resolved by emphasizing accents in different positions. For example, the meaning of the sentence "Tom will win?" with high tone at the end of sentence is that the speaker wonders whether "Tom will win", while "Tom will win" with a low tone at the end of sentence means that the speaker is confident that "Tom will not lose".

For the same sentence, the hearer can recognize the meanings of the different utterances intuitively but the computer cannot. Presently, a system for computer generated speech can annotate the prosodic marks and generate the speaker utterance for a sentence without determining the relationship between the meaning and utterance.

To improve the capability of the computer generated speech, the speaker's utterance generated by the computer must be synthesized based on the speaker's intention and the focus content that hearer can recognize from the utterance. The computer needs to know what focus part the speaker wishes and what is the speaker's intention; i.e., asking or confirming the information. In a sentence, focus is analyzed to assign

suitable accents at the correct positions of a speaker utterance so that this utterance can convey a precise meaning to hearer. A speaker's intentions must be revealed to define the intonation pattern. The relationship between speaker's intentions and focus information is used to define which parts of the sentence serve as the focus parts.

Consequently, the computer-generated speech system will annotate the prosodic marks, which are used to emphasize the focus part in a sentence, and produce a speaker utterance that conveys the content the speaker wants a hearer to recognize. For example, if Tom (speaker) say to Mary (hearer) that "Kim bought a flower?", then there are two possibilities that Mary may understand: (i) "Tom wants to confirm who bought a flower" or (ii) "Tom wants to confirm what does Kim buy". To reduce the ambiguity of speaker utterances, the speaker "Tom" must emphasize his utterance by focusing at the content that he wants the hearer to recognize. In this example, if the speaker Tom focuses at "Kim" (a person who bought a flower), then the sentence that Tom asks Mary, will be "$[\mathbf{Kim}]_F$ bought a flower". If Tom focuses at "a flower" (a thing that Kim bought) then the sentence will be "Kim bought $[\mathbf{a\ Flower}]_F$". The words in square brackets with the subscript F represent a focus part.

The advantages of the computer-generated speech that generates the utterance with focus by emphasizing tone are: (i) reducing the ambiguity of the meaning of an utterance, and (ii) improving the perception quality and increasing the clarification of an utterance so that hearer can recognize the focus content from the speaker's utterance easily.

1.3 Objectives

The main objective in this thesis is to analyze the relationships between focus and speaker's intention information, find the prosodic patterns for these relationships, and annotate the prosodic marks as the result. A unification-based formalism is used in our analysis to design the feature structures, grammar rules, lexicon entries and their constraints. The details of our objectives are composed of three parts: analysis of the relationships between focus and prosody, designing typed feature structures and constructing the environment for our analysis, and evaluation of our system.

Part I: Analysis The relationships between focus and prosody are analyzed following two aspects that require our investigation.

- Analyzing the relationships between focus and speaker's intention features with respect to prosodic phenomena. The syntactic and semantic features need to be considered before finding the focus components. The relationships between focus and speaker's intention must be investigated to discover how they can influence or are involved in the prosodic parameters in each prosodic phenomenon. This investigation is used to improve the diversity of prosodic generation for a sentence.

- Investigating how the speaker's utterances are influenced by a speaker's intention. We must investigate how to define the intonation patterns from the different speaker's intentions and find which parts of the sentence are emphasized or serve as the focus parts.

Part II: Design and Implementation Designing feature structures needs to cover the information of focus, speaker's intention, and prosodic information. These features must be considered since they have a strong effect on prosody. Furthermore, we design the feature structure to represent the relationships between focus and speaker's intentions and prosodic patterns. For constructing the FET environments, focus words, grammar rules, type hierarchy, and type constraints are designed in form of unification-based formalism. This environment is used to parse a sentence, to analyze the informational structure, including focus and speaker's intentions, to find the prosodic patterns and to annotate the prosody marks on a sentence as a result.

Part III: Evaluation In this thesis, the evaluation consists of two main parts: perceptual evaluation and the evaluation of prosodic annotation. For perceptual evaluation, a listening test is performed to evaluate whether listeners can recognize focus conveyed by emphasizing tone. To evaluate prosodic annotation, we compare the annotation of our system with the annotation from the CMU Communicator (CMU-COM) dataset which is our reference.

1.4 Contributions

Our main contribution is the design of Focus to Emphasize Tone (FET) system and its environment. The FET system is a new and unique system of focus analysis for prosodic generation. We design a set of constraints and relationships of focus, speaker's intentions and prosody in a unification-based formalism which is the main novelty compared to other existing approaches in spoken language generation area. This FET structure is integrated into a unification-based system for focus analysis and is used for the prosodic annotation. Consequently, another contribution is the prosodic modification system which is used to modify prosody of synthetic speech depending on its annotation. A more detailed list of our contributions is introduced below.

- **Designing the FET structure and its environment for prosodic analysis.** We analyze the relationships between focus with speech's intention and tones and design the FET structure to represent their relationships. The FET structure is a kind of knowledge representation between focus and prosodic domains. We build the FET grammar for the unification-based parser. Building the FET environment also includes (i) assigning the Typed Feature Structures (TFSs) of focus, and speech act features, and (ii) designing the type hierarchy, the FET typed constraints, focus word structures and grammar rules for the FET analysis. For the prosodic annotation, the FET grammar is used to parse a sentence for labeling the prosodic marks depending on focus and speaker's intention. The FET structure is different from the other constraint-based approaches proposed in [1] and [2]. On one hand, they developed their constraints and feature structures based on the matrix trees, which is a isomorphism of syntactic tree, and theme and rheme theory. On the other hand, the FET structure is designed with a regard to the focus and speaker's intention information that the focus structure contains the information of what are the focus parts, focus types, and etc.

- **Implementations.** We implement the FET subgrammar which is compatible with the unification-based formalism for the Linguist Knowledge Building (LKB) system. The LKB system with the FET subgrammar is used for our

FET analysis. It parses a sentence with the focus information and generates the FET structure with tone annotation as a result. For the FET subgrammar, we design focus words, the FET typed hierarchy, the focus and prosodic structures, the FET constraints, and rules corresponding to the LKB system.

- **Reducing ambiguity in speech utterances.** This ambiguity is reduced by increasing the diversity of prosodic generation for an individual sentence. For the same sentence, the FET system generates different intonation patterns depending on foci and speaker's intention. For example, interrogative and affirmative sentences of the same sentence are controlled by speaker's utterances. Since speech utterance, which emphasizes tones at the focus parts, can convey the speaker's intention with prosody to hearer, then this utterance, generated by the FET system, can reduce ambiguity to hearer.

- **Improving the content recognition rate for hearer.** Based on our FET system, if the synthetic speech is emphasized by tone at the focus parts, this utterance can convey the crucial content to hearer and is easy to recognize by hearer. For the application, if this system can be applied to "the automatic reminding system" or "automatic telephone operator", then hearer can recognize the content in speech easier specially for the low sound quality environment such as the telephone sound.

- **Perceptual and annotation evaluation of the FET system.** We evaluate whether listeners can recognize foci by emphasizing prosody. We perform the listening test and four dialogues from travel reservation domain are prepared for our test. At each test sentence in a dialogue, four different utterances are modified by the FET system and these different utterances represent the sound with different focus parts emphasizing by prosody. Listeners must select their most preference that make the most sense in the dialogue.

1.5 Outline of Thesis

In this thesis, we explain the basic concept of prosodic generation and survey the the constraint-based approach for the prosodic annotation. Designing of the FET feature

structure and the constraints to assign the prosodic patterns are described. Below is the outline of the thesis.

In chapter 2, a survey of prosodic generation is introduced including the prosodic representation and the approaches for prosodic generation.

Chapter 3 is the discussion of the problems that we want to solve in this thesis. The discussion is classified into two aspects of information domain to explain the prosodic phenomena. Furthermore the performance, data preparation, and feature selection issues are explained in this chapter.

From chapter 4 to chapter 6, designing the FET system is described including the preparation process, the FET analysis, and analysis of the relationships of focus with speech acts and prosodic features to find the prosodic patterns.

Chapter 4 is preprocessing step. The LKB system with the English Resource Grammar (ERG) [3] parses a sentence. The LKB system analyzes the syntactic and semantic structures and generates the Minimal Recursive Semantic (MRS) [4] representation. This step occurs before invoking the FET system. we scan the MRS structure and collect any components and their relations among them obtained from the preprocessing step. We select only required information, such as sentence mood, from the MRS representation, assign a speech act code referring to a main verb of a sentence, and transform the MRS structure to a set of focus words. These focus words are an input to the focus information analysis in the FET system.

Chapter 5 is the FET analysis. This chapter is the explanation of focus phenomena in a sentence. The goal of our analysis is to determine what are the focus parts and their components including the focus types, focus groups and so on.

In chapter 6, the relationships of the speech acts and tone are investigated and how these relationships are associated to identify the prosodic patterns for prosodic annotation. The categories of speech acts and the prosodic representation system are introduced.

For the last step, chapter 7 is the postprocessing process. We extract words and their prosodic marks as Tone and Break Index (ToBI) representations [5] from the FET structure. The extracting system scans the FET structure, and extracts only our required prosodic fields. These fields are a set of words and their tone marks for a sentence. We use the set of words with tone marks to modify synthetic speech, which

is generated by speech synthesis. We use the PRAAT [6] with our module to modify the prosody of the synthetic speech for a sentence. Our output is an audio file of the sentence with modified prosody. Modifying prosody follows the tone marks which are analyzed by the FET system.

In chapter 8, an example of our implementation is used to demonstrate our FET grammar for the LKB system [3], including the generation of a set of focus words, explanation of the FET environment.

Evaluation of our system is described in chapter 9. Two main evaluations are conducted: (i) the perceptual evaluation and (ii) the evaluation of prosodic annotation. The design of experiment and the experimental results for both evaluations are reported and concluded in this chapter.

Chapter 10 is the conclusion and future works. The FET system and its details are summarized including discussion about limitations of system. Finally, the future work is introduced at the end of this chapter.

Our research presented in the thesis has been published in part in conference proceedings and journal. The FET system, which is the main part in this thesis in chapter 4-8, and the evaluation of prosodic annotation in chapter 9 will appear in [7]. In chapter 4-5, the focus content structure and FET analysis are described in [8]. In chapter 5-7, the relationships of speech acts and prosody for each focus part, and summary of the FET implementation for the LKB system appear in [9].

Chapter 2

Background

The prosodic generation has been used mostly in the Spoken Language Generation (SLG) and Text-to-Speech (TTS) systems. Since the prosody conveys the discourse information and reduces the ambiguity in speech, a listener can better recognize precise meaning from the speech enriched with prosodic information. In this chapter, the existing prosodic models for SLG are described including the prosodic representation, the prosodic analysis, the approaches for prosodic generation and annotation, and the LKB [3] unification-based system for prosodic generation.

An overview of SLG is explained in section 2.1. Two general domains for spoken language system are engineering and linguistic domains, as described in section 2.1.1. The studies of SLG are reviewed in section 2.1.2. In SLG, the prosodic generation begins in the linguistic domain. Fujisaki [10] analyzes the computational module to describe prosodic feature structures and use them for prosodic generation in the engineering domain. An ordinary SLG system is composed of four main parts: discourse planning, surface realization, prosodic generation, and waveform generation. The applications, involving the prosodic generation of SLG, are introduced in section 2.1.3.

The roles of the prosodic models are explained and the prosodic representations are described in section 2.2.1. There are several levels of prosodic analysis and representation. For example, the phoneme is the smallest unit of sound utterance and it is represented by a phonetic symbol. The tone representation occurs mostly at syllable and word levels. One of the well-known prosodic annotation systems is Tone and Break Indexing (ToBI) system. ToBI is developed by Ohio State University Linguistics Laboratory (OSULL). The details of the system are described in section 2.2.2.

The prosodic features of speech are influenced by syntactic, semantic, focus, and speaker's intention information. The relationships of focus, speaker's intention, and prosodic features are discussed in section 2.3. Baart [11] and Davis and Hirschberg [12] described their theories, which explore these relationships. For example, Focus-Accent theory [11] and Giveness theory [12] are based on focus analysis (see section 2.3.1). Some constraint-based approaches, used to explain the relationship between

speaker's intention and prosodic structure, are introduced in section 2.3.2.

In the published work up to date, there are three different approaches to prosodic generation for SLG: machine learning, template-based, and unification-based approaches. They are reported in section 2.4. We use a unification-based approach based on the LKB system, which is introduced in section 2.5.

2.1 Overview of Spoken Language Generation

Basically, SLG is the integration between Natural Language Generation (NLG) and speech synthesis. NLG uses linguistic knowledge to analyze the context, generate the feature structures such as semantic and syntactic structures, and construct the sentences or context for communication. The speech synthesis using the signal processing techniques generates the human speech waveform of the sentences or context constructing from the NLG system. In this section, the general SLG is explained including the historical background of SLG, overview of the SLG processes, and the applications of the SLG.

2.1.1 Historical Background of Spoken Language Generation

Engineering and linguistic paradigms predominate spoken language research, where the objective is to speak and generate sound. Engineering paradigm considers how machines speak and synthesize sound using signal processing techniques, language analysis, algorithm and statistics to select speech units. The linguistic approach considers intonation, prosody of spoken language, and phonological representation. It is used to describe how human utters and what is the knowledge in human sound. Using representation or specific symbols is a method to annotate spoken language and a basic way to understand the human utterance.

Engineering Domain

One of the early system was a speech synthesis system for a "read aloud" machine. Such a system requires computation that computers of that time were incapable of performing, or the synthesis algorithms were not as sophisticated as the computer performance available required. Later, Klatt [13] proposed the signal processing

technique for a TTS system, called linear prediction coding synthesis. The system synthesizes each English word as low-quality synthetic sound. Few years later, the performance of computers and memory capacities were further developed. Moulines and Charpentier [14] proposed the concatenative synthesis system. This system utilized the high memory capacity development with low computational algorithm, called Pitch Synchronize Overlap Adding (PSOLA) [15]. It merges small units of human speech signals in corpus to generate sounds of word utterances. The output quality is moderate but unnatural. To increase the quality of speech generation, text processing was assembled into TTS system to support the correctness of transcription. Recently, an important speech generation system, called *Unit Selection System*, was introduced by Black [16]. Unit selection algorithms for speech generation can produce high quality synthetic sounds. It gathers huge speech corpus of recorded real human voice from monologues and employs Classification and Regression Tree (CART) algorithm [17] to select the proper sound units for generating synthetic sounds. However, a limitation of this system is that it is a domain-dependent system. Due to developing performance of speech generation to the present, speech generation is applied in many applications such as Concept-to-Speech (CTS) which is the integrated system between Speech Generation (SG) and NLG to improve the naturalness of synthetic speech. A summary of text-to-speech system is introduced by Dutoit [18]. Recently, an expressive speech synthesis system has been developed using high quality speech corpora. These speech corpora are generated using hidden markov model algorithm proposed by Pitrelli et al. [19].

Linguistic Domain

Studying spoken language in the linguistic domain is to explore how to describe human utterance based on the knowledge representation. Silverman et al. [5] investigates the patterns of human utterances and uses phonetic and prosodic symbols to represent each utterance unit. In spoken language, ToBI model is a representation of prosodic model, developed by Silverman et al. [5]. It is a well-known model for a standard labeling for English prosody. It is also frequently used with focus analysis to represent intonation in sentences. An example of focus analysis is given–new information theory, reported in [20, 21]. This theory defines given or new entities in discourse context.

Another example is *discourse segmentation and hierarchy* [22]. This method is based on two properties: purpose and intention. To consider the information status, *Focus-Accent Theory*, proposed by [11, 23], is frequently used to find focus components in speech utterance. *Salience or Accessibility* [24] determines the focus content in a discourse hierarchy. Following this analysis, the focus information is considered to define the prosodic and intonation patterns for a human utterance in SLG.

An interesting example of linguistic analysis is intonation modification called *Fujisaki's Model* [25], which is a well-known method. This method is a continuous intonation modification for pitch contour and it is a semi-automatic modification. It is based on prosodic computation and signal processing. This method modifies the intonation patterns of synthetic sounds by manipulating pitch contours.

2.1.2 Introduction to Spoken Language Generation System

In this section, language generation is described in detail. There are two considerations for SLG: "What to say" and "How to say it" [26]. To generate a spoken language, "What to say" is the content that we need to communicate to other people and it is called *Content Planning.* Content planning is a module in discourse planing which is composed of the decision making of content structures and selecting the discourse concept. It is a method to describe discourse structure. This structure is a hierarchy of discourse relations including rhetorical relations and discourse status.

"How to say it" is about considerations of sentence generation, following the content planning, that we want to communicate to other people. *Sentence Planning* creates the response sentence belonging to the discourse content. Templates for sentence structures are a part of sentence planning working with structured grammar system. It selects the syntactic structure, sentence scoping and lexical semantic structure for generating response sentence.

Surface Realization [27], based on constraint-based formalism, interprets information obtained from sentence planning. It generates feature structures including semantic role, morphology, functional word structure, and so on.

Prosodic Generation utilizes the linguistic features, including semantic and syntactic features from generated sentence structure. Prosodic generation creates the prosodic feature structures in form of an enriched prosodic representation. These

prosodic features are annotated as structured prosodic representations which correspond to discourse, and express the speaker's intention in an utterance. Using this representation, the system can build naturalness of sound by modifying prosodic and intonation parameters. Furthermore, speech acts and speaking styles of speaker are considered in prosodic generation as well. The speaking style includes the characteristics of speaker, such as gender, and age. The style also captures emotions such as anger, and sadness. The system modifies the prosodic features depending on the speaking style. It will attach to the output an enriched prosodic representation.

Following language generation, the output is interpreted to informational structures of language and prosodic structured representation. These representations are sent to speech generation system. The speech generation employs unit selection system [16] to generate synthetic sounds.

2.1.3 Applications for Spoken Language Generation

There are many applications for SLG system. Some obvious applications for SLG system are introduced and are shown how these applications can support human activities. Many research groups propose new techniques for spoken language generation. Four interesting applications are MAGIC, ILEX Museum Guide, GoalGetter, and Jupiter Weather Information System.

MAGIC is a multimedia presentation generation system for cardiac intensive care patient. It is used for producing spoken language presentation of patient status in multimedia environment using patient's record in a large medical corpus. In MAGIC system, developers employ CTS system which provide speech output allowing a hand free and eye free communication system.

ILEX Museum Guide is an NLG system built to serve as a museum guide. It uses a database of museum exhibits which contain variety of information about each exhibit. Rather than produce a canned description of a given exhibit, ILEX is intelligent in that it delivers unique descriptions of objects depending on a number of contextual factors. ILEX keeps track of which exhibits have already been seen, and hence when viewing a room of Roman swords, the system only gives background information for the first exhibit. As the visitor moves around the exhibits, only the particular details of each exhibit are explained, and these are often contrasted with previous exhibits

GoalGetter is an automatic sport reporter. It is data-to-speech system which generates spoken reports of soccer matches in Dutch. This application is developed by Klabbers et al. [28]. The system summaries a soccer match taking tabular information about the match as input. The system selects appropriate content and sentence structures. It employs a slot-filler technique to generate complete sentences. The speech generation in data-to-speech system generates the spoken sound as output. Two interesting systems in the same application domain as GoalGetter are SOCCER [29] and MIKE [30]. They generate commentaries, which are spoken descriptions of image sequences of soccer scenes.

Jupiter Weather information system [31] is developed by the spoken language group at MIT. In response to spoken user queries, the system finds web-based weather information systems, analyzes their content and generates a suitable reply. For this domain, they used 2000 typical messages from the system as training data. Within these 200 sentences, there were about 600 unique words. The majority of vocabulary items are place's names, and while the training data covers the names of the most frequently requested places, new names often occur, and hence the synthesis component must be able to handle this.

Furthermore there are a lot of applications using SLG. For example, automatic calling center for flight reservation is the interactive dialog system between human and machine. It uses both speech recognition and speech generation to communicate with human. The system will ask you some questions about your information to reserve your flight and book the ticket for you. There are various systems, developed for this application. For example, VoiceXML [32] is frequently used with dialog system. From different domains, automatic information systems are applied for train and bus schedule information system, medical information system and so on.

2.2 The Role of Prosodic Analysis in Spoken Langusge Generation

The improvement of speech quality in spoken language generation is partially based on the prosodic analysis by recognizing the patterns of prosodic features such as pitch and loudness. The prosodic features control the accentuation, boundaries, and loudness of spoken language. To develop SLG, the prosodic features are analyzed to improve the speech quality for conveying the meaning to listener. Linguists have

studied the prosodic phenomena in SLG between the acoustic features and prosodic features [33]. They developed annotation systems to describe prosody in acoustic state and to represent prosodic events using a set of prosodic symbols. Using an annotation system, the prosodic phenomena are represented by prosodic symbols to describe the prosodic patterns. Based on phonology, the prosodic representation is a part of phonological representation. The phonological representation is used to describe the units of sounds in text.

2.2.1 Phonological Representation

Two main representations in phonology are the phonetic and prosodic representations. The phonetic representation consists of a set of phonetic symbols, such that each symbol represents a sound unit. The smallest unit is called a phoneme which is a single sound of one character. The prosodic representation consists of symbols representing accent, stress, pause, and etc. For instance, tone marks are prosodic representation of the pitch contour on a syllable or a word, and the shapes of the pitch contours are usually used to describe accent and stress on the units of sound.

Phonetic Representation is a representation of human pronunciation for any particular language. There is an international standard for phonetic representation. International Phonetic Alphabets (IPA) system, reported by Association [34], represents a minimal set of sound units as phonemes for human speech. IPA is employed for annotation of phonemes for each alphabet as a phonetic transcription.

Prosodic Representation is the representation of intonation and duration in human speech. The prosodic representation is important for machine understanding in SLG system. Informational structure can influence prosodic features [33] and interpret them into a prosodic state for SLG. To modify prosody, some linguistic knowledge can affect acoustic features such as fundamental frequency (f_0), duration, and intensity. Following [33], prosodic representation conveys prosodic knowledge and can be interpreted into acoustic information. The representation can be divided into two primary methods.

In the first method, enriched and annotated text is represented using markup languages or annotation schemes such as Speech Synthesis Marked-up Language (SSML) [35] and Java Speech Marked-up Language (JSML) [36]. The example of SSML is shown in figure 2.1. SSML components are declared at the initial line, such as language and version, while pitch contour is defined for text "good morning" at the next line. The prosodic markup language is the markup language of prosodic features. The values of these features, such as duration and pitch, are modified to control the frequency and time parameters of the speech waveform. For example in this figure, the initial of prosodic contour (*prosody contour*) is assigned by (0%, +20Hz) and it means that the frequency is 20% higher than the original sound at the initial of the waveform.

The second method is the symbolic representation. The intonation annotation system, such as ToBI system, can exhibit prosodic events and be used to generate prosody following designed prosodic annotation.

```
<speak version="1.0" xmlns="http://www.w3.org/2001/10/synthesis"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xsi:schemaLocation="http://www.w3.org/2001/10/synthesis
            http://www.w3.org/TR/speech-synthesis/synthesis.xsd"
      xml:lang="en-US">
 <prosody contour="(0%,+20Hz)(10%,+30%)(40%,+10Hz)">
  good morning
 </prosody>
</speak>
```

Figure 2.1: Speech synthesis marked up language

2.2.2 Prosodic Labeling System Based on ToBI Framework

ToBI framework, proposed by Silverman et al. [5], is used to annotate the prosody including accent and boundary. The ToBI system is composed of four parallel tiers: orthographic tier, break index tier, tonal tier, and miscellaneous tier. In each tier, the set of symbols represents the prosodic components over the sound units. These units consist of a human speech utterance and rely on the time domain.

The Orthographic Tier

The orthographic tier is used to keep the word labeling of the orthographic transcription. In this tier, each word is segmented and is marked with the start and end times of the word.

Break Index Tier

Break index represents silence boundaries and rate them with the proper degree of juncture. Break index levels are used to control the silence between each pair of words. These levels correspond to the tone marks in tone tier to control the intonation over orthographic words. Five break indices are defined as levels from 0 to 4 as described below.

- **0** is no silence boundary or no break.
- **1** is intruded break mostly between words. It is the short break.
- **2** is a strong pause between words.
- **3** is used for the intermediate intonation phrase boundary. This index is marked between phrases, such as between noun phrase and preposition phrase, or after comma and conjunction such as "and", "or", and so on.
- **4** is used for the full intonation phrase boundary. This index is marked at the boundary tone and at the end of the sentence.

Tonal Tier

In this tier, the tones in the utterance are transcribed following the pitch contours. The pitch contour of tones can be represented with the set of tone marks. These marks are less abstract than the pitch contour. Based on the pitch events associated with intonational boundaries in a sentence, two primary types of tone marks are assigned: phrasal tone and pitch accent. These tone marks are low tone (L), high tone (H), and the combination between low and high tones. These marks can be included with the symbols of intonation boundaries such as full or intermediate boundaries.

Phrasal Tones The phrasal tone marks must be assigned to the intermediate or full intonation phrases. Seven types of phrasal tone marks are described below.

- **L- or H-** are the phrase accents occurring at any intermediate phrase boundary.
- **L% or H%** are the boundary tones occurring at every full intonation phrase boundary.
- **%H** is high initial boundary tone. This tone is marked at the beginning of the phrase. This mark defines the high pitch at the beginning of the utterance.
- **L- L%** is for a full intonation phrase. The low phrase accent is at the end of the phrase followed by the low boundary tone falling to a low point of the speaker's range. This mark is represented as the standard "declarative" contour of American English.
- **L- H%** occurs at the end of full intonation phrase. The low phrase accent is marked at the end of intermediate phrase followed by a high boundary tone called "continuation rise".
- **H- H%** is marked on a full intonation phrase. The high phrase accent is at the intermediate phrase ending followed by the high boundary tone which the pitch rises up to a very high point of the speaker's range. This mark is used for "yes-no question" contour.
- **H- L%** occurs at the full intonation phrase. The high phrase accent of the final intermediate phrase rises from the low boundary tone to a value in the middle of the speaker range.

Pitch Accents Pitch accents are pitch events associated with accented syllables or words. For American English, the ToBI labeling system includes five types of pitch accents.

- **H*** is called "peak accent". The pitch event of this mark occurs in the upper part of the speaker's pitch range near the middle of the pitch range.
- **L***, called "low accent", is in the lowest part of the pitch range.

- **L*+H** is "scooped accent". It is a low tone which is immediately followed by the sharp rise to a peak in the upper part of the pitch range

- **L+H*** is called "rising peak accent". This mark is a high peak tone which is immediately preceded by relatively sharp rise from a valley in the lowest part of the pitch range.

- **H+!H*** makes a step down pitch level from a high pitch.

Miscellaneous Tier

The miscellaneous tier includes comments such as silence, laughter, and so on. The comments are needed for particular transcription purposes. Each mark is attached the beginning and end time of each segment. Figure 2.2 shows intonational annotation on example sentence "I thought it was good". Each speech duration was marked with tone, text, and break level. In this figure, the top frame presents the waveform annotated with words while the bottom frame presents pitch contour with the ToBI annotation of the same sentence.

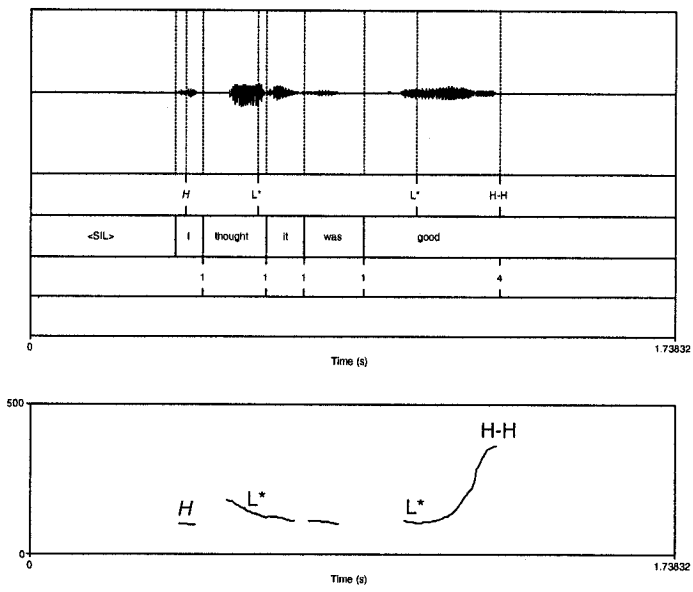


Figure 2.2: ToBI annotation

2.3 Information Structure for Prosodic Generation

Humans can perceive prosody intuitively but the computer cannot. Automatic prosodic generation is a difficult task for computer. There is no standard prosodic representation to describe the prosodic phenomena or events. Beckman and Hirschberg [37] studied how to determine the acoustic features relating to prosodic phenomena. Beckman proposed prosodic representation system based on accents and boundaries. These features are represented by orthographic parameters. The syntactic and semantic information are used to analyze and annotate the prosody. In the information domain, focus and speech acts are considered for prosodic generation.

Focus analysis can be used to define the focus parts in a sentence. The Focus-Accent theory [11] is one of the well-known approached to focus analysis, proposed by Dirksen [23]. It uses a metrical tree in the form of binary branching. The metrical tree is constructed to determine focus and non-focus on the syntactic tree. The concept of this theory is the isomorphism between metrical tree and syntactic tree to scope the focus parts. Focus is defined at the main content in the sentence called *topic* and the *new* information which must never appear before in the dialogue. On the contrary, the background information is insignificant content and is assigned to be non-focus in a sentence. The strong accents are located at focus parts, which are about the topic of the sentence. The weak accents are located at non-focus parts which are the background of the sentence. The discussions of focus analysis, are reported in [12, 38, 39] and are summarized in section 2.3.1.

Another feature that influences prosody is speaker's intention or speech act. Speech acts are used to identify the actions or intention of speaker so that hearer can respond appropriately. Speech act is a speaker's intention in a sentence. The group of verbs, which perform these actions, are called performative verbs [40]. The details of speech act analysis are reported in section 2.3.2

2.3.1 Focus Analysis

In the information structure, the syntactic and semantic features need to be considered to generate the focus structure. When speaker utters a sentence to hearer, the focus information is used to describe which parts of the sentence should be the

main content or the new information and listener should pay attention to these parts. Haji-Abdolhosseini [1] proposed that there is correspondence between informational structure and intonation structure. Focus can be analyzed to improve the prosodic generation. Generally, the concept of focus can be used to express the intonation patterns, which represent prosodic phenomena. The researchers [2, 1] proposed several concepts of focus structures which are based on syntactic and semantic structures.

Focus Analysis Based on Syntactic Structure

The *focus and background* theory analyzes the focus structure from the syntactic information. Following this theory, a sentence is composed of two parts; focus and non-focus (background). Focus projection can define the part of the sentence as a focus (see in figure 2.3). This projection depends on the syntactic structure. The focus part can be represented by the narrow or wide focus. The wide focus covers verb phrase, subject-verb, or a larger syntactic constituent. The rest of them are narrow focus such as noun phrase. For example, the sentence "Tom bought the red car" can have different foci. These foci are inferred from the speaker's intention. In (2.1), the focus of the sentence is at "what did Tom buy" so it is marked at only noun phrase, called the narrow focus. The focus in (2.2) is represented by the wide focus which is the larger syntactic constituent The focus covers the whole verb phrase including verb with noun phrase. For this sentence, the focus is "What did Tom do". The *focus and background* theory is designed for the single focus. To handle more than a single focus, the scopes of focus are defined and semantic information need to be considered for the focus analysis.

$$\text{Tom bought } [\text{the red car}]_F \tag{2.1}$$

$$\text{Tom } [\text{bought the red car}]_F \tag{2.2}$$

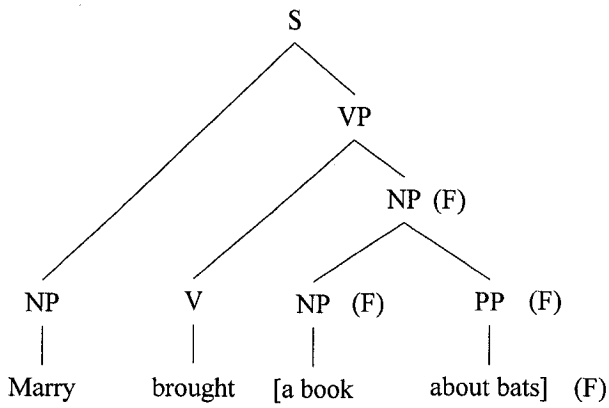


Figure 2.3: Focus and accent tree

Focus Analysis Based on Semantic Structure

The "topic-new information", "focus-accent" or "given-new information" theories [41] analyze foci based on the semantic information. These theories have the same concepts to define the focus parts in a sentence and are used for the dialogue framework. Topic information is a distinguished content in the sentence which can represent the main content. New information is a new content which is concerned with the topic information. For example, in the dialogue below, the topic information is in italic font while new information is in bold font.

Tom: I *got a vintage watch*
Peter: Did you buy *it* from **antique store or jewelry store**?
Tom: **Antique store**. (topic is watch)
Peter: **Where**? (topic is antique store)
Tom: In **Soho**. (topic is antique store)

Using the MRS representation [4], the focus can be assigned according on the semantic structure. For instance, the sentence "Mary bought a flower" is represented by the pattern (2.3). It is labeled on index no 2 representing the focus "What did Mary buy" as shown in the pattern (2.4).

Sentence: Mary bought a flower

MRS:{ 1:Proper_name(x), 2:a{y,3}, 3:flower(y), 4:buy_v2(e,x,y)} (2.3)

Top: {5}, Link: {1}, Tail: {4}, Focus: {2} (2.4)

Top: {5}, Link: {1}, Focus: {2,4} (2.5)

2.3.2 Speech Acts Theory

Two main types of speech acts are direct and indirect speech acts. Austin [42] focused on the direct speech act which is composed of the performative verbs and their related components. Searle and Grice proposed their ideas to explain the indirect speech acts. Searle considered the illocutionary act while Grice is interested in the condition surrounding the interpretation of indirect speech acts. A logical approach, called plan-based approach, to speech act theory is proposed by citetCohen:77. The details of speech act approaches are described below.

Austin's Approach

Austin observed several performative verbs, such as warn and promise, and their compositions. Austin considered that speech acts do not depend on only lexical form. Although an utterance does not contain a performative verb, the act of these verbs (ordering, warning, promising) still be accomplished. For instance, "I order you to shut the door", which can be performed in exactly the same way as "Shut the door". The hearer also realizes this sentence is the instruction without the verb "order". Austin, who defined "speech acts", explained the "performative verbs" utterances and proposed that the utterance of any sentence can be mainly classified to three kinds of acts: locutionary act, illocutionary act, and perlocutionary act. Locutionary act is the utterance of a sentence with particular meaning. Illocutionary act is the utterance of a sentence which performs the act of asking, answering, promising, etc. Perlocutionary act is the production of certain effects of feeling, thought, or actions of addressee in uttering a sentence [42]. The example sentences of these acts are shown below.

Locutionary act:	"You name this ship the Titanic."
Illocutionary act:	"You can't do that." (protesting)
Perlocutionary act:	"You can't do that." (stop the addressee do something.)

Searl's Approach

The narrow explanation of illocutionary act is reported by Searle. Searle considered that speech act is generally represented by the illocutionary act rather than the other acts. The speaker performs the illocutionary act when speaker intends that hearer recognizes speaker's intention to perform the act. The speech acts can be used to describe the speaker's intentional state. Searle classified illocutionary act into five major classes: assertive, directive, commissive, expressive, and declarations.

Assertive:	Committing speaker to something's being the case such as suggesting, swearing, and concluding.
Directive:	Attempt by a speaker to get addressee to do something such as asking, advising, and ordering.
Commissive:	Committing the speaker to some future courses of actions such as promising and planning.
Expressive:	Expressing the psychological state of the speaker about a state of affairs such as apologizing and welcoming
Declaration:	Bring about a different state of the world via the utterance.

Grice's Approach

H. Paul Grice is interested in the conditions surrounding the interpretation of indirect speech acts. He outlines a framework of conversational maxim to explain how a listener could determine speaker intentions.

- Maxim of quantity: Make your contribution as informative as is required (for the current purpose of the exchange) and do not make your contribution more informative than is required.

- Maxim of quality: Do not say what you believe to be false, do not say that if you lack adequate evidence.

- Maxim of relation: Be relevant.
- Maxim of manner: Avoid obscurity of expression, avoid ambiguity, be brief (avoid unnecessary prolixity), be orderly.

An example is described for the maxim of quality. If Tom said "Tim is a fine friend" both Tom and listener Mary know that Tim is not nice to Tom. This situation is the maxim of quality. The statement is ironic. Mary believes the opposite of what he or she literally says.

Plan-Based Approach to Speech Act Theory

Cohen and Perrault [43] considered that AI planning model can be used to define speech act patterns. This model provides the adequacy criteria for speech acts and their components for context dependency. The model involves beliefs, desires, and intention states for conveying the information to hearers. Perrault and Allen [40] employed the Belief, Desire, and Intention (BDI) model to construct the speech act structures. The BDI model was described in [44, 40]. In this approach, the speech act is embedded in context-dependent precondition which is used to declare the effect state of a speech act in the BDI model. The plan-based approach rely on the consistency of truth in form of logical expression. For some situations, such as joking, and kidding, speaker wants the hearer to recognize the indirect meaning which cannot be interpreted directly from the sentence. For example, "Only a millionaire" has direct and indirect meaning depending on the utterance. Designing the speech act structure must be determined based on what is the speaker's intention of the sentence. The BDI model includes the belief, knowledge, and desire patterns as shown in the patterns (2.6), (2.7), and (2.9) respectively. The action schema of BDI model is composed of a set of parameters with constraints about the type of each variable, and three states: precondition, effect, and body described below.

- Preconditions: Condition that must already be true in order to successfully perform the action.
- Effects: Condition that become true as a result of successfully performing the action.

- Body: A set of partially ordered goal states that must be achieved in performing the action.

$$B(S,P) \quad : \quad \text{"S believes the proposition P"} \tag{2.6}$$

$$KNOW(S,P) \quad : \quad \text{"S knows that P"} \tag{2.7}$$

Knowledge is defined as "true belief"

$$KNOW(S,P) \quad \equiv \quad P \wedge B(S,P) \tag{2.8}$$

Theory of Desire (WANT)

$$\text{WANT(S,P)} \quad : \quad \text{if S wants P to be true} \tag{2.9}$$

If ACT is the name of action,

$$W(S, ACT(H)) \quad : \quad \text{"S want H to do ACT"} \tag{2.10}$$

The plan-based speech act system analyzes a speaker's intention that speaker wants hearer to recognize. The system considers who the speaker is and who speaker are talking to, what speaker performs to hearer, what the action in the sentences is. The basic features of speech acts are speaker (S), hearer (H), speech act types (SP), and performative verbs (ACT). The example of action schema in BDI model is shown in patterns (2.11) and (2.12). The further features, that must be considered, are derived from syntactic and semantic features. The syntactic features are used to describe the structure of a sentence while the semantic features describe the meaning of words and their relations in the sentence. Only these features may not be enough for the prosodic analysis. The system requires the focus information to generate the utterances that are suitable to the speaker's intention of the sentence.

$$INFORM(S,H,P) \tag{2.11}$$

$$\begin{aligned} Constraints &: Speaker(S) \wedge Hearer(H) \wedge Proposition(P) \\ Precondition &: Know(S,P) \wedge W(S, INFORM(S,H,P)) \\ Effect &: Know(H,P) \\ Body &: B(H, W(S, Know(H,P))) \end{aligned}$$

$$REQUEST(S,H,ACT) \tag{2.12}$$

$$\begin{aligned} Constraints &: Speaker(S) \wedge Hearer(H) \wedge ACT(A) \wedge H\ is\ agent of ACT \\ Precondition &: W(S, ACT(H)) \\ Effect &: W(H, ACT(H)) \\ Body &: B(H, W(S, ACT(H))) \end{aligned}$$

Speech Acts and Prosody

The prosody can present speech at different tone levels and different durations. The prosodic features can be modified to improve the quality of speech. The prosody coveys the speaker's intentions to hearer. For the same sentence, different prosodies in speaker utterances can be possibly interpreted to different meanings.

Speech act theory is used to analyze speaker intention that speaker wants hearer to recognize. Analyzing speech act theory and their features can help us find the prosodic patterns. In the example sentence below, the different utterances can have different meanings depending on the speech act types. For example, in "Only a millionaire", the first utterance has tone emphasis at "a" in the sentence as shown in (2.13). The speech act type of this utterance is "representative" and it means "not a rich person". Another utterance has tone emphasis at "millionaire" in the sentence as shown in (2.14). The speech act type is "directive" and this utterance means "the rich person". The relationships between speech act and prosody are investigated, so when speech act types change then the prosodic patterns also change.

Only	a	millionaire	
H*	L+H*	L-H%	(2.13)
Only	a	millionaire	
H*		L*+H L-H%	(2.14)

The Schemas for Speaker's Intention

The researchers analyze prosodic features from syntactic, semantic and focus information which are used to define the strong and weak accents in a sentence. The syntactic and semantic features are derived from parsing a sentence. A speech utterance conveys speaker's intention or speech act in that utterance. The speaker assigns a speech act type for each utterance. To analyze prosody from speech acts, the feature structures for prosodic generation need to be designed (see chapter 5). Therefore the LKB parser with LinGO English Resource Grammar (ERG) [45] using Head-Driven Phrase Structure Grammar (HPSG), described in section 2.4.3, is employed to find

syntactic and semantic structures. The plan-based speech act theory [40] is used to define the action schemas based on the illocutionary act. The action schemas for speech act are composed of four parts: speech act type, sentence condition, speaker intention, and hearer perception. Speech act type is presented with a performative verb. Sentence condition must be proposition of sentence and not exclude the performative verbs. Speaker's intention is the state which assigns the performative verb. This state presents the relation between speaker and hearer that "speaker performs an act to hearer". Hearer perception is the state that presents what the hearer reacts to speaker.

For example, the sentence "Where are you leaving from?" has speech act type as "directive" and its performative type is "question". The performative verb of this sentence is "ask". Technically, the performative verbs do not need to appear in the sentence and they can be recognized from the sentence pattern. In this case, this sentence is interrogative sentence that speaker asks information of hearer. The speaker wants to request information from hearer. The reaction of hearer is to answer the question to speaker. The action schemas are illustrated below.

Sentence: "Where are you leaving form?"
S is represent speaker, and 1st person pronoun
H is represent hearer, and 2nd person pronoun

Speech act type:	Directive (question)
Sentence condition:	leave(H, from(X)) ; X is a place
	(Hearer leaves from X)
Speaker intention:	Ask(S, H, leave(H, from(X)));
	(Speaker asks hearer that "hearer leaves from X")
Hearer perception:	Answer(H, leave(H, from(X)))
	(Hearer answers that "hearer leaves from X")

2.4 Prosodic Generation Approaches

This section focuses on investigating the prosodic models for prosodic generation. Generally, there are three main prosodic approaches to compute prosody: template-based systems, machine learning & stochastic systems, and unification-based systems.

2.4.1 Template-Based System

Template-Based System contains a set of sentence structures with the associated prosodic features. The slot-filler technique is employed [46] to annotate tones in form of prosodic representations. Template-based system is frequently used in NLG. It is easy to add new template or design new content by hand. However, a disadvantage of a template-based system is that it is domain dependent, i.e., it is not flexible to apply to other domains. Furthermore it is time-costly to build the system because adding new templates requires significant manual effort. In the template-based method, the moderated features are accents (weak and strong) and boundary (short, medium, and long). In the information domain, this system analyzes the focus and non-focus of the content words and relation between sentences for assigning prosodic feature structure. Roy [47] focused on the content selection which refers to the template content for Spoken Language (SL). His system is a semi-automatic description generation system, which is not very flexible. Theune [46] analyzes contrast in content for an automatic sport reporter. Their system analyzes weak and strong accents in sentences. It is illustrated in figure 2.4 [46]. This figure shows the metrical tree of syntactic structure for an example sentence. S is represented as strong accent and W is represented as weak accent, while F+ means focus word and F- is a non-focus word. However template-based method cannot be used with a new topic content without a template.

2.4.2 Machine Learning and Stochastic-Based Systems

Machine Learning includes the efficient techniques, such as decision tree and rule induction, to analyze the prosodic phenomena and predict the prosodic models. Using a decision tree method, Taylor [48] represents the phonological structures as associated prosodic features, and maps them to the sound units in the speech corpus. This method can reach to the smallest sound units, i.e., phonemes, and employ the unit selection system to generate the speech sound using the concatenation cost to select the best unit from speech corpus. CART [17] is the classification and aggregation algorithm for selecting the sound units from the speech corpus. For SLG, a limitation of decision tree methods is that they can only select or classify speech units, based on the condition of various features. This method is difficult to apply for deep

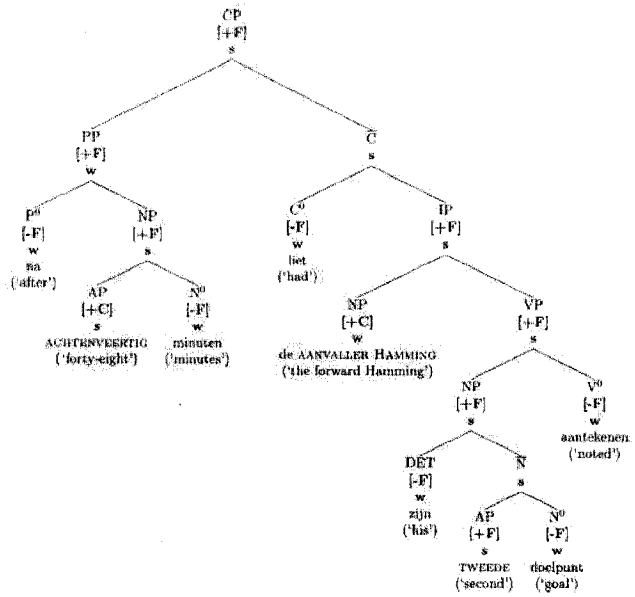


Figure 2.4: Focus tree and focus projection

semantic feature structures due to the complexity of semantic analysis and various semantic features. Another machine learning technique, Pan et al. [49], investigates prosodic modeling using a rule-based system as a semi-automatic learning system to find prosodic model. Her system employs rule generator called Ripper [50], which is a tool to generate a set of rules from linguistic features for finding prosodic modeling. A set of rules expresses the correlation between linguistic and prosodic features. However, the disadvantages of rule-based system for SLG are rule redundancy and the missing value problem of prosodic features in a content.

A Stochastic-Based System employs a statistical model to compute the frequencies of co-occurrence features in sentence and between sentences. These frequencies are derived from the frequency of linguistic features related to prosody for SLG. This system analyzes the content and related contents using statistical methods to find prosodic patterns. Recently, a stochastic-based system for sentence planning (SPot) was proposed by Walker and Rambow [51]. The training process uses Boosting SpoT [52], which is based on a randomized sentence plan generator. It produces a set of candidate sentences. This set is derived from a sentence ranker that is trained with human feedback. Another efficient statistical method using N-grams model, proposed by Oh and Rudnicky [53], is robust and is used often in natural language

processing area. The system calculates the frequency to confirm the information status. The system scores the model and selects the possible prosodic feature structures. Then, the last step is filling in the prepared slot. This system is a fully-automatic system and provides the flexibility to apply to a new domain. However a limitation of this system is that it has no capability to predict new or unseen information.

2.4.3 Unification-Based System

Klien [2] selects the standard HPSG approach for prosodic analysis and representation. He considers the ways to modify phonological attributes which can handle prosodic structures. His work has three steps: (i) representing prosodic structure, (ii) defining a prosodic relation between phonology values and metrical trees and (iii) incorporating the relation into prosodic constraints within a constructional hierarchy. On the other hand, Haji-Abdolhosseini [1] not only employs the standard HPSG approach but also utilizes information structures to generate metrical tree to find prosodic structure. The isomorphism of syntactic structure and prosodic structure was explained in [1], and [2]. It is an interesting aspect and should be expanded to other related structures.

In SLG, not only syntactic feature but also semantic and focus features affect prosodic phenomena. Following the research work mentioned, they use grammatical theories, knowledge representation and the correspondence of different knowledge structures to define prosodic structures. The relationship among these structures are investigated following the grammatical theories based on unification. Information structure is analyzed to find the prosodic structure. The feature structures and type hierarchies of the prosodic models in the unification-based system are compared and described in the next section.

2.4.4 Comparison of Prosodic Models in Unification-Based System

Developing spoken language generation requires the prosodic analysis to improve the quality of synthetic sounds to be natural. The prosodic representation systems, such as ToBI, are proposed to support speech implementation. The syntactic and semantic structures are considered parallel with prosodic structure for the lists of lexicon items. A reason to separate between prosodic and syntactic/semantic structures is to reduce

the complexity of linguistic analysis. In addition, many linguistic systems, such as part of speech tagging, do not need prosodic structures for their analysis. However, the prosodic, semantic, and syntactic information are required for spoken language analysis.

Recently, Klien [2] and Haji-Abdolhosseini [1] studied the relation between syntactic/semantic structures and prosodic structures. Klein focuses on the model of prosodic constituents which are based on syntax. He believes that prosodic structures are related in some ways to syntactic structures. He designs the prosodic constituents for HPSG following the Prosodic Isomorphism Hypothesis (PIH). PIH is the isomorphism between syntactic and prosodic structures. Haji considers a constraint-based approach to describe the information-prosody correspondence in HPSG. He proposed information-based model of prosodic constituency. In his model, the syntactic semantic and prosodic structures are generated in parallel. Each structure requires different constraints to be imposed. On the other hand, Asudeh and Klein [54] do not focus on prosodic structure. They analyze phonological contexts, and consider syntax-phonology interface and prosodic modification in context.

Feature Structures

In the present formulation of HPSG, all signs consist of at least two attributes: *PHON* and *SYNSEM* as shown figure 2.5(a). *PHON* is a list of phoneme strings while *SYNSEM* includes syntactic and semantic information. Klein extended the *PHON* attributes by including lists of prosodic features as shown in figure 2.5(b). His *PHON* attribute has two prosodic types: leaner and metrical tree. Leaner represented a word or phrase which is normally unstressed. Metrical tree is the opposite of leaner. Metrical tree is composed of list of prosodic domain (*DOM*) and Designated Terminal Element (*DTE*) for marking the strong elements. Haji extended Klein's work. He adds *TONE* feature in *PHONE* attribute and prosodic domain was changed to Tone Domain (*T-DOM*). Furthermore, he constructs two more attributes: Intonation Domain (*DOM*) and information structure (*INFO*) which refer to given and new information as shown in figure 2.5(c). New information in term of theme corresponds to rise-fall-rise intonation while given information in term of rheme corresponds to fall intonation. Asudeh included three features into *PHON*, *SEGMENTS*, *PROSODY*

and *PHONOLOGICAL-CONTEXT* (*p-ctxt*) as shown in figure 2.5(d). In his paper, he only described *p-ctxt* feature and construction for phonetic modification.

Type Hierarchy of Prosodic Constituents

Klein's model does not relate to information status of prosodic constituents in the sentence. His model is matched between syntactic and prosodic structures. The prosodic type hierarchy of Klein's model is shown in figure 2.6(a). For Haji's model, prosodic type hierarchy is similar to Klein's prosodic type hierarchy except that Haji's prosodic type hierarchy includes *TONE* feature into the hierarchy as shown in figure 2.6(b).

Klein's type hierarchy of phrasal construction is the cross-classify prosodic phrases under syntactic phrases as shown in figure 2.7(a). Following PIH, Klein assumes that all syntactic phrases can match some prosodic phrases. Even some prosodic phrases cannot correspond to syntactic constituents in fact. On the other hand, Haji's type hierarchy of phrasal construction is not cross-classified because he uses information status and does not need to refer to syntax. Prosodic structure can be defined over the list of domain objects instead of a list of partial prosodic structures. His type hierarchy of phrasal construction is shown in figure 2.7(b).

2.5 Linguistic Knowledge Building System

The LKB system [3] is a grammar and lexicon development environment for using with constraint-based formalism. The LKB system is developed by Center for the Study of Language and Information (CSLI) at Stanford University. Briefly, the LKB system is used for Natural Language Processing (NLP) research such as teaching, parsing, generation of unification-based formalisms. The system is designed to analyze TFSs and is developed to support the HPSG. The other comparable systems are ALE and PAGE based on HPSG. The LKB system includes TFSs and unification, as the operation. The LKB system is a software package for writing linguistic programs, such as building the grammars and lexicons for NLG. Two main advantages of the LKB system are (i) enabling computational linguistics to adopt techniques from theoretical linguistics with minimal reinterpretation and (ii) allowing formal theories to be tested

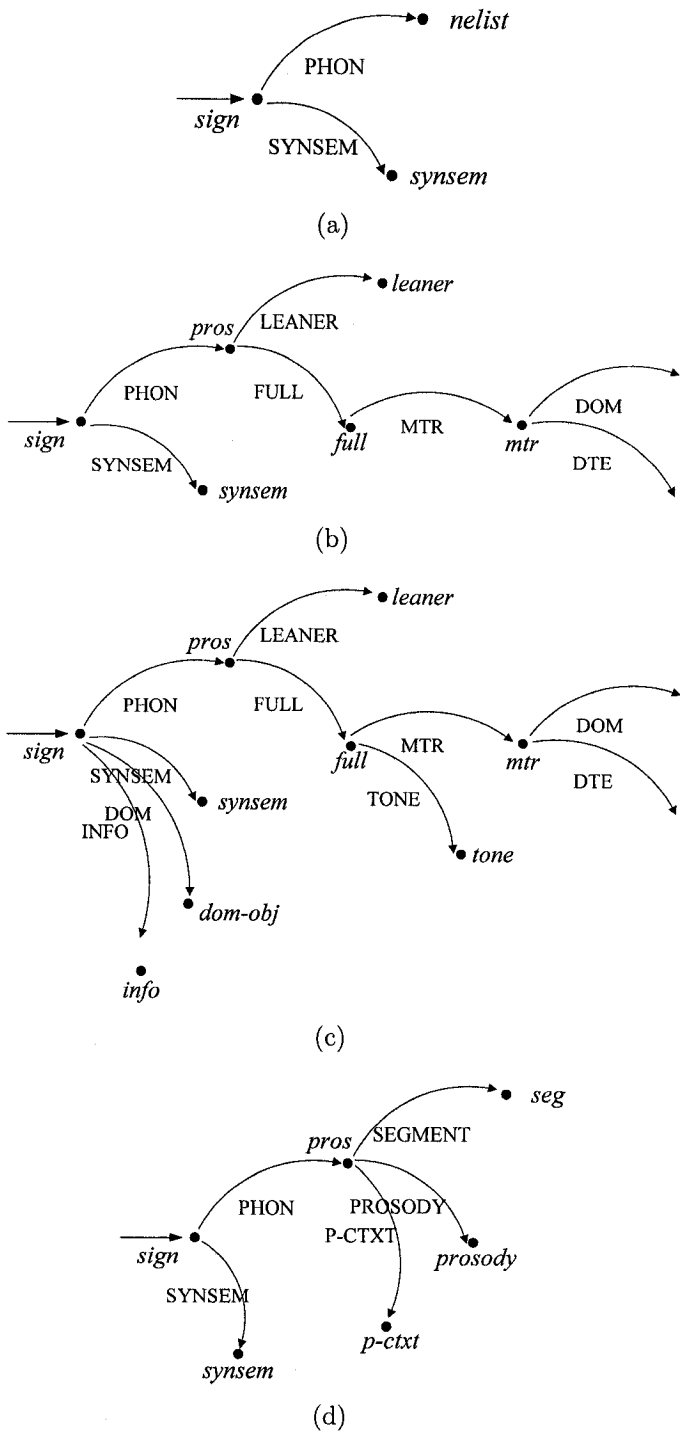


Figure 2.5: Feature structure of (a) HPSG (b) Klein's model (c) Haji's model and (d) Asudeh's model

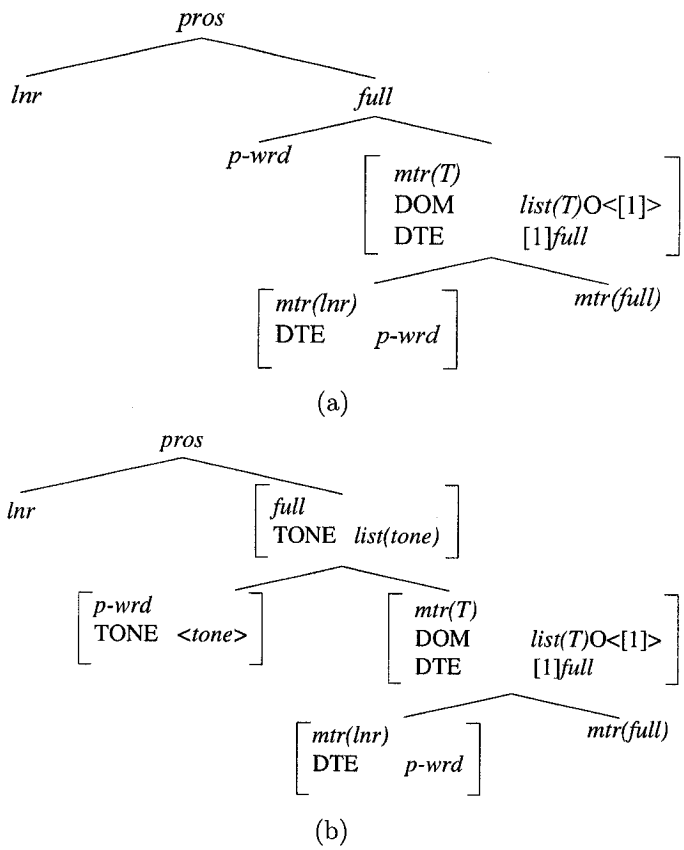


Figure 2.6: Prosodic type hierarchy: (a) Klein's hierarchy and (b) Haji's hierarchy

on a dataset, so that different formalism for Natural Language phenomena can be validated. The LKB system includes the flat semantic representation called Minimal Recursive Semantic (MRS) representation, which is the structure to represent the semantic description, i.e., semantic features and their values. The primary English grammar is generally used with the LKB system called ERG [45] developed by LINGO linguistic laboratory at Stanford University.

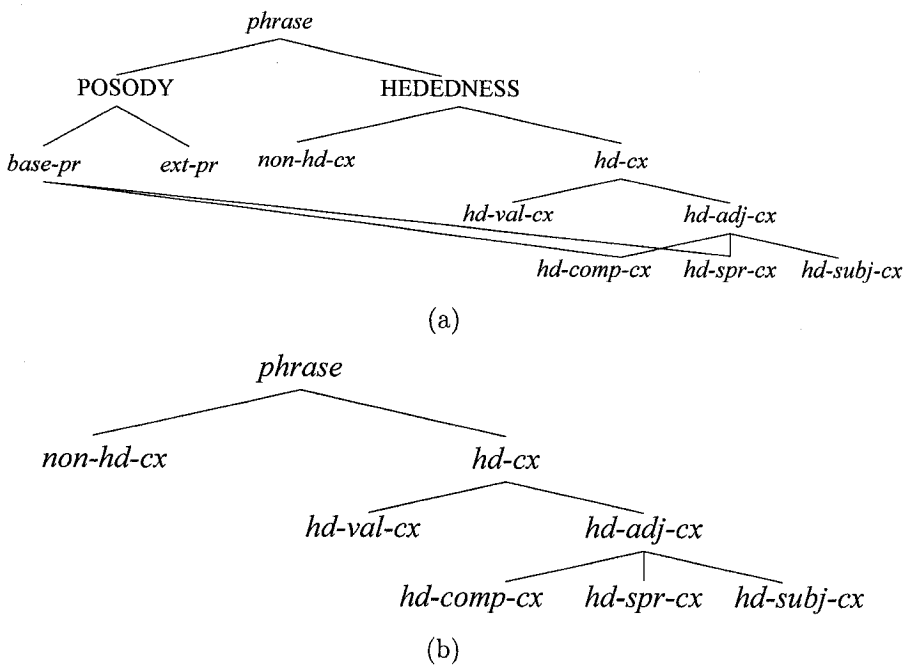


Figure 2.7: Type hierarchy of phrase construction: (a) Haji's type hierarchy of phrase construction and (b) Klein's type hierarchy of phrase construction

Chapter 3

Aspects of Developing a Unification-Based Formalism for Prosodic Generation

The design of unification-based grammar for prosodic generation is the main issue in this thesis. In this chapter, we focus on the issues relevant to focus system of a unification-based approach to prosodic generation. As a result of designing the grammar, the measurement of how well a listener can recognize the speech utterances containing the different prosodies for the same sentence is performed by a perceptual evaluation.

This is about overview of this chapter. The data preparation and feature selection for a unification-based system in the prosodic generation are discussed in section 3.1. Two feature aspects are described in section 3.2 and the explanation of the performance issue is given in the section 3.3. The last section is the introduction to the unification-based system for the prosodic generation. Two states, linguistic and speaker's intention states, are summarized in this section.

3.1 Dialogue Preparation and Feature Selection for Unification-Based System in Prosodic Generation

There are two parts of preparation process which need to be discussed: the dialogue preparation and the feature selection for the prosodic unification-based system.

3.1.1 Dialogue Preparation

Building a dialogue corpus is one of the main difficulties in prosodic generation. The requirements of building dialogue corpus are such that a great number of annotated entries are necessary for effective learning of prosodic analysis. The manual corpus preparation is very costly and even then it may be impossible to achieve sufficient coverage in real applications.

Generally, the dialogue corpus is annotated by linguists. The different dialogue corpora are created for different purposes. Linguists need to organize, annotate, and do the indexing for each sound unit in the corpus. A digital signal processing

system analyzes speech dialogue and then creates the index files following required prosodic and linguistic features. This preparation is for any Prosodic Annotation and Generation (PAG) before employing knowledge representation and learning methods to recognize the prosodic patterns. We explore the dialogue corpus called the CMU-COM Corpus [55]. These dialogues are in the traveling reservation domain. They are annotated with the prosodic representation, called the ToBI.

3.1.2 Feature Selection

One of the important issues in prosodic research is feature selection that can influence prosodic analysis. Several features have been investigated and have strong affects on developing the prosodic generation such as pitch, and break. McKeown and Pan [27] report that the linguistic features including semantic and syntactic features affect prosodic analysis and can be used to improve the sound quality for Spoken Language Generation (SLG) system. On one hand, exploring features from NLG for prosodic model was developed by Pan et al. [49] to recognize the prosodic features from human spoken sounds. The research proves that the syntactic features influence prosodic model while there are some curiosities about the effects of semantic features on a prosodic model. Pan makes conclusions that feature selection for prosodic model depends on learning methods. One of her conclusions is that semantic features can have a strong effect on prosodic models. On the other hand, Theune [46] considers the prosodic model based on linguistic knowledge. For example, Focus theory, which applies linguistic knowledge in the information domain, is used to assign prosodic patterns. This approach analyzes prosody without considering the raw speech signal.

Following these research results [33, 49, 46], two issues should be investigated.

- Flat semantic features can be considered to interpret the affects of semantic features on prosodic phenomena.

- Speaker's intention, i.e., investigating integration of particular features in the information domain, which have the relationships with prosody; for example, focus and non-focus parts in a sentence and the speaker's intention features, which influence the values of prosodic features in prosodic generation

Linguistic information can be obtained from context, but cannot provide the information of the speaker's intention and focus, which are the features directly influencing prosody. The input information of SLG systems must be composed of information analyzed from context, such as syntax and semantics, and the features which cannot be inferred from context such as speaker's intention and the focus part of a sentence.

3.2 Two Feature Aspects in Prosodic Generation

How features influence prosodic generation is explored from different points of view; i.e., utterance, speaker's intention and types of a speaker. The features can be used to design the structures and hierarchy to cover the prosodic phenomena. Two issues of prosodic generation are summarized: flat semantics and speaker's intention features. Designing the Typed Feature Structures (TFSs) representing the prosodic phenomena is considered for many reasons as described below.

1. The complex structures required for prosodic phenomena can be represented as instances of complex feature structures in a unification-based grammar. This structure can be used to interpret the relationships of focus, speaker's intention, and prosody.

2. The flexibility of analysis when informational structure is an incomplete structure. In some cases, there are missing parameters. The system can refer to the other related features and then constrain to prosodic structures.

3. The convenience of integrating prosodic concepts across domains. The integration of concepts is applied by using the same information structure. It can be applied for multilingual informational structures.

4. The prosodic structures can have complex structure that can be represented by a tree or Attribute-Value Matrix (AVM), which is easy to understand. The levels of prosodic feature structures assign relationships between information and prosodic domains.

3.2.1 Flat Semantic Features

Pan et al. [49] indicated that syntactic hierarchies and their features influence prosody in spoken sounds. Although some research papers [56, 49, 46] have considered semantic features for prosodic analysis, they also express only a few semantic features which can influence prosodic generation. As we know, many semantic features are derived from the informational domain, such as focus-nonfocus and new-given information, greatly affect the control of prosody and also improve the quality of prosodic generation. To design structures, this research considers syntactic, semantic, and informational parameters within a unification-based grammar. Complicated design is needed to cover several features. However, only features relating to Focus theory are considered and we examine how these features can affect prosody. Defining the feature structures to represent the prosodic phenomena is a complicated task. We need to investigate how the semantic features can be established to analyze prosody.

3.2.2 Speaker's Intention Features

Speech acts are types of speaker's intentions. The speaker's intention information cannot be retrieved directly from context or word meaning. These information are controlled by the speaker based on a situation in dialogue. A speaker expresses these features as psychological features which affect the prosody of speaker utterances. The speech act features convey the the speaker's intention in spoken phrases, such as comparing, emphasizing, questioning, and uncertainties. In this research, the feature structures are designed to represent the prosodic patterns for different types of speaker's intentions. For example, people can speak the same sentence as an affirmative or interrogative sentence, which depends on the intention of the speaker. Designing the feature structures need to cover the types of the speaker's intentions with respect to prosodic phenomena. Since this design is a complicated task, we set a limit to a small sets of categories of speaker's intentions.

3.3 Performance Issues in Prosodic Generation

Much research has developed PAG for specific domains such as the medical area. Furthermore, domain specific research is often used for various purposes. Several

SLG applications make different requirements on system performance. In general, three main performance measurements for a SLG system are (i) correctness of spoken sounds, (ii) listening quality of spoken sound, and (iii) the preference of a group of listeners. The details will be explained in chapter 9. They are used to evaluate how well an SLG system performs on the same tasks. Correctness of spoken sound counts the frequency of correct word utterances. Listening quality of spoken sound makes use of a group of listeners to evaluate the quality of spoken sound as good, fair, poor or bad quality. Sometimes, listening quality is used to measure the naturalness of perception. The preference of a group of listeners is performed to compare performance of prosodic generation for different SLG systems. However the comparison of prosodic generation systems is complicated because different systems use different features, different datasets and individual learning systems to analyze the prosodic phenomena.

3.4 Introduction to Unification-Based System for Prosodic Generation

Developing appropriate techniques for prosodic generation demands an investigation of prosodic phenomena. These techniques can be used to improve adaptation and the quality of utterances for an SLG application. The objective is to analyze prosodic features and to improve the performance of prosodic generation regarding the diversity of speaker utterances in the same sentence. The relationship between informational and prosodic structures are considered to find the prosodic patterns. This relationship is expressed through design of focus structure and their constraints. The constraints and unification-based subgrammar are designed following prosodic phenomena and they employ a unification-based parser for prosodic generation.

We use the constraint-based system to control a set of features which involve the relationship between informational and prosodic structures. The unification-based subgrammar is designed to constrain syntactic, semantic, focus, and speaker's intention features of the different prosodic phenomena. The grammar parses the sentence with the focus information to annotate the prosodic marks on a phrase or sentence.

This research investigates how to analyze semantics, focus, and speaker's intention for the prosodic generation system. A unification-based approach for prosodic generation is proposed. In chapter 2, some methods are introduced to find prosodic

patterns. Those methods, [27, 53], are the learning methods, which use a corpus training to find prosodic patterns, while Theune [46] used a template-based method. Some preliminary methods are proposed by Klien [2] and Haji-Abdolhosseini [1]. They used a unification-based system to find prosodic constituency from syntactic structures. There are some reasons, given below, to believe that the unification-based system will efficiently generalize typed feature structure of prosody and constituency which can be used to represent the prosodic phenomena. According to two issues in the previous section, the analysis considers two states: the linguistic state and the speaker's intention states.

1. A unification-based model supports multi-language. The feature structures of syntactic, semantic, and speech act can be used to determine the particular feature structures for many languages and to describe the prosodic attributes.

2. Hierarchical constituency. The unification-based model can represent complex structures in the form of a hierarchy, or a metrical tree, which is feasible to render prosodic structure.

3. Flexibility of the grammatical system. A unification-based model can support a rule inferencing system and be flexible to set the complicated grammar rules which are important for analyzing semantics, focus and speech act. One efficient unification-based system is Head-Driven Phrase Structure Grammar (HPSG)[57].

3.4.1 Linguistic State

In the linguistic state, the semantic constraints that employ informational features from Focus-Accent Theory [11] are considered. The relationships of syntactic, semantic and prosodic information are investigated to design the constraints for prosodic phenomena in a unification-based subgrammar. To illustrate these structures, an example of a lexical word structure is shown in figure 3.1. This figure shows the semantic and syntactic feature structure for the word "dog". It describes the status of this word such as part of speech, subject-verb agreement, etc. The feature description in this figure was reported in [58]. A semantic description of a sentence "a dog catches

a cat" is illustrated in figure 3.2. The semantic mode of this sentence is proposition (PROP) for this sentence. This structure describes the semantic representation of these sentences. More descriptions for semantic feature structure are reported in [58].

$$\left\langle dog, \begin{bmatrix} SYN & \begin{bmatrix} HEAD & \begin{bmatrix} noun & \\ AGR & \boxed{1} \end{bmatrix} \\ SPR & \left\langle DET \quad \left[AGR \ \boxed{1}\right] \right\rangle \\ COMPS & \langle\rangle \end{bmatrix} \\ SEM & \begin{bmatrix} MODE & ref \\ INDEX & i \\ RESTR & \left\langle \begin{bmatrix} RELN & dog \\ SIT & m \\ INST & i \end{bmatrix} \right\rangle \end{bmatrix} \end{bmatrix} \right\rangle$$

Figure 3.1: Lexical word structure

$$\begin{bmatrix} MODE & PROP \\ INDEX & S \\ RESTR & \left\langle \begin{bmatrix} RELN & catch \\ SIT & o \\ Catcher & i \\ Catched & j \end{bmatrix}, \begin{bmatrix} RELN & dog \\ SIT & m \\ INST & i \end{bmatrix}, \begin{bmatrix} RELN & cat \\ SIT & n \\ INST & j \end{bmatrix} \right\rangle \end{bmatrix}$$

Figure 3.2: Semantic description of the sentence "the dog catches the cat"

Furthermore, the Focus-Accent theory is important in the linguistic domain for using grammatical systems for machine understanding. For example, Haji-Abdolhosseini [1], proposed the use of informational structures to describe the Focus-Accent theory. This research will extend the idea of informational structure for prosodic analysis and design the unification-based subgrammar for prosodic phenomena as a kind of prosodic grammar.

3.4.2 Speaker's Intention State

Speech act (or speaker's intention) is an important concept for dialogue systems as reported by Campbell [59]. Speech act types correspond to prosody phenomena

and the relationships of speech act and prosodic features need to be investigated to define the constraints representing the prosodic phenomena. An example of the informational structure is shown in figure 3.3. The structure is composed of two main subfeature structures: speech act structure (*SPACT*) and focus information structure (*Focus_Info*). The *SPACT* structure contains speech act code, and sentence type. The *Focus_Info* structure contains features such as focus type, focus position, and focus part.

$$\begin{bmatrix} SPACT & [\ldots] \\ Focus\_Info & \left\{ \begin{bmatrix} actor & \\ index & i \\ \ldots & \end{bmatrix}, \begin{bmatrix} act & \\ index & j \\ \ldots & \end{bmatrix}, \begin{bmatrix} actee & \\ index & k \\ \ldots & \end{bmatrix} \right\} \end{bmatrix}$$

Figure 3.3: Informational structure

3.5 Summary

We summary the processes of a unification-based system for prosodic generation into three steps below:

1. Designing feature structures to represent the input information: the feature structures are designed to cover focus, semantics, speaker's intention information. These features are considered to have a strong affect on prosody.

2. Analyzing the relationships of syntactic, semantic, and speaker's intention features in information domain with respect to prosodic phenomena. Based on the focus theory, the relationships between prosodic and sematic features are used to improve the diversity of prosodic generation for a sentence. We consider how the relationships of prosodic features affect other features. Some prosodic patterns can be analyzed by learning from a prosodic annotation corpus while some patterns can be defined by using the intonational theories.

3. Constructing subgrammar, including grammar rules, type hierarchy, type constraints, lexicon, and typed feature structure, in the form of unification-based

formalism for the prosodic generation. This grammar is used to analyze the informational structure and annotate the prosodic marks on a sentence as a result.

Chapter 4

Analysis of Semantic Representation to Generate Focus Content Structure

The semantic representation is analyzed to find the focus information and to generate the Focus Content (FC) structure as the preprocessing process. The FC structure is provided to the Focus to Emphasize Tone (FET) analysis as an input. To analyze semantics and produce a semantic representation, an HPSG parser is employed to parse an input sentence. The HPSG parser used in this thesis is the Linguist Knowledge Building (LKB) system [3]. The LKB system is a unification-based system. The particular grammar used for LKB system is called LinGO English Resource Grammar (ERG) [45]. The LKB system with ERG can analyze syntactic and semantic structures. The basic components for the LKB system is summarized in section 4.1. The system generates the semantic information which is represented by the Minimal Recursive Semantic (MRS) representation [4]. The details of the MRS representation are described in section 4.2.

For the FET analysis, the MRS representation is transformed to the FC structure. The FC structure contains the information of "actor" (a person or a thing that acts something in a sentence), "act" (an activity in that sentence), and "actee" (the object of the act) parts. The details of this transformation are explained in section 4.3. The FC structure provides a set of focus words to the FET analysis. These focus words contain focus information of the input sentence. The example of this transformation is described in section 4.4.

4.1 Basic Components for Linguistic Knowledge Building System

A survey of unification is reported by Knight [60]. This survey includes some of history, description of algorithm for unification in the several areas. An early unification method is proposed by Robinson [61]. Later, a linear algorithm for unification is invented by Paterson and Wegman [62]. In 1982, Martelli and Montanari [63] proposed an efficient unification algorithm which was implemented using Pascal. It shows a

good performance for a typical practical conditions. Our system employed the unification algorithm of the LKB system which is implemented in LISP and proposed by Copestake [3].

In this section, unification-based parsing and the LKB system are described. TFS, unification, type constraints and other elements of the LKB system are explained in [3]. The summary of these properties and their examples are described below.

The main function of the grammar is to map between descriptive structures. For example, the result of context free parsing "`(S (NP Tim) (VP jump) )`" is a labeled bracket notation as description. The words "`Tim`" and "`jumps`" are mapped to NP and VP respectively. In language generation, grammar is used for generating character strings from partial information provided. For example, the above structure of the sentence "Tim jumps" can be partially represented by "`(S (NP ?) (VP jumps ) )`". "?" is a placeholder which represents a living thing who can "jump" such as "Mary jumps", or "Kim jumps". The grammar for the LKB system is composed of four main components: type system, lexical entries, grammar rules and start structure.

4.1.1 The LKB Type System

Copestake [3] described that type system acts as the defining framework for the grammar. The type system considers which structures are mutually compatible and which features can occur. It is an inheritance system which allows generalizations to be expressed. The type system is composed of two main parts: type hierarchy and type constraints. For the TFSs, the typed hierarchy, unification, typed inference (described below) are required to define a set of typed constraints.

Type Hierarchy

Type Hierarchy indicates specialization and consistency of types. Following Copestake's declaration [3], all type hierarchies in the LKB system must obey the conditions below:

Properties of type hierarchy

- **Unique top**: There is a single hierarchy containing all the type with a unique top type

- **No cycles**: There are no cycles in the hierarchy
- **Unique greatest lower bounds**: Any two types in the hierarchy must either be incompatible, in which case they will not share any descendants, or they are compatible, in which case they must have a unique highest common descendant (referred to as the unique greatest lower bound)

For example, a type hierarchy is shown in figure 4.1 [3] and the relationships between any pairs of types can be described with one of the categories below:

1. The types have no decedents in common (e.g., *vertebrate* and *invertebrate*)
2. The types have a hierarchical relationship (e.g., *animal* and *bee*) in which case the unique greatest common descendant is trivially the lower type.
3. There is a unique greatest common descendant. For example, *vertebrate* and *swimmer* have *vertebrate-swimmer* as a common descendant. *Fish* is the descendant of *vertebrate-swimmer* and *cod* and *guppy* are also common descendants, but *fish* is above both of them in the hierarchy.

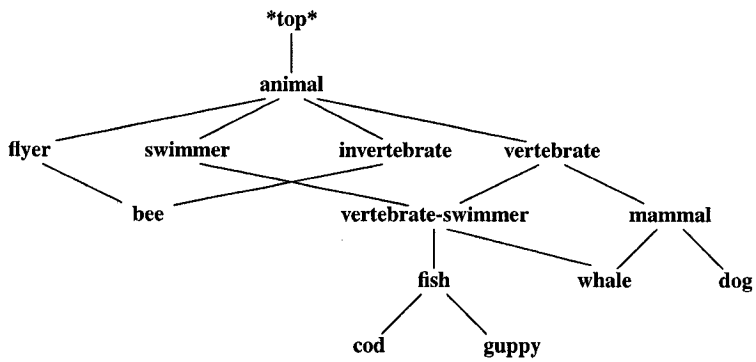


Figure 4.1: Type hierarchy of animal

4.1.2 Typed Feature Structure

TFS can be represented as a graph. One graph has exactly one type on each node with a labeled arcs connecting to other nodes. The labels on the arcs are called features.

To define the TFS for the LKB system, any TFSs must follow the conditions below. These conditions are defined by Copestake [3].

Properties of typed feature structures

- **Connectedness and unique root:** A TFS must have a unique root node: apart from the root, all nodes have one or more parent nodes
- **Unique features:** Any node may have zero or more arcs leading out of it, but the label on each edge must be unique.
- **No cycles:** No node may have an arc that intervenes between it and the root node.
- **Types:** Each node must have a single type, which must be present in the type hierarchy
- **Finiteness:** A TFS must have a finite number of nodes

For example, the graph notation in figure 4.2 represents the TFS of `np_rule`. Rather than using the graph notation to represent TFS which is cumbersome, the Attribute-Value Matrix (AVM) is used to represent the TFS as the alternative notation which is easy to understand. The AVM of the TFS of `np_rule` is shown in figure 4.3.

In the LKB system, the description language is used to represent the AVM of TFSs. The description language is in the form of a script language and it is easier for editing as illustrated in figure 4.4. The main differences between AVM and description language are that the AVM notation has the types inside the square bracket, while the description language puts them outside, and the description language requires the conjunction symbol &.

4.1.3 The LKB Unification

Unification is an operator and it combines two typed feature structures into the most general feature structure, which retains all the information which they individually contain. If there is no such feature structure, unification fails.

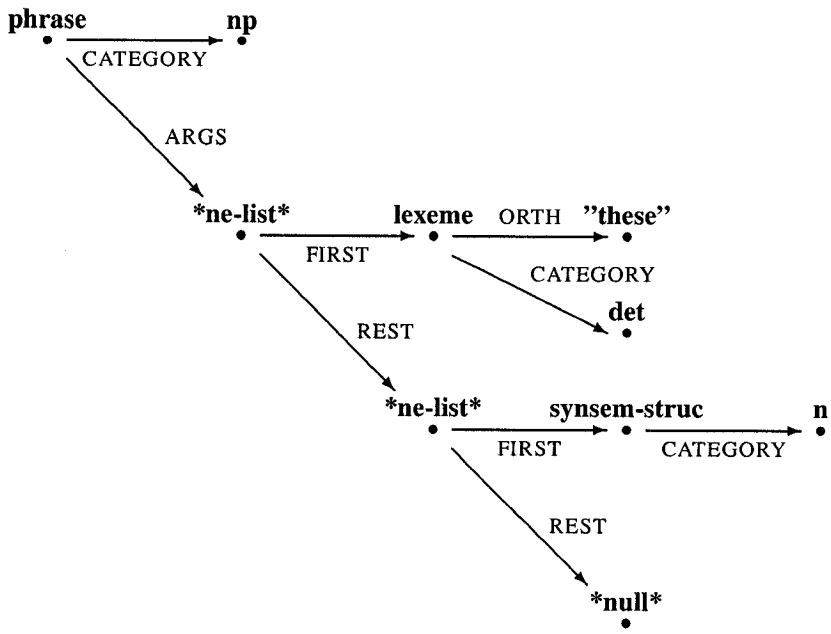


Figure 4.2: Graph representation

$$\begin{bmatrix} phrase \\ CATEGORY & np \\ & \begin{bmatrix} *ne-list* \\ FIRST & \begin{bmatrix} lexeme \\ ORTH & "these" \\ CATEGORY & det \end{bmatrix} \\ ARGS & \\ REST & \begin{bmatrix} *ne-list* \\ FIRST & \begin{bmatrix} synsem-struc \\ CATEGORY & n \end{bmatrix} \\ REST & *null* \end{bmatrix} \end{bmatrix} \end{bmatrix} \tag{4.1}$$

Figure 4.3: AVM representation

To describe unification, we need to understand the subsumption which is a kind of relation between two TFSs. Following the subsumption described in [3], TFSs can be regarded as being ordered by specificity. TFS specificity can be determined automatically, based on a notion of the information the TFS contain. For example, TFS in the AVM (4.2) contains more information than (4.3). The AVM (4.2) specifies that G and F.H are equivalent. The AVM (4.3) leaves this open and contains no information that is not in (4.2). Thus, (4.2) is strictly more general than (4.3). We can conclude that (4.2) subsumes (4.3).

```
example := phrase &
           [ CATEGORY np
             ARGS     ne-list &
                      [ FIRST   lexeme &
                                [ ORTH      "these",
                                  CATEGORY  det ],
                        REST    *ne-list* &
                                [ FIRST     synsem-struc &
                                            [ CATEGORY n ] ],
                                  REST      *null* ] ] ] .
```

Figure 4.4: Description language representation

$$\begin{bmatrix} \mathbf{t} & \\ F & \boxed{0}\begin{bmatrix} \mathbf{u} & \\ F & \boxed{1}\,\mathbf{a} \\ H & \boxed{1} \end{bmatrix} \\ G & \boxed{1} \\ J & \boxed{0} \end{bmatrix} \tag{4.2}$$

$$\begin{bmatrix} \mathbf{t} & \\ F & \boxed{0}\begin{bmatrix} \mathbf{u} & \\ F & \boxed{1}\,\mathbf{a} \\ H & \boxed{1} \end{bmatrix} \\ G & \mathbf{a} \\ J & \boxed{0} \end{bmatrix} \tag{4.3}$$

In the LKB system, the most general TFS of all is always [*top*]. The subsumption can be described as follows:

Properties of subsumption

A TFS FS1 subsumes another TFS FS2 if and only if the following conditions hold:

- **Path values:** For every path, P in FS1 with a value of type **t**, there is a corresponding path P in FS2 with a value which is either **t** or a subtype of **t**.

- **Path equivalences:** Every pair of paths P and Q which are reentrant in FS1 (i.e., which lead to the same node in the graph) are also reentrant in FS2.

Unification for the LKB system can be defined in term of subsumption.

Properties of unification

The unification of two typed feature structures F and G is the most general typed feature structure which is subsumed by both F and G, if it exists.

The symbol for unification in the LKB system is defined by ⊓. For example, (4.4) shows the unification of two TFSs. Both structures have the same type `lexeme` but they contain the different features. The first structure has the feature `ORTH` while the second structure has the feature `CATEGORY`. The result of unification is shown in the last structure. The type of this structure is `lexeme` and this structure obtains the features from the first and second structures together.

$$\begin{bmatrix}\text{lexeme} & \\ \text{ORTH} & \text{“these”}\end{bmatrix} \sqcap \begin{bmatrix}\text{lexeme} & \\ \text{CATEGORY} & \text{np}\end{bmatrix} = \begin{bmatrix}\text{lexeme} & \\ \text{ORTH} & \text{“these”} \\ \text{CATEGORY} & \text{np}\end{bmatrix} \tag{4.4}$$

With respect to unification, the term “failure” is defined by the symbol ⊥ which stands for inconsistency. For example, the unification of following two structures in (4.5) is ⊥. Both structures are the same types `synsem-struc` and contain the same features `CATEGORY`. However, the features `CATEGORY` of these structures are defined with different types or are referring to different nodes. The `CATEGORY` of one structure is defined as vp while the `CATEGORY` of another structure is np. Therefore, the inconsistency occurs because of the inconsistent types for the path `CATEGORY`.

$$\begin{bmatrix}\text{synsem-struc} & \\ \text{CATEGORY} & \text{vp}\end{bmatrix} \sqcap \begin{bmatrix}\text{synsem-struc} & \\ \text{CATEGORY} & \text{np}\end{bmatrix} = \bot \tag{4.5}$$

4.1.4 Type Constraints and Inheritance

Described by Copestake [3], the primary purpose of type constraints is that they can be used to allow generalizations to be expressed, so that lexical entries and other

descriptions can be kept succinct. The secondary purpose is to avoid error creeping into a grammar, such as misspelt features names. In the LKB system, the constraint on a type is expressed as a TFS.

Let the *substructure* of a TFS be the TFS rooted at each node in the structure. The well-formed TFS is described in terms of conditions on each substructure.

Properties of a well-formed TFS

- **Constraint:** Each substructure of well-formed TFS must be subsumed by the constraint corresponding to the type on the substructure's root node.
- **Appropriate features:** The top-level feature for each substructure of a well-formed TFS must be the appropriate feature of the type on the substructure's root node.

The structures in (4.6-4.7) are the examples of well-formed and not well-formed structures. These examples are analyzed by using the grammar in table 4.1. The structure in (4.6) is the well-formed TFS. The types in this structure are matched with the type in the table 4.1 [3].

$$\begin{bmatrix} \text{phrase} & \\ \text{CATEGORY} & \text{s} \\ \text{ARGS} & \text{*list*} \end{bmatrix} \tag{4.6}$$

The structure in (4.7) is not well-formed and do not subsume well-formed structures also. This structure contains the wrong type on the feature `CATEGORY` following the table 4.1.

$$\begin{bmatrix} \text{phrase} & \\ \text{CATEGORY} & \text{lexeme} \\ \text{ARGS} & \text{*list*} \end{bmatrix} \tag{4.7}$$

Table 4.1: Constraints and appropriate features for the tiny grammar

type	constraint	appropriate features
top	[*top*]	
string	[string]	
list*	[*list*]	
ne-list*	[*nelist* ; FIRST *top* ; REST *list*]	FIRST REST
null*	[*null*]	
synsem-struc	[synsem-struc ; CATEGORY cat]	CATEGORY
cat	[cat]	
s	[s]	
np	[np]	
vp	[vp]	
det	[det]	
n	[n]	
phrase	[phrase ; CATEGORY cat ; ARGS *list*]	CATEGORY ARGS
lexeme	[lexeme ; ORTH string ; CATEGORY cat]	CATEGORY ORTH
root	[root ; CATEGORY s ; ARGS *list*]	CATEGORY ARGS

The structure in (4.8) is not well-formed but it subsumes well-formed structures. This structure contains the wrong type on `CATEGORY` but it is compatible with a valid type.

$$\begin{bmatrix} \text{phrase} & \\ \text{CATEGORY} & \text{*top*} \\ \text{ARGS} & \text{*list*} \end{bmatrix} \tag{4.8}$$

Type inference

Type inference [3] takes a non-well-formed TFS and returns the most general well-formed structure which it subsumes. Thus it always preserves the information in the initial structure. It is always possible to find a unique most general well-formed structure from a non-well-formed structure if the latter subsumes any well-formed structures. The LKB system carries out type inference on all entries such as lexical entries, grammar rules and so on. It is an error to define an entry which cannot be converted into a well-formed TFS. The unification of well-formed TFSs is used to provide a well-formed result from non-well-formed structure that subsumes well-formed structure. The conditions of the unification of well-formed TFS are described below:

> **Properties of well-formed unification:** The well-formed unification of two TFSs `F` and `G` is the most general well-formed TFS which is subsumed by both `F` and `G`, if it exits.

For instance in [3], let the constraint on **`swimmer`** be the structure (4.9), the constraint on **`mammal`** is the structure (4.10) and the constraint on **`whale`** is the structure (4.11).

$$\begin{bmatrix} \text{swimmer} & \\ \text{FINS} & \text{boolean} \end{bmatrix} \tag{4.9}$$

$$\begin{bmatrix} \text{mammal} & \\ \text{FRIENDLY} & \text{boolean} \end{bmatrix} \tag{4.10}$$

$$\begin{bmatrix} \text{whale} & \\ \text{HARPOONED} & \text{boolean} \\ \text{FINS} & \text{true} \\ \text{FRIENDLY} & \text{boolean} \end{bmatrix} \tag{4.11}$$

Considering the unification between two TFSs as shown in (4.12). In type `mammal`, feature `FRIENDLY` is defined as `true`. The result of this unification is illustrated in (4.13). This structure is not a well-formed structure. It lacks the feature `HARPOONED` and the value of `FINS` must be `true`.

$$\begin{bmatrix} \text{mammal} & \\ \text{FRIENDLY} & \text{true} \end{bmatrix} \sqcap \begin{bmatrix} \text{swimmer} & \\ \text{FINS} & \text{boolean} \end{bmatrix} \tag{4.12}$$

$$\begin{bmatrix} \text{whale} & \\ \text{FINS} & \text{boolean} \\ \text{FRIENDLY} & \text{true} \end{bmatrix} \tag{4.13}$$

To make a well-formed structure, the additional constraint information (the feature that does not appear in (4.13) but appears in (4.11) must be added to make a well-formed structure as shown in (4.14). The feature HARPOONED is included in this structure.

$$\begin{bmatrix} \text{whale} & \\ \text{HARPOONED} & \text{boolean} \\ \text{FINS} & \text{true} \\ \text{FRIENDLY} & \text{true} \end{bmatrix} \tag{4.14}$$

Conditions on type constraints.

The final part of the description of the typed feature structure formalism concerns the construction of full type constraints from the descriptions and the conditions on the type constraints.

There is a series of conditions on full type constraints which determine how the

local constraints are expanded into the full constraints. The properties of type constraints is reported below:

Properties of type constraints

- **Type:** The type of the TFS expressing the constraint on a type `t` is always `t`
- **Consistent inheritance:** The constraint on a type must be subsumed by the constraints on all its parents. This means that any local constraint specification must be compatible with the inherited information, and that in the case of multiple inheritance, the parent's constraints must unify.
- **Maximal introduction of features:** Any feature must be introduced at a single point in the hierarchy. That is, if a feature, `F`, is an appropriate feature for any of its ancestors, then `F` cannot be appropriate for a type which is not a descendant of `t`. Note that the consistent inheritance condition grantees that the feature will be appropriate for all descendant of `t`.
- **Well-formedness of constraints:** All full constraint feature structures must be well-formed.

For example, a set of TFSs in (4.15) is used to define the type lists in the LKB system. With these TFSs, `*ne-list*` (non-empty-list) can be represented by the AVM as shown in (4.16) and can be expanded to the AVM in (4.17). With this definition in (4.15), the structure can be typed without causing an infinite structure.

$$
\begin{aligned}
list &:= *top* * . \\
null* &:= *list* * . \\
ne-list* &:= *list* * \,\&\, \begin{bmatrix} \text{FIRST} & *top*, \\ REST & \text{*list*} \end{bmatrix}.
\end{aligned}
\tag{4.15}
$$

$$\begin{bmatrix} \text{*ne-list*} & \\ \text{FIRST} & \text{*top*} \\ \text{REST} & \text{*ne-list*} \end{bmatrix} \tag{4.16}$$

$$\begin{bmatrix} \text{*ne-list*} & & \\ \text{FIRST} & \text{*top*} & \\ \text{REST} & \begin{bmatrix} \text{*ne-list*} & \\ \text{FIRST} & \text{*top*} \\ \text{REST} & \text{*list*} \end{bmatrix} \end{bmatrix} \tag{4.17}$$

Lexical Entries

Following Copestake's description, lexical entries are used to define the relationships between the characters in a word and some linguistic description of word or the description of a particular sense of word. In the LKB system, the `ORTH` feature, which represents the orthography of word, is used to specify the string in the lexical entries. For example, the structure in figure 4.5 represents the lexical entry of the word "whale". In this structure, the type `whale` contains two features: orthography (`ORTH`) and category (`CATEGORY`) which is represented as noun (n) in this structure.

```
whale := word &
         [ ORTH      "whale",
           CATEGORY  n       ] .
```

Figure 4.5: Lexical entry

4.1.5 Grammar Rules and Start Symbol

Grammar rules are the TFSs that describe how to combine lexical entries and phrases to make further phrases. For example, the TFS grammar rule in figure 4.6, the mother phrase structure includes the daughter phrase structure. The mother phrase contains the category `s` which is represented sentence while the daughter phrase contains the feature `ARGS`. The feature `ARGS` includes the order of the list elements corresponding to the linear order of the daughter.

```
s_rule := phrase &
          [ CATEGORY    s,
            ARGS        [ FIRST    [ CATEGORY np],
                          REST     [ FIRST    [ CATEGORY vp],
                                     REST *null* ] ] ].
```

Figure 4.6: Grammar rule

For the start symbol in the LKB system, the equivalent of the start symbol in the standard category in represented by **root**. The example structure is shown in figure 4.7.

```
root := phrase &
        [ CATEGORY   s ].
```

Figure 4.7: Root

4.1.6 Example of Parsing for the LKB System

Parsing in the LKB system is processed by unification with the basic components, such as type system, grammar rules and so on, as described above. To understand parsing in the LKB system, an example of unification among three TFSs is illustrated in (4.18). The first TFS represents the structure of a noun phrase (`np`). The second and third structures are the lexical entries of the words "the" and "dog" with their descriptions. For example the category (`CATEG`) of "dog" is noun (`n`) which is a singular noun (`sg-word`). The result of unification is shown in the TFS of (4.19). The noun phrase includes the values of the lexical entries which correspond to the TFS of noun phrase in the first structure of (4.18).

$$
\begin{bmatrix}
\text{phrase} & & & \\
\text{CATEG} & \text{np} & & \\
\text{NUMAGR} & \boxed{1}\ \text{agr} & & \\
\text{ARGS} & \begin{bmatrix}
\text{FIRST} & \begin{bmatrix} \text{syn-struc} & \\ \text{CATEG} & \text{det} \\ \text{NUMAGR} & \boxed{1} \end{bmatrix} \\
\text{REST} & \begin{bmatrix} \text{FIRST} & \begin{bmatrix} \text{syn-struc} & \\ \text{CATEG} & \text{n} \\ \text{NUMAGR} & \boxed{1} \end{bmatrix} \\ \text{REST} & \text{*null} \end{bmatrix}
\end{bmatrix}
\end{bmatrix}
\sqcap
\begin{bmatrix}
\text{ARGS} & \begin{bmatrix} \text{FIRST} & \begin{bmatrix} \text{word} & \\ \text{ORTH} & \text{“the”} \\ \text{CATEG} & \text{det} \\ \text{NUMAGR} & \text{agr} \end{bmatrix} \end{bmatrix}
\end{bmatrix}
\sqcap
\begin{bmatrix}
\text{ARGS} & \begin{bmatrix} \text{REST} & \begin{bmatrix} \text{FIRST} & \begin{bmatrix} \text{sg-word} & \\ \text{ORTH} & \text{“dog”} \\ \text{CATEG} & \text{n} \\ \text{NUMAGR} & \text{sg} \end{bmatrix} \end{bmatrix} \end{bmatrix}
\end{bmatrix}
\tag{4.18}
$$

$$
=
\begin{bmatrix}
\text{phrase} & & & \\
\text{CATEG} & \text{np} & & \\
\text{NUMAGR} & \boxed{2}\ \text{sg} & & \\
\text{ARGS} & \begin{bmatrix}
\text{FIRST} & \begin{bmatrix} \text{word} & \\ \text{ORTH} & \text{“the”} \\ \text{CATEG} & \text{det} \\ \text{NUMAGR} & \boxed{2} \end{bmatrix} \\
\text{REST} & \begin{bmatrix} \text{FIRST} & \begin{bmatrix} \text{sg-word} & \\ \text{ORTH} & \text{“dog”} \\ \text{CATEG} & \text{n} \\ \text{NUMAGR} & \boxed{2} \end{bmatrix} \\ \text{REST} & \text{*null*} \end{bmatrix}
\end{bmatrix}
\end{bmatrix}
\tag{4.19}
$$

4.1.7 English Resource Grammar

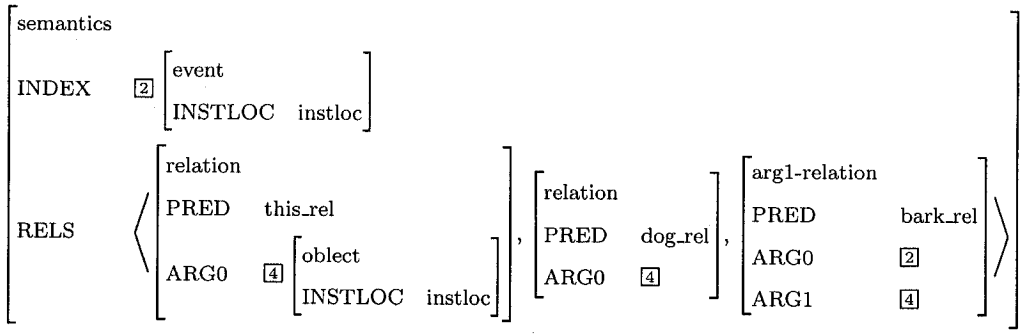
ERG is a broad-coverage, linguistically precise HPSG-based grammar of English [45]. It is the primary grammar used by the LKB system. ERG is embedded in the MRS

structure for the semantic description in form of a flat semantic representation. Reporting in [45], this grammar contains good coverage of the constructions most frequently found in the Verbmobil data which is concerned with meeting scheduling and travel reservations. ERG contains the transcriptions of some 10,000 utterances, which vary in length from one word to more than thirty words. The hand-built lexicon of around 10,000 words is somewhat tuned to this domain, augmented more recently to accommodate the vocabulary found in electronic commerce email messages studied by some of CSLI's industrial affiliates.

4.2 Minimal Recursive Semantic Representation

In the LKB system, the MRS structure is used to represent the semantic features, their types and their values in the TFSs. For example, the MRS representation of the sentence "The dog barks" is shown in (4.20). The MRS structure is composed of two main features: `INDEX` and `RELS`. The `INDEX` links to an object or event variable. `RELS` takes a list as a value. In this feature structure, the semantics is built as a list of Elementary Predications (EPs) [3]. EP is the combination of predicate with its arguments. Conjunction between EPs is implicit. In (4.20), `RELS` list is equivalent to $[this(c) \wedge dog(c) \wedge barks(e, c)]$. The MRS representation is a flat semantics because there are no embedding of predications. Each EP consists of a relation which contains the features: `PRED` and `ARG0`. `PRED` has a string value corresponding to the predicate symbol. `ARG0` is the event argument for verbs, nouns, and etc. The predicates, which require more than arguments, can be assigned by `ARG1`, `ARG2`, and so on. Each subtype of relation is marked with a fixed number of arguments. Equivalence of arguments is implemented by co–indexation. For instance, the `ARG1` of the relation corresponding to "bark" in the TFS is co–indexed with the `ARG0` of "dog".

The LKB system is distributed by CSLI Linguistic Grammars Online (LinGO) Laboratory at Stanford University. The LKB system is a grammar and lexicon environment to use with constraint-based linguistic formalisms. Generating the MRS representation, the LKB system requires a particular grammar called ERG. The LKB system with ERG can parse an sentence and analyze the syntactic structure and semantic structure of a sentence. The result of this parsing is the syntactic tree and the MRS representation.



(4.20)

The MRS representation represents the semantic structure by using a set of reference numbers to define the relation of these objects to their arguments and the identities or properties of the objects. These reference numbers connect objects together and then form the hierarchy of the relation among these numbers. Described by Copestake [4], the general MRS approach is about what this inventory of relation features consists of, being equally compatible with the use of thematic roles such as ACT described by Davis and Hirschberg [12] and a semantically-bleached nomenclature, such as ARG1, ARG2.

The MRS structure is composed of *TOP*, *LTOP*, *LZT*, and *HCONS*. *TOP* is the top handle while LTOP represents the local top. LZT is the feature that introduces the bag of elementary predications (*EPs*) which is a list in the feature structure system. *HCONS* introduces the handle constraints, which are also implemented as a list. The type *qeq* represents the equivalence between the positions of argument lists. The reference numbers, which start with h, are represented the hierarchical positions of semantic structures. The reference numbers, which start with x are the objects or nodes in the sentence. Each object has the description about the person's number and types of noun such as third person singular noun (*3SG*). The reference numbers which start with e are the events of the sentence. Each event describes the tense and environments of related objects such as past tense (*PAST*) and indicative mood (*INDICATIVE*). The details of the MRS representation are described in Copestake et al. [4]. For example, the results of parsing the sentence "Mary bought a book about bats" by the LKB system with ERG are the syntactic tree as shown in figure 4.8 and the the MRS representation in figure 4.9.

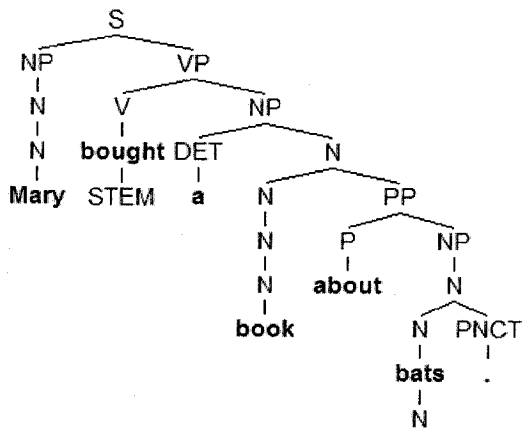


Figure 4.8: Syntactic tree of "Mary bought a book about bats"

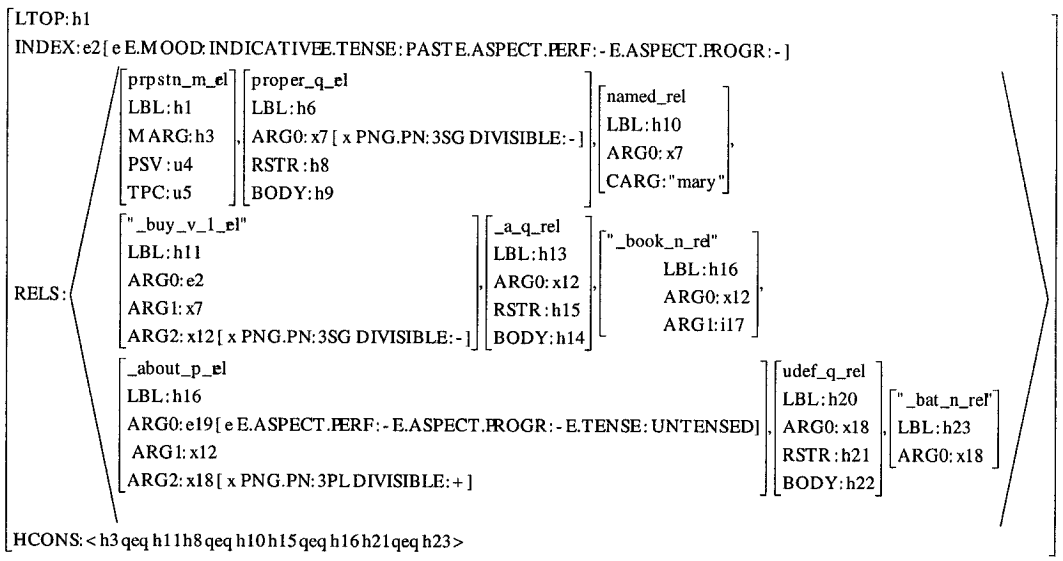


Figure 4.9: MRS representation of "Mary bought a book about bats"

It is complicated to use the MRS representation to retrieve the focus information from their relation among the lists of objects. Transforming the MRS representation to a comprehensive structure is done to support the analysis of focus information. In the next section, the transformation process is explained, and it is explained how to rearrange the feature structure, retrieve some necessary information, and generate the FC structure.

4.3 Transformation from MRS Representation to FC Structure

Illustrating the MRS representation requires a comprehensible structure which can be in form of AVM. The AVM represents clearly the semantic relations of the objects in the sentence and makes feasible the analysis of focus information and generation of the FC structure.

The MRS representation is transformed to AVM by scanning each feature inside the MRS representation. The reference numbers are kept and mapped to their objects. Every connection that is related to this object and this reference number needs to be recorded as shown in figure 4.10. These reference numbers and their agreements are illustrated on the syntactic tree in figure 4.11.

```
Line
1: prpstn_m_rel:→
2: MARG:go to h3
3: LBL= h3:→"_buy_v_1_rel":→ARG0:→[e2 E.MOOD: INDICATIVE E.TENSE: PAST],
4:                          ARG1→3SG,
5:                                go to h8:ARG0→ named_rel(x7)→ "mary"(x7)
6:                                go to h6:ARG0→ proper_q_rel(x7)
7:                          ARG2:→3SG,
8:                                go to h13: ARG0→ "_a_q_rel"(x12)
9:                                go to h15: ARG1→ "_about_p_rel"(x12)
10                                              ARG2:→3PL,
11                                                    go to h20:ARG0→ udef_q_rel(x18)
12                                                    go to h21:ARG0→ "_bat_n_rel"(x18)
13                                go to h15:ARG0→ "_book_n_rel"(x12)
14 PSV:→u4,
15 TPC:→u5,
```

Figure 4.10: Scanning into the MRS representation

By scanning into the MRS representation, the semantic feature structure is extracted along with the required features from the MRS representation. The AVM illustrates clearly the semantic relations of the objects in the sentence and is more comprehensible than the MRS representation as shown in figure 4.12.

Since the FC structure requires focus parts (actor, act, and actee parts) for the sentence, each focus part is declared depending on the main verb of the sentence. For example, the semantic relation of the sentence "Mary bought a book about bats" is represented in figure 4.13. The relation '*'buy_v_1_rel*" is on the top of the relation

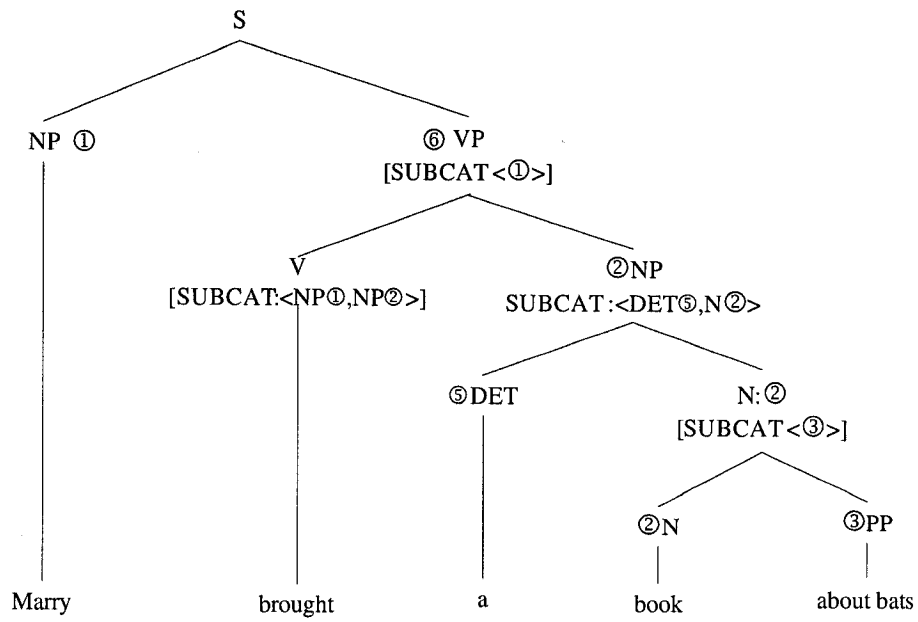


Figure 4.11: Syntactic structure with the reference number and their agreements

and is defined as the act part. The relation "*buy_v_1_rel*" connects between actor and actee. In figure 4.12, the actor is represented by the object "Mary" and its reference number is x_7. The actee is represented by the list of objects "a book about bats" and the reference number of actee is x_{12}.

Determining the properties of objects inside the AVM, the feature structures of example words in the sentence are illustrated in figure 4.14. "Mary" in figure 4.14(a) is a proper name and this object is a third person singular noun. Figure 4.14(c) shows that the word "book" is an object of the third person singular noun while figure 4.14(d) is the object of the third person plural noun. Figure 4.14(b) shows the event of the act "buy". This structure describes the connection between "actor" and "actee" depending on the relation "buy".

To generate the FC structure, the list of contents obtains the semantic information from the AVM. The list of content is composed of the structures of actor, act and actee parts. The content structure of our example is shown in figure 4.15. Each focus part has a reference number (*Index*) and the related reference numbers (*RIndex*) which are connected to the related objects. This semantic information can be used to generate the FC structure representing a set of focus words as inputs of the FET analysis. In figure 4.16, the FC structure contains the index number linking focus

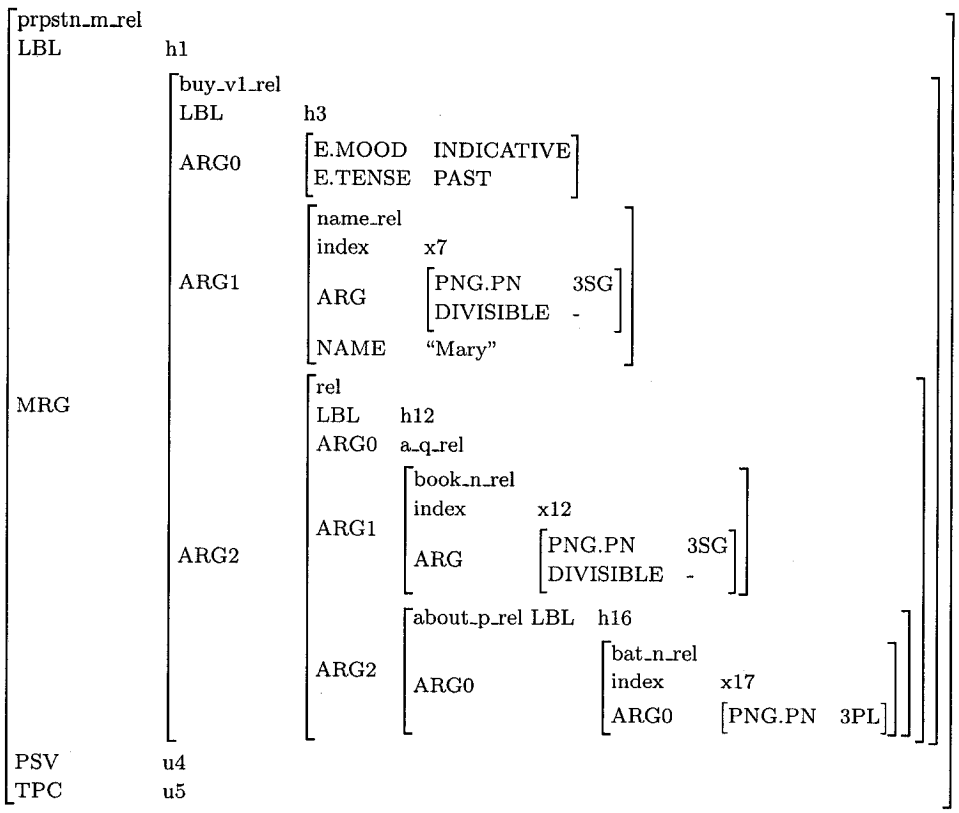


Figure 4.12: AVM of semantic representation for "Mary bought a book about bats"

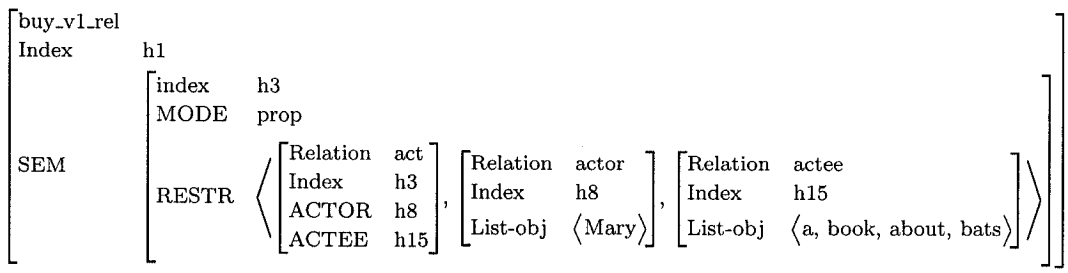


Figure 4.13: Semantic information of the relation "buy_v_1_rel"

parts (actor act, and actee parts) together, their relations and the list of objects or list of words in the sentence.

4.4 A More Complex Example of Transformation of MRS Representation to FC Structure

In FET system, each focus part contains the lists of words. For example, let us consider the sentence "a young boy bought a red flower for his mother". The words in this sentence are grouped as [a, young, boy], [bought], [a, red, flower], and [for, his,

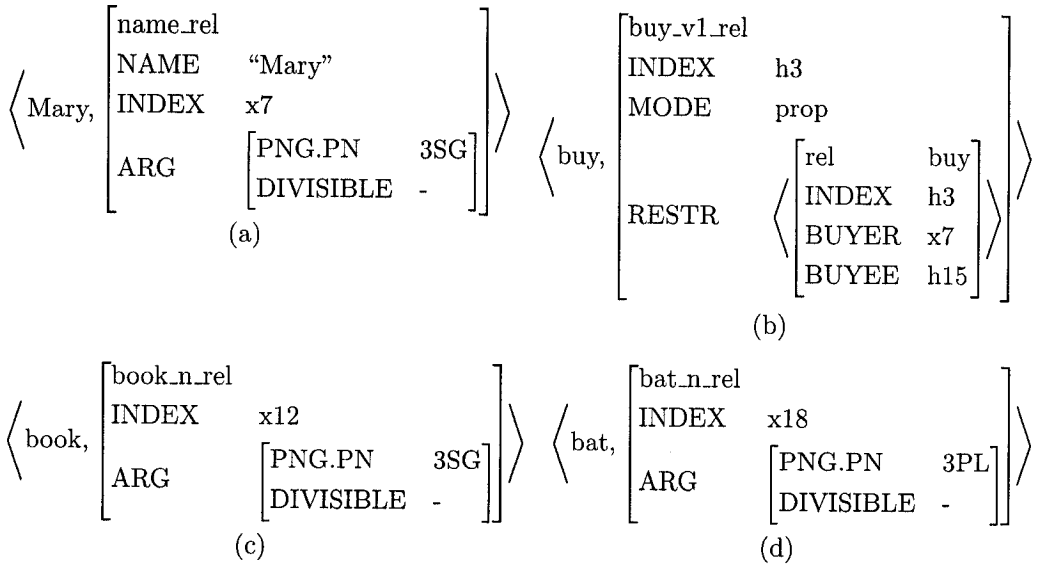


Figure 4.14: Semantic structures of: (a) Mary (b) buy (c) book (d) bat

$$\begin{bmatrix} \text{Relation} & \text{buy} \\ \text{Index} & \text{h3} \\ \text{ACTOR} & \langle \text{Mary} \rangle \\ \text{ACTEE} & \langle \text{a, book, about, bat} \rangle \end{bmatrix}$$

Figure 4.15: Semantic information of the relation "Buy"

$$\left\{ \begin{bmatrix} \text{Relation} & \text{actor} \\ \text{List-obj} & \langle \text{Mary} \rangle \\ \text{Index} & \text{h8} \\ \text{RIndex} & \text{h3} \end{bmatrix}, \begin{bmatrix} \text{Relation} & \text{actee} \\ \text{List-obj} & \langle [\text{a, book}], [\text{about, bat}] \rangle \\ \text{Index} & \text{h15} \\ \text{RIndex} & \text{h3} \end{bmatrix}, \begin{bmatrix} \text{Relation} & \text{act} \\ \text{List-obj} & \langle \text{buy} \rangle \\ \text{Index} & \text{h3} \\ \text{RIndex} & \{\text{h8,h15}\} \end{bmatrix} \right\}$$

Figure 4.16: Semantic information of the sentence "Mary bought a book about bats"

mother], based on the focus parts. The lists of words in a sentence can be marked by prosodic marks sequentially. If we mark the tone on this sentence, then one of the results is [a young boy]$_{t_1}$, [bought], [a, red, flower], [for his mother]; t_i represents the tone mark at order no. i. The tone is marked at the actor part and the focus of this sentence is "who bought a red flower". However, the hierarchy of the MRS structure must be minimized to be a flat tree structure. We design and implement the algorithm called Focus Content Scoping (FCS) to reduce hierarchy for a simple sentence.

Focus Content Scoping

The FCS's algorithm starts from the bottom or leaf nodes of the MRS structure. The *n-relation* is declared as the noun phrase relation. *q-relation* is the quantifier relation while *a-relation* is the adjective relation. The *p-relation* is the preposition phrase relation. The level of *p-relation* is usually lower than *n-relation*'s level to modify the noun phrase. We move the level of *p-relation* to upper level for our prosodic analysis. The *c-relation* is the conjunction relation. The *c-relation* is divided to left node index (*L-index*) and right node index (*R-index*). The FCS's algorithm is described as Algorithm 1 below.

Algorithm 1 Focus Content Scoping (FCS)

Input: A tree of MRS structure (T) for the input sentence (S).
Output: Flat Tree.

1: **repeat**
2: start from leaf node
3: move *a-relation* and *q-relation* to be in the same level of their *n-relation*
4: **for** each $p - relation\ in\ T$ **do**
5: merge the preposition node with a next node
6: move the node to upper level
7: **end for**
8: **for** each $c - relation\ in\ T$ **do**
9: merge *R-index* with conjunction
10: move both *L-index* and *R-index* to the upper level
11: **end for**
12: **until** T is a flat tree

For the example, the sentence "A young boy bought a red flower for his mother" is parsed by the LKB system with ERG. As a result, The MRS representation of this sentence is shown in figure 4.17. The FCS begins by scanning into this MRS representation following the reference numbers or indices. All connections of objects depending on the reference numbers are recorded and the result of scanning the MRS representation is illustrated in figure 4.18. The indentation in this figure represents the scanning into the next level of hierarchy inside the MRS representation.

Using the scanning result, the MRS representation is transformed to the AVM as shown in figure 4.19. The AVM is a standard matrix structure and we used the

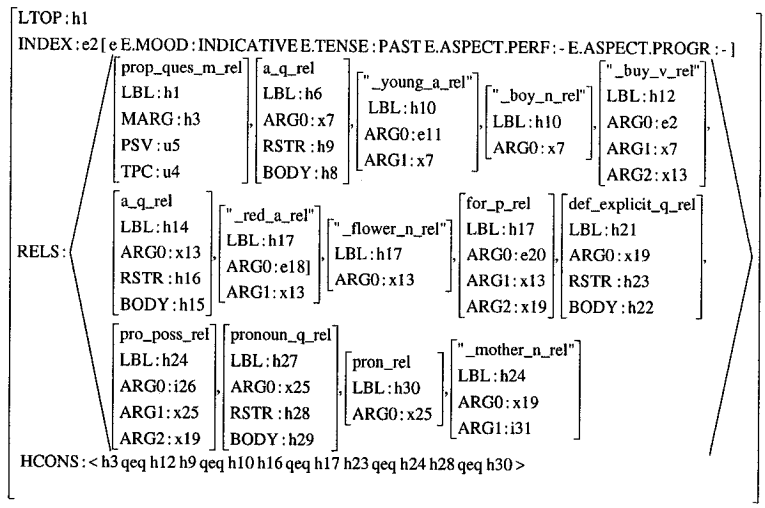


Figure 4.17: MRS representation of "A young boy bought a red flower for his mother"

```
Line
1 :  prop_ques_m_rel:→
2 :  MARG:go to h3
3 :  LBL = h3:→"_buy_v_rel":→
4 :                ARG0:→PAST,
5 :                ARG1:→3SG,
6 :                        go to h9:ARG0→ "_boy_n_rel"(x7);
7 :                        go to h9:ARG1→ "_young_a_rel"(x7); go to h6:ARG0-¿ _a_q_rel(x7);
8 :                                ARG2→3SG,
9 :                                        go to h16:ARG1→ "_red_a_rel"(x13);
10:                                       go to h16:ARG1→ _for_p_rel(x13);
11:                                               ARG2→3SG,
12:                                                       go to h23:ARG0→ "_mother_n_rel"(x19);
13:                                                       go to h23:ARG2→ pro_poss_rel(x19);
14:                                                               ARG1→3SG,
15:                                                                       go to h28:ARG0→ pron_rel(x25);
16:                                                                       go to h27:ARG0→ pronoun_q_rel(x25);
17:                                                       go to h21:ARG0→ def_explicit_q_rel(x19);
18:                                       go to h16:ARG0→ "_flower_n_rel"(x13);
19:                                       go to h14:ARG0→ _a_q_rel(x13);
20:  PSV:→u5,
21:  TPC:→u4,
```

Figure 4.18: Scanning into the MRS representation following the reference numbers for the sentence "A young boy bought a red flower for his mother"

AVMs to represent the relations of the MRS components inside the sentence. This MRS representation is represented by a hierarchy in figure 4.20(a). Following the FCS's algorithm at line 3, the nodes or leaves of *a-relation*, *q-relation*, and *n-relation* at the same level are combined together as shown in figure 4.20(b). Using line 4, the *p-relation* node merges with the next node and moves to the upper level as illustrated in figure 4.20(c). Finally, the hierarchy is the flat tree structure and we can define words or groups of words for actor, act and actee parts. In this example, the actor

part is <a, young boy>, the act part is <bought> and the actee part is <[a red flower],[for his mother]>. The focus content structure is shown in figure 4.21. This content structure is the input of the FET analysis.

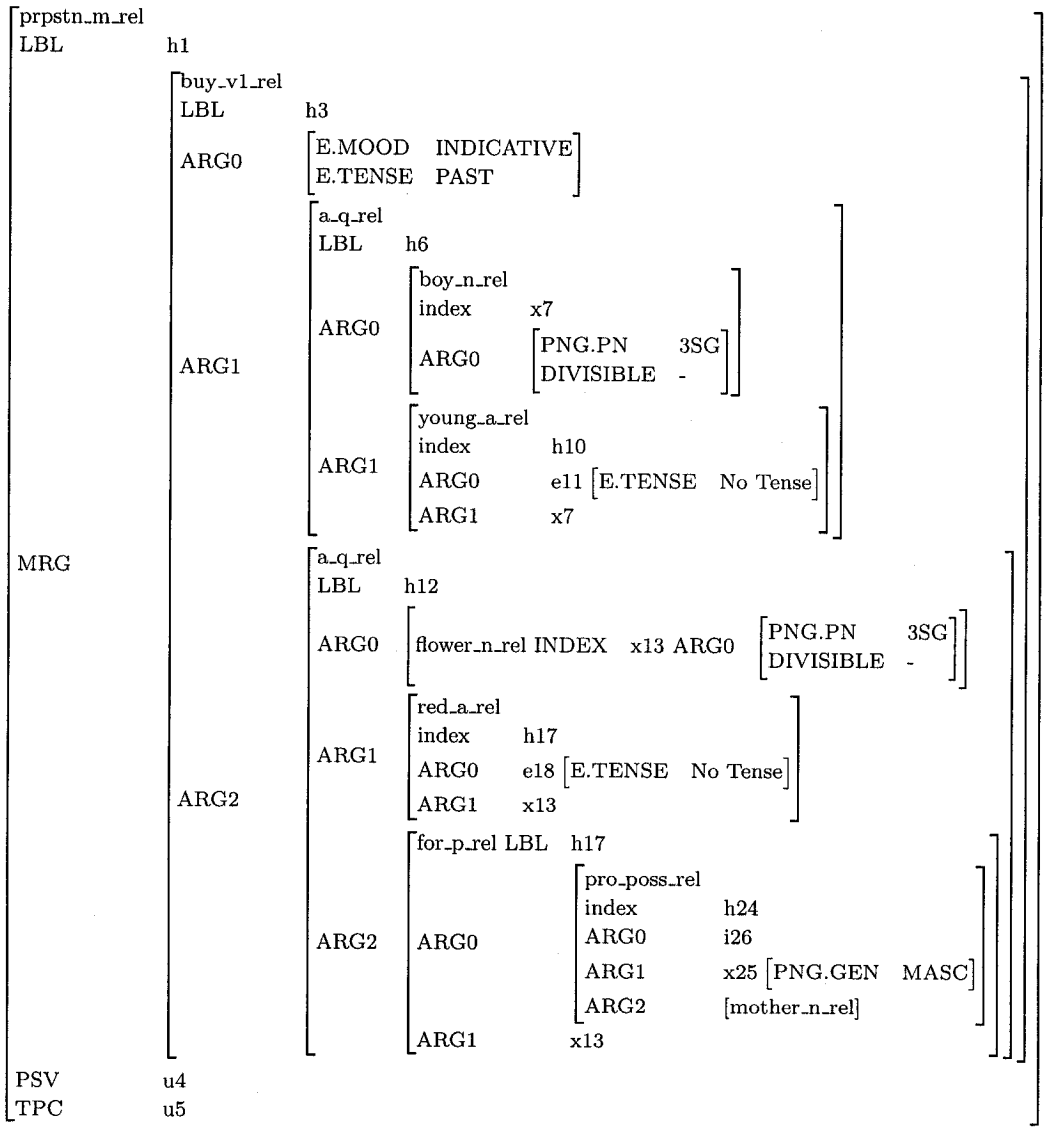


Figure 4.19: Transforming the MRS representation to the AVM for the sentence "A young boy bought a red flower for his mother"

4.5 Summary

In this chapter, we introduced the basic components for the LKB system including typed feature structure, constraints, grammar rules, and lexicon. The LKB system with the ERG grammar is a unification-based parser for English language. One of

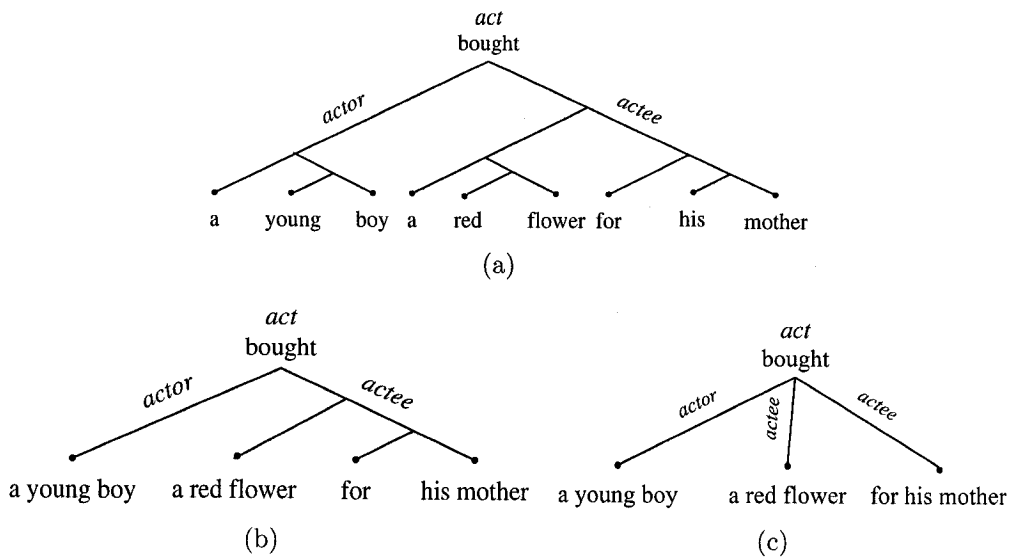


Figure 4.20: Trees of MRS representation: (a) hierarchy of MRS representation for the sentence "A young boy bought a red flower for his mother" (b) combining the *a-relation*, *q-relation*, and *n-relation* nodes together (c) collapse the *p-relation* and move to the upper level

the results from the LKB system is the sematic representation called MRS. Our focus analysis uses the MRS representation to define the focus parts and speaker's intention. We described how to transform the MRS representation to the focus content structure by using our technique called the focus content scoping in section 4.3. An example of this transformation is shown in section 4.4

$$\left\{ \begin{bmatrix} \text{Relation} & \text{actor} \\ \text{List-obj} & \langle \text{a, young, boy} \rangle \\ \text{Index} & \text{h9} \\ \text{RIndex} & \text{h3} \end{bmatrix}, \begin{bmatrix} \text{Relation} & \text{actee} \\ \text{List-obj} & \langle [\text{a, red, flower}], [\text{for, his, mother}] \rangle \\ \text{Index} & \text{h15} \\ \text{RIndex} & \text{h3} \end{bmatrix}, \begin{bmatrix} \text{Relation} & \text{act} \\ \text{List-obj} & \langle \text{buy} \rangle \\ \text{Index} & \text{h3} \\ \text{RIndex} & \{\text{h9,h15}\} \end{bmatrix} \right\}$$

Figure 4.21: FC structure of the sentence "A young boy bought a red flower for his mother"

Chapter 5

Focus to Emphasize Tone Analysis

The FET analysis is proposed to determine how the speaker's utterances are influenced by speaker's intentions. Focus information can be used to indicate a part of a sentence conveying a speaker's intention. Focus can scope the content that a speaker wants hearer to recognize. In this thesis, speech act is considered as a feature which involves a speaker's intention in a speech utterance. The relationships of focus parts and speech acts are analyzed to find prosodic patterns. The input of the FET analysis, including syntactic and semantic contents, is obtained from the preprocessing stage as the FC structure. In section 5.1, the general concepts of foci and tones are introduced to understand what is the relationship between them. Because of the different speaker's intentions, focus information needs to emphasize tone at a correct position to convey the precise meaning to hearer. A speaker's intention in a sentence must be revealed to determine focus information. Different focus positions depending on speaker's intentions are explained in section 5.2. The focus constraints are designed to analyze focus types and focus components. The descriptions of these constraints are in section 5.3. Finally, the details of the FET structure are explained in 5.4.

5.1 Introduction to Focus and Tone Analysis

In Spoken Language Understanding (SLU), the focus and tone information involve the interactions among three domains: syntactic, semantic, and intonation domains. Generally, the analysis of these interactions is based on two main architectures: syntax-intonation interaction and information-intonation interaction as shown in figure 5.1.

5.1.1 Syntax-Intonation Interaction

Human cognition coordinates processing of syntactic, semantic, and intonation domains, as shown in figure 5.1(a), to form a phrase or sentence, define the meaning and utter the spoken sound. Several researchers [1, 64] explain how interactions and relationships among these domains develop into SLG.

Syntax-intonation interaction, proposed by Steedman [65], provides the speaker's

utterance knowledge. This interaction basically distinguishes the accent as strong, weak or no accents in a sentence. Considering the syntactic and prosodic structures, the syntactic structure is more complicated than the prosodic structure. The syntactic structure is represented in a hierarchical tree with branches of different depths called a syntactic tree. An example syntactic tree is shown in figure 5.2(a). The prosodic structure can be represented as a prosodic tree [66], which has only one level of depth and leafs are spread horizontally as shown in figure 5.2(b). Each node of the prosodic tree can be labeled by accent tone marks such as strong accent (*strong*), weak accent (*weak*) or tone marks (H* and L-L%). Making the prosodic tree by manipulation of the syntactic tree is an interaction between syntactic and intonation domains and is heavily syntax-driven. This method increases complexity since a type hierarchy must cross-classify prosodic phrases under syntactic phrases.

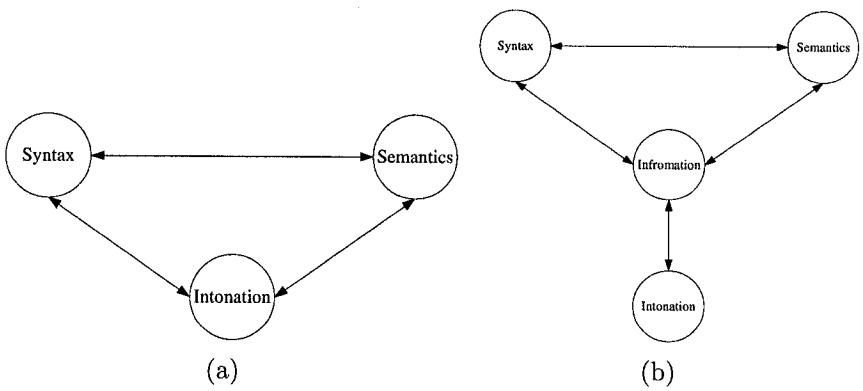


Figure 5.1: Comparison between (a) architecture for spoken language understanding, and (b) revised architecture for spoken language understanding

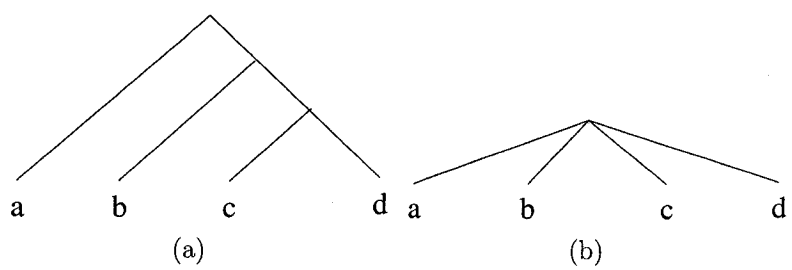


Figure 5.2: Comparison between (a) syntactic tree, and (b) prosodic tree

5.1.2 Information-Intonation Interaction

Syntactic-semantic interaction is frequently used for many NLP research tasks such as parsing and text analysis. The syntactic and semantic domains together are used to develop an information domain which is the intermediate level between syntax-semantic domain and intonation domain as shown in figure 5.1(b). In the information domain, the information structures are manipulated based on syntactic and semantic knowledge, as well as, focus and prosodic information. The information structure can help to improve the intonation analysis and promote a more precise prediction for a speech utterance.

In this thesis, the research focus is on the interactions between intonational and information structures. Avoiding interaction between syntax and intonation domains directly reduces the complexity of parsing. The semantics-information and the information-intonation interactions are considered in the FET analysis as shown in figure 5.3. We need to understand the relationships of semantics, focus, and intonation requiring the investigation of Focus-Accent theory [67]

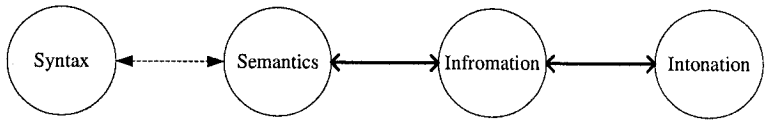


Figure 5.3: Semantic-Intonation interaction for SLU

5.1.3 Focus Theories for Tone Emphasis

To predict the intonation, the semantic and discourse information are analyzed to determine what parts in a sentence are in focus and how to emphasize prosody for speaker's utterance. Some semantic knowledge is considered to be a part of discourse relations. Since the target is intonation annotation for the speaker's utterance, the most significant information in a sentence needs to be distinguished by emphasizing tone. There are many theories to analyze the focus content (or the significant content) in a sentence such as the Belief and Design Inference (BDI) theory [68], given-new theory [69], and theme-rheme theory [70]. Some theories [70, 69] are based on psychological concepts but the recent theory described in [68] is based on the notions of information for communication.

The comparison of three main focus theories which can operate to Focus-Accent theory is shown in table 5.1. The BDI system divides a sentence into three parts in semantic domain. These parts are actor, act and actee parts. The *actor* part represents a person or a thing that acts something in a sentence, *act* represents an activity in that sentence, and *actee* represents the response of the activity. The given-new theory in the information domain separates a sentence into three parts: given, body and new parts. The given part is the background information while the new part is the information which never occurs in the background information. The body part is the event that relates the given and new parts. The theme-rheme theory is in the information domain. The theme information can be marked as the strong accent, while the rheme information is marked with the weak accent or no accent.

Table 5.1: Comparing three models: Theme-Rheme theory, Given-New information theory, BDI theory

BDI	ACTOR	ACT	ACTEE
Given-New	Given	Body	New
Theme-Rheme	Theme	Rheme	

Considering these theories in table 5.1, the actor of BDI in the semantic domain are the given and theme parts in the information domain. The act and actee parts of BDI are the body and the new parts of given-new theory. Consequently, the BDI, based on the notions of information, can be developed for use with Focus-Accent theory in the information domain. This theory expresses what are more and less important contents in a sentence, and we focus upon the important content.

The speech acts are used to explain the states of the speaker's intentions. The speech acts relate to discourse and transfer the knowledge from the semantic to the intonation domains. There are many speech act types such as suggestion, question, assertion, order, and so on. The relationships of sentence moods, speech acts and tones are exploited for prosodic prediction. Using speech acts not only defines what types of speech acts occur in a sentence but also defines who acts and is acted upon in a sentence as actor and actee. For instance, "Mary buys a book" is represented by *buy(X, Y)*; X is "Mary" (actor) and Y is "a book" (actee). The relationships of speech acts and tones are determined to select the possible intonations depending on focus parts in a sentence.

5.2 Defining Focus in a Sentence

Focus information is used to describe how to define a significant part in the sentence that the speaker and listener must heed. The concept of foci can be considered to express the prosodic phenomena, which is further information in addition to written content. A focus part is defined as a part of the speaker's interest in a sentence and is annotated with prosodic marks. For example in table 5.2, the sentence "Mary bought a book about bats" has different focus positions. Based on the focus theory, a sentence is composed of two parts; focus and non-focus. The focus projection defines a part of sentence as a focus (see figure 5.4). It can be manipulated within the syntactic/semantic structure. The focus projection distinguishes a part of the sentence, and marks the strong accent on that part. This focus part is that speaker intends the hearer to recognize, when the speaker utters a sentence. The different focus parts can convey different speaker's intentions to the hearer as shown table 5.2. In this section, an example sentence with different focus positions is presented depending on the different speaker's intentions. This sentence has four possible focus positions. Each focus position conveys a speaker's intention to hearer.

Table 5.2: Different foci in the example sentence

No.	**Focus**	**Speaker intend to focus at...**
i	$[MARY]_F$ bought a book about bats.	Who bought a book about bats?
ii	Mary bought [a BOOK about bats]$_F$.	What did Mary buy?
iii	Mary [BOUGHT a book about bats]$_F$.	What did Mary do?
iv	Mary bought a book about bats	What happened?

Instead of deriving focus projection from syntactic structure, the semantic information, derived from an HPSG parser, is analyzed to define the focus parts in the sentence as described in chapter 4. The focus parts (actor, act and actee) are determined from the semantic structure as shown in figure 5.5. The focus parts are the components of predicate logic, based on the BDI theory. These parts also refer to the speaker's intentions in the subjects of a proposition. Generally, the speaker utters the content at a focus part with different levels of emphasizing tones. The most emphatic tone is notified as a primary focus which is specified at the most significant content in the sentence. The rest of emphatic tones are a local focus. The hearer can recognize the local focus when the speaker utters a sentence. To define the focus

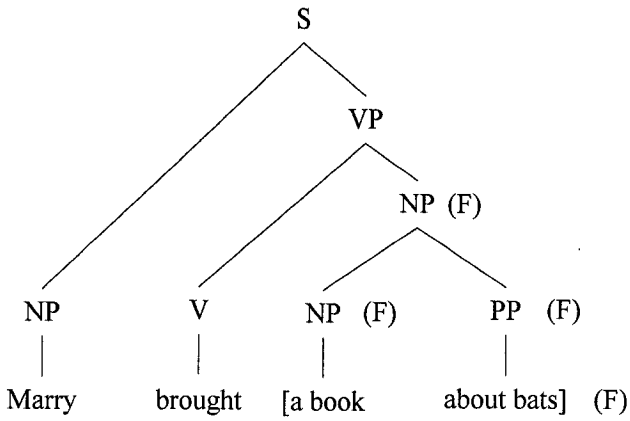


Figure 5.4: Focus projection on syntactic tree

types, the focus model in this thesis must acknowledge two types of foci: *w-focus*, and *s-focus*. The *w-focus* represents wide focus, that focus covers a word, a list of words or a phrase. The *s-focus* represents single focus, that the focus is placed on individual word in a list of words. The details of *w-focus* and *s-focus* are described in section 5.3.

$$\begin{bmatrix} \text{ACT} & \langle \text{buy} \rangle \\ \text{ACTOR} & \langle \text{Mary} \rangle \\ \text{ACTEE} & \langle \text{a, book, about, bat} \rangle \end{bmatrix}$$

Figure 5.5: Focus Parts of "Mary bought a book about bats"

The FET system defines the focus part of a given sentence. For example, "Mary bought a book about bats" is defined a focus part as shown in figure 5.6. In this figure, the focus part are assigned to actor, act, and actee parts. The focus can be placed on different parts of the sentence depending on the speaker's intentions. The focus part is the place in which the strong accent is marked. If a speaker emphasizes the strong accent at the focus part, then hearer can perceive the content of the speaker's interest in the sentence. Normally, the speaker focuses at the content that the hearer must recognize. For example, in the sentence, "Mary bought a book about bats", there are different parts to serve as focus. In the example sentence, five possible focus situations depending on the speaker's intention are described below.

$$\left\{\begin{bmatrix} Relation & actor \\ List\text{-}obj & \langle Mary \rangle \\ Index & r_1 \\ RIndex & s_1 \end{bmatrix}, \begin{bmatrix} Relation & actee \\ List\text{-}obj & \langle [a,\ book], [about,\ bat] \rangle \\ Index & r_2 \\ RIndex & s_2 \end{bmatrix}, \begin{bmatrix} Relation & act \\ List\text{-}obj & \langle buy \rangle \\ Index & s_1 \\ RIndex & \{r_1, r_2\} \end{bmatrix}\right\}$$

Figure 5.6: FC structure of "Mary bought the book about bats"

Focus on Actor. In this situation, the speaker wants the listener to recognize who bought a book about bats. The speaker focuses on the actor part "Mary" and Mary is a person who bought a book about bats. In figure 5.7, the *List–obj* contains a word, or list of words that must be in focus. *FET–obj* is the list of words that must be emphasized by tone and are marked as the strong accent. *Relation* declares what is the focus part and its relation in the sentence. *Index* is an index number of the object while *RIndex* is an index number of the related object. In this situation, the *List–obj* and *FET–obj* contain the same word "Mary".

Sentence: $[MARY]_F$ `bought a book about bats.`

$$\begin{bmatrix} w\text{-}focus & \\ Relation & actor \\ List\text{-}obj & \langle Mary \rangle \\ Index & r_1 \\ RIndex & r_2 \\ FET\text{-}obj & \langle Mary \rangle \end{bmatrix}$$

Figure 5.7: Wide focus at actor "Mary"

Wide Focus on Actee. In this situation, the speaker wants the hearer to recognize what Mary buys. The speaker focuses on the actee part "a book about bats" which is a thing that Mary buys. In figure 5.8, the *List–obj* contains two lists of words: `[a, book]` and `[about, bats]` while the *FET–obj* merges two list in the *List–obj* together as `[a, book, about, bats]`.

Single Focus on Actee. In this situation, a single focus can specify in details what the speaker wants hearer to recognize. Since the speaker wants to specify the focus on actee "a book about bats" then there are two cases in which the speaker can

Sentence: Mary bought [a book about bats]$_F$.

$$\begin{bmatrix} \textit{w-focus} & \\ \textit{Relation} & \textit{actee} \\ \textit{List-obj} & \Big\langle \big[a,\ book\big], \big[about,\ bats\big] \Big\rangle \\ \textit{Index} & r_2 \\ \textit{RIndex} & s_1 \\ \textit{FET-obj} & \big\langle a,\ book,\ about,\ bats \big\rangle \end{bmatrix}$$

Figure 5.8: Wide focus at actee "a book about bats"

focus, depending on the intentions of the speaker, as shown in figure 5.9. The speaker can choose to focus "a book" or "about bats". In our system, an *s-focus* contains a single word from a list of words or phrase in the sentence. Therefore the last element of the list of words is defined as a primary focus to emphasized tone. The *s-focus* structure is illustrated in figure 5.10. In this figure, if the *List–obj* contains a list of words $a_1, a_2, \ldots, a_n$ then the last element of this list is declared as an *s-focus* and the *FET–obj* contains only a_n

$$\begin{bmatrix} \textit{s-focus} & \\ \textit{Relation} & \textit{actee} \\ \textit{List-obj} & \Big\langle \big[a,\ book\big], \big[about,\ bat\big] \Big\rangle \\ \textit{Index} & r_2 \\ \textit{RIndex} & s_1 \end{bmatrix} \rightarrow$$

$$\begin{bmatrix} \textit{s-focus} & \\ \textit{Relation} & \textit{actee} \\ \textit{List-obj} & \big\langle a,\ book \big\rangle \\ \textit{Index} & r_2 \\ \textit{RIndex} & s_1 \\ \textit{FET-obj} & \big\langle book \big\rangle \end{bmatrix} \vee \begin{bmatrix} \textit{s-focus} & \\ \textit{Relation} & \textit{actee} \\ \textit{List-obj} & \big\langle about,\ bat \big\rangle \\ \textit{Index} & r_2 \\ \textit{RIndex} & s_1 \\ \textit{FET-obj} & \big\langle bat \big\rangle \end{bmatrix}$$

Figure 5.9: Splitting two cases of the single focus of actee

Case 1: Single Focus on "about bats". In the first case, the speaker wants hearer to recognize detail about the book. The speaker focuses on "bats" and "bats" is the topic of the book that Mary buys. Because the speaker wants to mention "a book about bats", the focus type requires a primary focus and must be only a single

$$make\_s\text{-}focus:\left[List\text{-}obj \quad \langle a_1, a_2, ..., a_n \rangle\right] \rightarrow \begin{bmatrix} s\text{-}focus & \\ Relation & focus\_part \\ List\text{-}obj & \langle a_1, a_2, ..., a_n \rangle \\ Index & r_i \\ RIndex & s_j \\ FET\text{-}obj & \langle a_n \rangle \end{bmatrix}$$

Figure 5.10: Focus at the last element

word that hearer can recognize. The *FET-obj*, contains only the word "bats" which came from the phrase "a book about bats" as shown in figure 5.11. This focus is declared as *s-focus*.

Sentence: Mary bought a book about $[BATS]_F$.

$$\begin{bmatrix} s\text{-}focus & \\ Relation & actee \\ List\text{-}obj & \langle about, bats \rangle \\ Index & r_2 \\ RIndex & s_1 \\ FET\text{-}obj & \langle bats \rangle \end{bmatrix}$$

Figure 5.11: Single focus at "about bats"

Case 2: Single Focus on "a book". In the second case, the speaker wants hearer to recognize what Mary bought about bats. The focus is at "book". The speaker scopes content into detail in "a book about bat". Therefore, "book" is an *s-focus*. See figure 5.12.

Sentence: Mary bought a $[BOOK]_F$ about bats.

$$\begin{bmatrix} s\text{-}focus & \\ Relation & actee \\ List\text{-}obj & \langle a, book \rangle \\ Index & r_2 \\ RIndex & s_1 \\ FET\text{-}obj & \langle book \rangle \end{bmatrix}$$

Figure 5.12: Single focus at "a book"

Focus on Act. In this situation, the speaker wants hearer to recognize what Mary does. The speaker focuses on "bought a book about bats". "bought" is the act part that Mary performs and the related content of "bought" is "a book about bats" which is the actee part. This focus shows what Mary does. In figure 5.13, there are two components of focus parts: act and actee parts. The act part "buy" is the *s-focus* while the actee part "a book about bats" defines the focus type as *w-focus*.

Sentence: Mary [bought a book about bats]$_F$.

$$\left[\begin{array}{ll} \textit{w-focus Relation} & \{act, actee\} \\ \textit{List-obj} & \langle [buy], [a,\ book,\ about,\ bats] \rangle \\ \textit{Index} & s_1 \\ \textit{RIndex} & \{r_1, r_2\} \end{array}\right] \rightarrow$$

$$\left\{ \left[\begin{array}{ll} \textit{s-focus} & \\ \textit{Relation} & act \\ \textit{List-obj} & \langle buy \rangle \\ \textit{Index} & s_1 \\ \textit{RIndex} & \{r_1, r_2\} \\ \textit{FET-obj} & \langle buy \rangle \end{array}\right], \left[\begin{array}{ll} \textit{w-focus} & \\ \textit{Relation} & actee \\ \textit{List-obj} & \langle [a,\ book], [about,\ bats] \rangle \\ \textit{Index} & \{r_1, r_2\} \\ \textit{RIndex} & s_1 \\ \textit{FET-obj} & \langle a,\ book,\ about,\ bats \rangle \end{array}\right] \right\}$$

Figure 5.13: Wide focus at "bought a book about bats"

No Focus. If the speaker's focus is over all content in the sentence, then the focus will cover the entire sentence. In the prosodic system, the speaker cannot emphasize tone over the entire sentence. Therefore this situation has no focus to emphasize tone.

5.3 Focus Constraints and Rules

Based on the analysis of semantic information, we now consider the focus parts. Three focus parts are defined as actor, act, and actee. If a focus is on a sentence then the focus can be on actor, act, actee or their combinations as shown in figure 5.14. For instance, "Mary bought a book about bats" can have focus at actor part "Mary" (case 4: ''[Mary]$_F$ bought a book about bats."), actee part "a book about bats" (case 7: ''Mary bought [a book about bats]$_F$") and so on. Figure 5.14 shows eight possible combinations of foci.

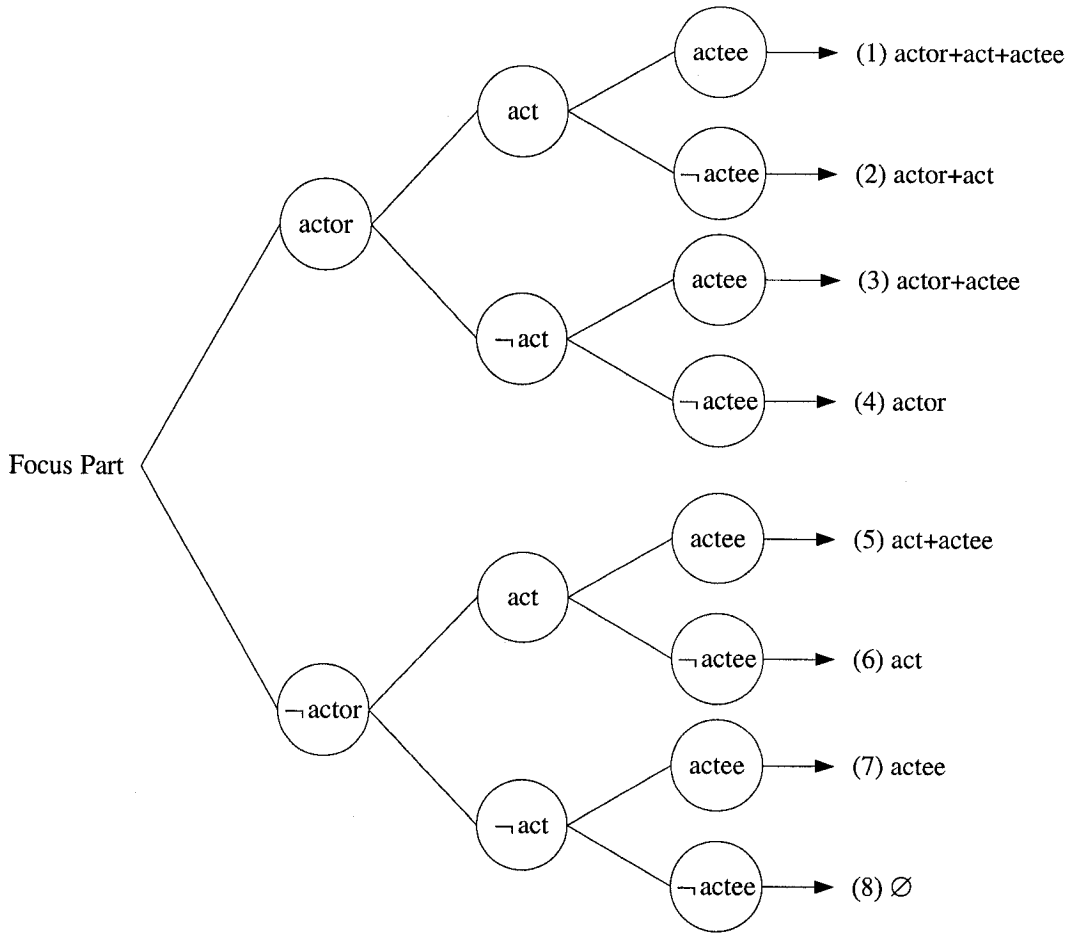


Figure 5.14: Focus part on semantic information

The actor and actee parts can be assigned as a single focus (*s-focus*) or wide focus (*w-focus*) while the act part is labeled only as an *s-focus* as shown in figure 5.15. Generally, the focus does not cover only the act part. If the focus covers the act part, then the focus must cover at least one of the related parts (actor or actee).

The focus types are defined following the situations as shown in table 5.3. A focus covers a sentence based on the different situations. If the focus covers the whole sentence then the focus types define *w-focus* at the actor part, *s-focus* at the act part and *w-focus* at the actee part (see line no. 1) or the focus is not defined. If the focus covers the actor part, then the focus type is defined as *w-focus* or *s-focus* (line no. 4). One case that does not define the focus types is no focus part (∅) as shown in line no. 8.

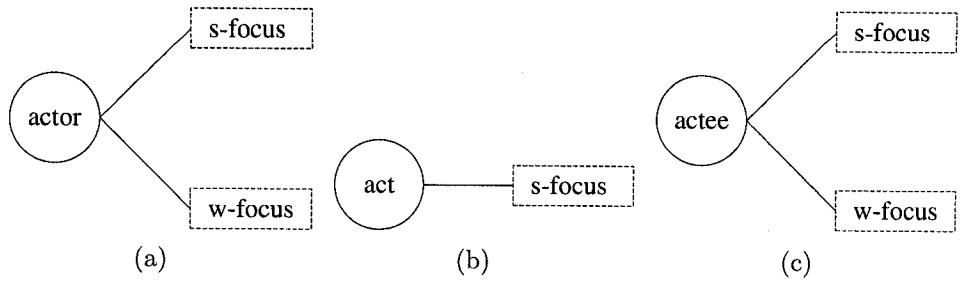


Figure 5.15: Focus types for each semantic part: (a) actor, (b) act, and (c) actee

Table 5.3: Focus parts and focus types

Line No.	Group	Focus Parts	Focus Types
1	A	actor+act+actee	w-focus(actor),s-focus(act), w-focus(actee), or undefined
2	B	actor+act	w-focus(actor),s-focus(act)
3	C	actor +actee	undefined
4	D	actor	w-focus(actor) or s-focus(actor)
5	E	act+actee	s-focus(act),w-focus(actee)
6	F	act	s-focus(act)
7	G	actee	w-focus(actee) or s-focus(actee)
8	H	$\oslash$	undefined

We define the constraints to select the focus types following the different situations. The constraints of focus are categorized to five cases.

(a) An s-focus on the actor or actee parts. The last node in the list of objects is defined as the focus position to emphasize tone (*FET–obj*), see figure 5.16(a). For example, <a, red, book> is the list of objects in the actee part. The *FET–obj* is assigned <book> as the focus position to emphasize tone as shown in figure 5.16(b).

(b) A w-focus on the actor or actee parts. The list of objects is equal to the *FET–obj* in the sentence as shown in figure 5.17(a). Since the *w-focus* is marked at the actee part "a book about bats" then *FET–obj* is equal to the list of objects <a,book,about,bats>, see figure 5.17(b).

$$make\_s\text{-}focus: \begin{bmatrix} s\text{-}focus & \\ Relation & focus\_part \\ List\text{-}obj & \langle a_1, a_2, ..., a_n \rangle \\ Index & r_i \\ RIndex & s_j \end{bmatrix} \rightarrow \begin{bmatrix} s\text{-}focus & \\ Relation & focus\_part \\ List\text{-}obj & \langle a_1, a_2, ..., a_n \rangle \\ Index & r_i \\ RIndex & s_j \\ FET\text{-}obj & \langle a_n \rangle \end{bmatrix}$$

(a)

$$make\_s\text{-}focus: \begin{bmatrix} s\text{-}focus & \\ Relation & focus\_part \\ List\text{-}obj & \langle a,\ red,\ book \rangle \\ Index & r_i \\ RIndex & s_j \end{bmatrix} \rightarrow \begin{bmatrix} s\text{-}focus & \\ Relation & actor \\ List\text{-}obj & \langle a,\ red,\ book \rangle \\ Index & r_1 \\ RIndex & s_2 \\ FET\text{-}obj & \langle book \rangle \end{bmatrix}$$

(b)

Figure 5.16: *s-focus* structure: (a) marking *s-focus* and (b) an example of marking *s-focus* at the actee part "a, red, book"

$$make\_w\text{-}focus: \begin{bmatrix} w\text{-}focus & \\ Relation & focus\_part \\ List\text{-}obj & \langle a_1, a_2, ..., a_n \rangle \\ Index & r_i \\ RIndex & s_j \end{bmatrix} \rightarrow \begin{bmatrix} w\text{-}focus & \\ Relation & focus\_part \\ List\text{-}obj & \langle a_1, a_2, ..., a_n \rangle \\ Index & r_i \\ RIndex & s_j \\ FET\text{-}obj & \langle a_1, a_2, ..., a_n \rangle \end{bmatrix}$$

(a)

$$make\_w\text{-}focus: \begin{bmatrix} w\text{-}focus & \\ Relation & actee \\ List\text{-}obj & \langle a,\ book,\ about,\ bats \rangle \\ Index & r_3 \\ RIndex & s_1 \end{bmatrix} \rightarrow \begin{bmatrix} w\text{-}focus & \\ Relation & actee \\ List\text{-}obj & \langle a,\ book,\ about,\ bats \rangle \\ Index & r_3 \\ RIndex & s_1 \\ FET\text{-}obj & \langle a,\ book,\ about,\ bats \rangle \end{bmatrix}$$

(b)

Figure 5.17: *w-focus* structure: (a) marking *w-focus* and (b) an example of marking *w-focus* at the actee part "a, red, book"

(c) A w-focus on actor or actee parts containing the multiple lists of objects. The lists are merged together to be the *FET–obj* as shown in figure 5.18.

For example, if the actee part is "a book about bats in the black forest", then the lists of objects are <[a, book], [about, bats], [in, the, forest]> and the *FET–obj* can is assigned as <a, book, about, bats, in, the, black, forest>.

$$merge\_list\_w\text{-}focus: \left[List\text{-}obj \left\langle \begin{matrix} [a_1, a_2, ..., a_n], \\ [b_1, b_2, ..., b_n], ..., \\ [m_1, m_2, ..., m_n] \end{matrix} \right\rangle \right] \rightarrow \left[\begin{matrix} w\text{-}focus & \\ Relation & focus\_part \\ List\text{-}obj & \left\langle \begin{matrix} [a_1, a_2, ..., a_n], [b_1, b_2, ..., b_n] \\ ..., [m_1, m_2, ..., m_n] \end{matrix} \right\rangle \\ Index & r_i \\ RIndex & s_j \\ FET\text{-}obj & \left\langle \begin{matrix} a_1, a_2, ..., a_n, b_1, b_2, ..., b_n \\ ..., m_1, m_2, ..., m_n \end{matrix} \right\rangle \end{matrix} \right]$$

(a)

$$merge\_list\_w\text{-}focus: \left[List\text{-}obj \left\langle \begin{matrix} [a, book], \\ [about, bats], \\ [in, the, black, forest] \end{matrix} \right\rangle \right] \rightarrow \left[\begin{matrix} w\text{-}focus & \\ Relation & actee \\ List\text{-}obj & \left\langle \begin{matrix} [a, book], [about, bats], \\ [in, the, black, forest] \end{matrix} \right\rangle \\ Index & r_3 \\ RIndex & s_1 \\ FET\text{-}obj & \left\langle \begin{matrix} a, book, about, bats, \\ in, the, black, forest \end{matrix} \right\rangle \end{matrix} \right]$$

(b)

Figure 5.18: Merge focus structure: (a) marking *w-focus* of multiple lists of objects and (b) an example of marking *w-focus* at the actee part "a, red, book"

(d) An s-focus on actor or actee parts containing the multiple lists of objects. If the focus type is an *s-focus* and there are m sets of lists of objects (multiple lists of objects), then these lists of objects can be split into the *s-focus* of each list of objects, as shown in figure 5.19(a). For example, if the multiple lists of objects of the actee part is <[a, book], [about, bats], [of, the, black, forest]> then the *FET–obj* requires a primary focus and must be <book>, <bats>, or <forest>, as shown in figure 5.19(b).

(e) A focus on the act part. Two cases of defining the focus types are shown in figure 5.20(a). In the first case, the *s-focus* is marked at the act part while the *w-focus* is marked at the actee part. In the second choice, the *s-focus* is marked at the act part and the *w-focus* is marked at the actor part. In our example, the act part is "bought" while the actor and actee parts are "Mary" and "a book", respectively.

Since, in the first case, the act part connects to actee part, the focus type of act part is an *s-focus* and the focus type of actee part is a *w-focus*. The *FET–obj* is assigned <bought> for the act part and <a,book> for the actee part. In the second case, the act part connects to the actor part. The focus type of act part is *s-focus* and the focus type of the actor part is the *w-focus*. The *FET–obj* is assigned <bought> for the act part and <Mary> for the actor part. (See figure 5.20(b).) These five cases cover all possible situations of focus for simple sentences.

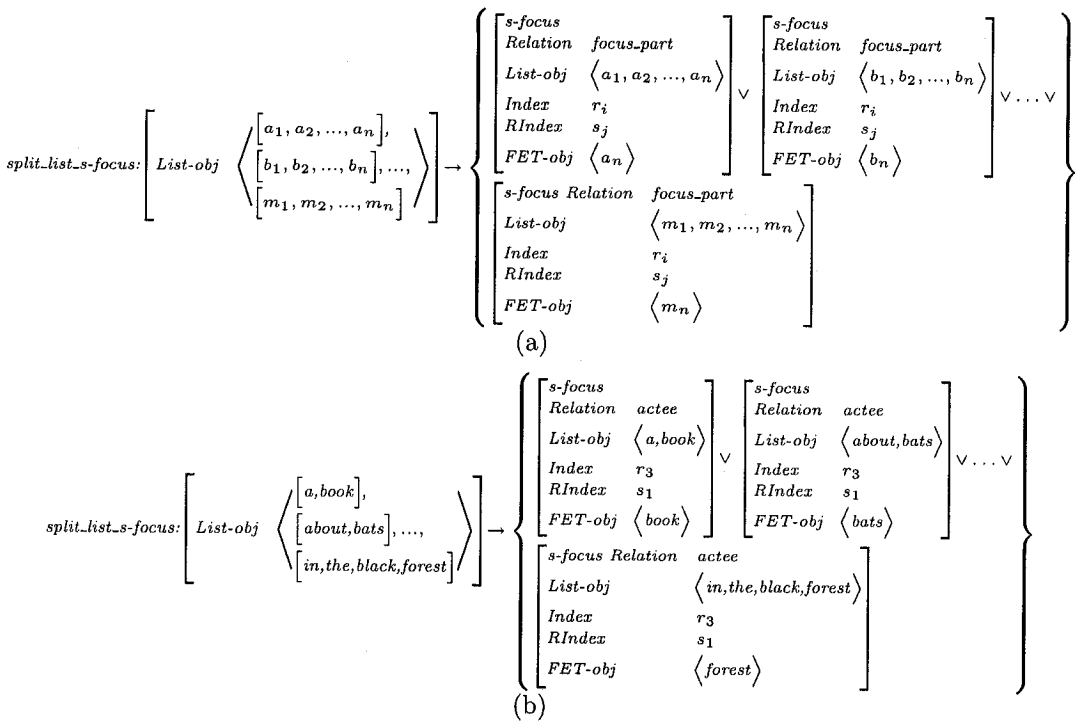


Figure 5.19: Merge focus structure: (a) marking *s-focus* of the multiple lists of objects and (b) an example of marking *s-focus* at the actor or actee parts

5.4 Focus to Emphasize Tone Structure

The FET structure is designed to contain the feature structure of focus and prosodic information. The prosodic structure is a subfeature structure of the focus information structure. The FET structure consists of three main features: the list of words (*List-obj*), focus information (*Focus–Info*), and prosodic information (*Prosody*), so that its structure is inside the *Focus–Info* structure as shown in figure 5.21. The details of FET structure for the LKB system are explained in chapter 7

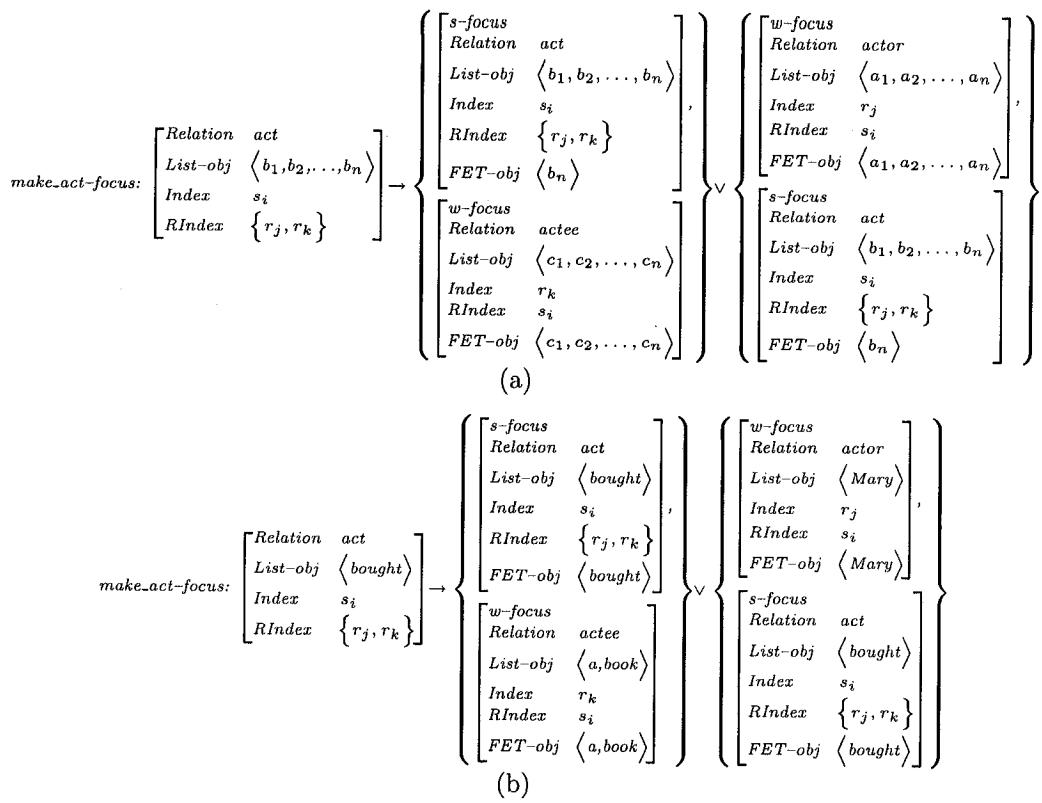


Figure 5.20: Merge focus structure: (a) marking *w-focus* for the act part and (b) an example of marking *w-focus* at the act part

5.4.1 Focus Information Structure

The *Focus–Info* structure uses focus and prosodic features. It includes the focus part and list of focus words (*FET-obj*) from the FC structure. The *Focus–Info* structure also contains the focus type (*FCType*) and focus group (*FCGroup*) described in table 5.4. *Focus–Part* represents actor, act, or actee part, while the *FCType* is *w-focus* or *s-focus*. These features are analyzed following the focus constraints in section 5.3. Inside the FET structure, the prosodic information structure (*Prosody*) is the significant subfeature structure of the *Focus–Info* structure because of it contains the information about relationships of speech acts and tones for each focus part. These relationships are described in chapter 6. The coarse *Focus–Info* structure is illustrated in figure 5.22

$$\textit{focus–word:}\begin{bmatrix}\textit{List–obj} & \langle a_1, a_2, \ldots, a_n \rangle \\ \textit{Focus–Info} & \begin{bmatrix}\textit{FET–obj} & \langle w_1, w_2, \ldots, w_m \rangle \\ \textit{Focus} & \begin{bmatrix}\textit{focus–part} & \\ \textit{Prosody} & [\,]\end{bmatrix}\end{bmatrix}\end{bmatrix}$$

Figure 5.21: Focus to emphasize tone structure

$$\textit{Focus–Info:}\begin{bmatrix}\textit{FET–obj} & \langle w_1, w_2, \ldots, w_m \rangle \\ \textit{Focus} & \begin{bmatrix}\textit{Focus–Part} & \textit{focus–part} \\ \textit{Focus–Pos} & \textit{focus–pos} \\ \textit{FCGroup} & \textit{fcgroup} \\ \textit{FCType} & \textit{fctype} \\ \textit{Prosody} & [\,]\end{bmatrix}\end{bmatrix}$$

Figure 5.22: Information structure

5.4.2 Prosodic Structure

The *Prosody* structure contains features of focus, speaker's intention, and prosody. Based on the relationship among these features we determine the tone marks. The *Prosody* structure is composed of sentence mood (*STMood*), speech act code (*SP-Code*), prosodic mark (*Prosody-Mark*), and tone mark (*Tone*), which includes accent tone (*Accent-Tone*) and boundary tone (*Bound-Tone*). The *STMood* contains the type of a sentence, such as affirmative sentence (*aff*), and is derived from the FC structure. The *SPCode* is adopted from [71], and it denotes the speech act code, such as intending (*EN0ab*). The prosodic information structure is illustrated in figure 5.23.

$$\begin{bmatrix}\textit{focus-part} & \\ \textit{Prosody} & \begin{bmatrix}\textit{STMood} & \textit{mood} \\ \textit{SPCode} & \textit{code} \\ \textit{Prosodic-Mark} & \begin{bmatrix}\textit{prosody-mark} & \\ \textit{Tone} & [\,]\end{bmatrix}\end{bmatrix}\end{bmatrix}$$

Figure 5.23: Prosodic information structure

In the prosodic domain, a word, marked with a tone, is called a prosodic word [1], while a word which does not have tone marks is called a leaner [1]. Tones are set as marked or unmarked for a word. Tone marks can be separated into two main groups:

Accent-Tone and *Bound-Tone* based on the ToBI representation, as shown in figure 5.24. The accent tone can occur at any part of the sentence except at the end of phrase or sentence, and does not include the duration symbol (- or %). The examples of accent tones are `H*`, `L*`, `L+H*`. The boundary tone occurs at the end of phrase or sentence, such as `L-`, `H-`, `L-L%`.

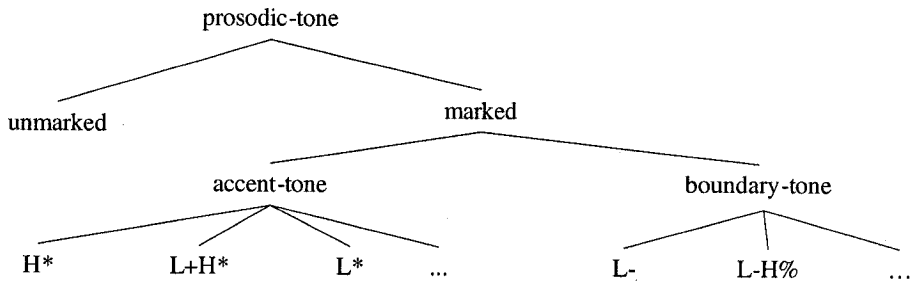


Figure 5.24: Tone tree

The result of our FET system is a set of tone marks annotated on the sentence. The *Prosodic-Set* structure in figure 5.25 is composed of two main features: prosodic function (*Prosody–Funct*), prosodic mark (*Prosody-Mark*). The *Prosody–Mark* is defined following the tone tree in figure 5.24. The *Prosody-Mark* structure represents two prosodic features: accent tone (*Accent–Tone*) and boundary tone (*Bound–Tone*).

The prosodic information is used to determine what tone should be labeled on a phrase or word in a sentence. The prosodic functions are separated into *marked-info* and *unmarked-info* as shown in figure 5.26. The structure *unmarked-info* means no prosodic information is to be found at that word. The structure *marked-info* marks a prosodic word which is attached by prosodic information. There are two types of *marked-info*: strong tone (*strong*) and weak tone (*weak*). The strong tone means that words are emphasized (*Em*), or highly emphasized (*hEm*) by high tones. The weak tone represents the word that has low priority or low necessity in the content and is de-emphasized by low tones.

Following the prosodic function tree, a prosodic function is assumed to a combination of accent and boundary tones and is represented in the *Prosodic-Set* structure (see figure 5.25(a)). For example, the phrase "this black book" is marked with the high emphasized tone at this phrase. The speaker expects the listener to recognize

$$\textit{Prosody-Set:}\begin{bmatrix}\textit{Prosody-Funct} & \\ \textit{Prosody-Mark} & \begin{bmatrix}\textit{Accent-Tone} & \textit{accent}\\ \textit{Bound-Tone} & \textit{bound}\end{bmatrix}\end{bmatrix}$$

(a)

$$\textit{Prosody-Set:}\begin{bmatrix}\textit{hem\_lg-break} & \\ \textit{Prosody-Mark} & \begin{bmatrix}\textit{Accent-Tone} & \textit{L+H*}\\ \textit{Bound-Tone} & \textit{L-L\%}\end{bmatrix}\end{bmatrix}$$

(b)

Figure 5.25: Prosodic structure: (a) prosodic structure with variables and (b) an example of prosodic structure

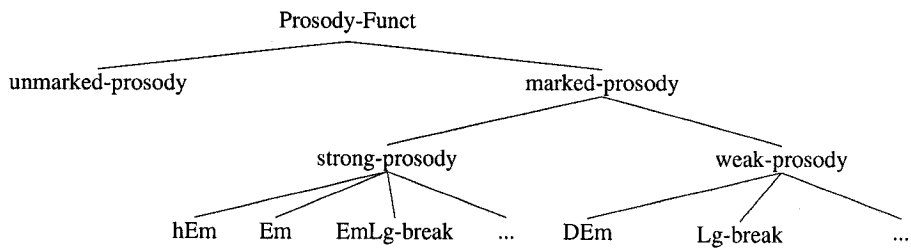


Figure 5.26: Prosodic function tree

this emphasis by using a strong accent and high tone as shown as figure 5.25(b).

The prosodic function represents a combination of accent and boundary tones which is explained in table 5.4. We can select high tone emphasis (*hEm*) for a focus word which is labeled by the `L+H*`. The different prosodic phenomena are explained by the different prosodic functions. For example, the prosodic phenomena for the Yes-No question sentence, requires a high tone at the end of the sentence. The prosodic function *EmLg-break* is selected to label `H-H%` on the last word of sentence, which always appears on a yes-no question sentence.

5.5 Summary

We analyzed focus conveyed by prosody in a sentence. The speaker's intention can be used to identify what part serves as the focus part. The focus content structure contains actor, act, and actee parts. Our FET analysis uses a constraint-based approach. Five focus constraints are explained in section 5.3. These constraints are designed to control the focus information including focus parts and focus types (*s-focus* and

Table 5.4: Mapping prosodic marks and accent-boundary tones

Prosody-Mark	**Accent-Tone**	**Bound-Tone**
dEM_Sh-break	L*	L-
dEm	L*	nobound
dEm_EmSh-break	L*	H-
dEm_EmLg-break	L*	H-H%
Sh-break	noaccent	L-
hEm_Sh-break	L+H*	L-
hEm	L+H*	nobound
sEm_Lg-break	L*	L-L%
Lg-break	noaccent	L-L%
hEM_Lg-break	L+H*	L-L%
Em_EmLg-break	H*	H-H%
hEm_EmLg-break	L+H*	H-H%
hEm_EmSh-break	L+H*	H-
EmLg-break	noaccent	H-H%
EmSh-break	noaccent	H-
Em	H*	nobound
Em_EmSh-break	H*	H-
Em_Lg-break	H*	L-L%
Em_Sh-break	H*	L-L%
no-mark	noaccent	nobound

w-focus). The FET structure, described in 5.4, is built to support our focus analysis and focus constraints. This structure contains focus, speaker's intention, and prosodic information.

Chapter 6

Prosodic Generation using Speech Acts and Foci

The basis of the FET analysis is derived from the relationship of speech acts and tones depending on the focus parts. In the previous chapter, we explain how the explanation of focus phenomena in a sentence is used to determine what are the focus parts and their components, including the focus types, focus groups, and so on. For each focus part, the relationships of the speech acts and tones are investigated and how these relationships interact to identify the tone patterns for prosodic annotation. The categories of speech acts and the tone representation system are introduced in section 6.1. The tone patterns are gathered from two datasets: automatic tone annotation and manual tone annotation datasets. The specifications of these datasets and comparison between the datasets are described in section 6.2. In section 6.3, speech act analysis by considering prosodic phenomena is expressed to find prosodic information for the FET structure. Designing the FET structure and its components is explained in section 6.4.

6.1 Introduction to Speech Acts and Tone Marks for Focus to Emphasize Tone System

Speech act is a feature which is used to study the speaker's intentions. It is also considered to indicate the focus parts. If there is a focus in the sentence, the emphasized tones must be marked at the focus part of the sentence. Since the speech acts involve the focus to emphasize tone analysis, the speech act represents the speaker's intention conveying the prosody to hearer. Consequently, speech acts can help to annotate tone marks for the FET system.

Speech Acts

In defining the speech act types, the speech act classification for the FET analysis is based on Baller's constraints [71]. He classifies the speech acts into four main categories: *expression*, *appeal*, *interaction*, and *discourse*. Ballmer and Brennenstuhl [71] defines that *expression* is an often uncontrolled mirroring of emotional states of a human being. *Appeal* is a linguistic function clearly directed towards a hearer. It

is unidirectional from speaker to hearer and the speaker tries to some extent to get control over the hearer. Both *expression* and *appeal* are monologues. On the other hand, Ballmer and Brennenstuhl [71] describe *interaction* as the linguistic function involving speaker and hearer in mutual verbal action. *Discourse* is the higher linguistic function and can become operative. Both interaction and discourse are dialogues. Generally, *expression* is represented by the emotional model while the *discourse* model is originally derived from theme-rheme theory. The main speech act categories are shown in figure 6.1 [71]. For the FET analysis, only one direction from speaker to hearer is considered in the relationships of speech acts and tones. The *appeal* and *interaction* categories have the most impact on the FET system based on the act part.

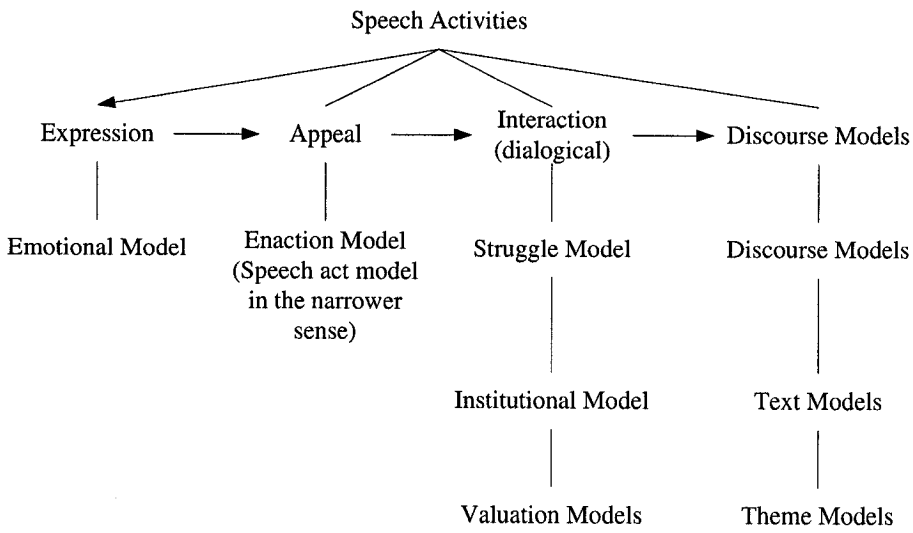


Figure 6.1: Speech act categories

ToBI Representation

Tone and Break Indexing (ToBI) is a standard of prosodic representation system which is invented by [5]. The ToBI representation is composed of two main groups of tone marks separated by time break after tone marked on a word or phrase. The first group is no time break. The tone marks can govern only a word, called "Accent Tone". The accent tones are represented by `H*` (peak accent), `L*` (low accent), `L+H*` (rising peak accent), and L*+H (scooped accent). The second group is short and long time breaks. The boundary tone marks cover a word or a group of words (phrase).

Boundary tones are represented by these symbols L-, H-, L-L%, H-L%, L-H%, H-H%. The descriptions of these prosodic marks in ToBI system has been reported in chapter 3 and the complete ToBI system is described in [5].

6.2 Investigation of Tone Patterns Depending on Focus Parts

The relationships of speech acts and prosodic marks are analyzed to identify tone patterns. The knowledge of (i) intonation theories and (ii) pattern learning from the tone annotation corpus are required for our analysis. In this section, the tone annotation corpus is explored to capture the common tone patterns for each speech act type depending on the focus components. Two datasets, used in this investigation, are an automatically tone annotated dataset and a manually tone annotated dataset. Automatically tone annotated dataset is a part of the CMU-COM corpus from Carnegie Mellon University (CMU) [55]. Investigating the tone patterns for each speech act in automatically tone annotated dataset is described in section 6.2.1. Manually annotated tone dataset is derived from the OSULL [72]. Learning the tone patterns for each speech act in manual annotation dataset is explained in section 6.2.2. Furthermore, the intonation theory is discussed and the theory can be used to identify tone patterns for each speech act and how the intonation theory can be used to find the relationships of speech acts and tones depending on focus parts in section 6.2.3.

6.2.1 Tone Patterns in Automatically Tone Annotated Dataset

The CMU-COM corpus is composed of a number of sentences with the ToBI marks. This corpus is gathered from dialogues in traveling reservation domain. Labeling ToBI marks in this corpus is done by an automatic ToBI marking system by using a procedure of the Sphinx III speech recognition system [73]. A hundred sentences of this corpus are analyzed using the LKB parsing system. Sixty two sentences can be parsed by the LKB system and it produces the syntactic trees and the MRS representations by parsing these sentences. The MRS representations are transformed to the FC structures, which include the actor, act, and actee parts.

Based on the speech act classification [71], the speech act codes are inferred from

the act part of the FC structure. The act part defines what the speech act categories can be for each sentence. A sentence can be marked by more than one code corresponding to semantic information. We label the speech act codes on sixty two sentences. An example of marking speech act code is shown in table 6.1. In this table, the speech act codes presented in the second column and the main verbs in the third column must correspond to each other. In the second column, *EN0aa* represents "wishing" and *EN0ab* represents "intending". The last column represents the types of sentences. S represents the non-question sentence while *y-n-q* means "Yes-No question" and *wh-q* is "WH-question".

Table 6.1: Speech act code with verb and tone marks

Line No.	Speech Act Code	Verb	Tone Marks (Act part)	POS	Tone Marks (Actor part)	Tone Marks (Actee part)	Sentence Types
1	EN0aa, EN0ab	would like to	H*	P	H*	H-H%	y-n-q
2	EN0ab	would like to	-	V	H*	L-L%	wh-q
3	EN0ab	need	H*	STEM	H*	!H* L-L%	s(if)
4	EN0ab	have (profile)	-	STEM	∅	L+H*	s
5	EN0ab	have (place)	-	STEM	H*	L+H* L-L%	s
6	EN0ab	plan	-	STEM/NP	H*H*	H-H%	y-n-q

To retrieve tone patterns from this dataset, some clues can be discussed considering categories of speech acts and these clues can help us to define the relationships of speech acts and tone patterns. Analyzing the relationships between focus parts and tone marks, the possible tone marks can occur on the actor and actee parts based on these sixty two sentences. In the actor part, only two possible cases for tone marks occur at actor part as shown in figure 6.2. The first case is no tone mark at the actor part. The second case is that tone mark `H*` or its repetition is at actor part. For example, in line 4, the tone mark at the actor part is none, whereas, in line 5 and 6, tone marks at actor part are `H*` and `H* H*`, respectively.

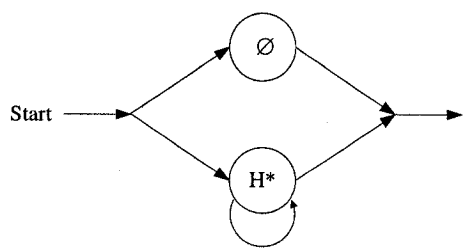


Figure 6.2: Tone marks at the actor part in the CMU-COM dataset

From the sixty two sentences, several tone marks occur at the actee part. These

marks are classified into four groups of accent tones: ∅, H*, L+H*, and !H* and four groups of boundary tones: ∅, H-H%, L-H%, and L-L%. The combinations can occur between these groups on the actee part such as H* L-L% and L+H* L-L%. (See figure 6.3) Only H* and L+H* can repeat before combining to a boundary tone such as H* H* L-L%.

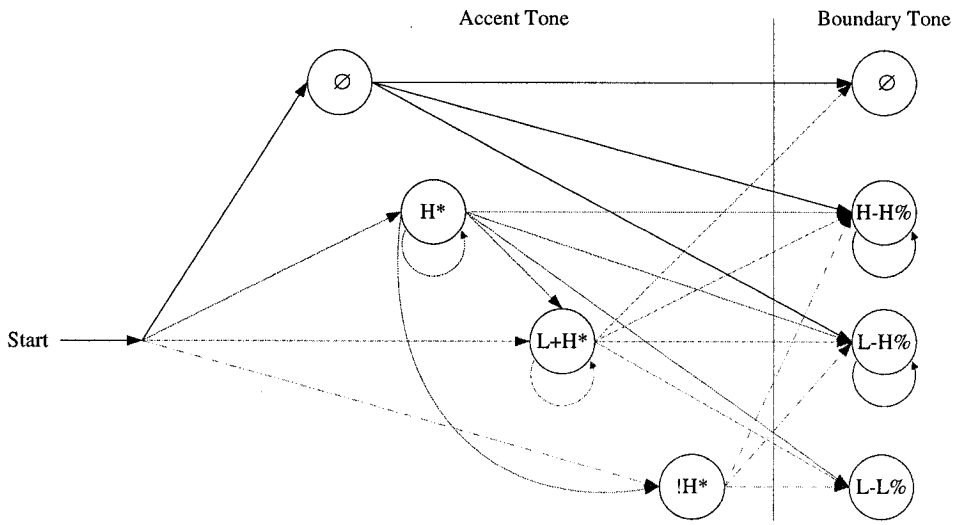


Figure 6.3: Tone marks at the actee part in the CMU-COM dataset

6.2.2 Tone Patterns in Manually Annotated Dataset

By observing the automatic annotation dataset, we see that some possible tone marks do not appear in this dataset. A reason that some tone marks are not marked is the limitation of the automatic ToBI marking system, which is insensitive to retrieve some tone patterns by using fundamental frequency pattern recognition. For example, the annotation system avoids marking low tone and covers only the tones emphasizing the utterance such as high tone or peak accent. Therefore a manually annotated dataset is investigated by employing the dataset from the OSULL [72] which contains 61 sentences with marked tones by linguists. The speech act codes are assigned to each sentence for this datasets. The same speech act codes are grouped together and analyzed them at tone marks of the actor, act and actee parts with the sentence types as shown in table 6.2.

Comparing between the CMU-COM dataset and the OSULL dataset, the OSULL marked by linguists is more sensitive to mark tones than the CMU-COM dataset using

Table 6.2: Speech act codes, verbs, and tones in the OSULL's dataset

Line No.	Speech Act Code	Verb	Tone Marks (Act Part)	Tone Marks (Actor Part)	Tone Marks (Actee Part)	Sentence Types
1	EN0ab	have	-	-	L+H* L- L+H* L-L%	s
2	EN0ab	need	-	-	L* H-, L* H-, L+H*, H* H-H%	s
3	EN0ab	need (loan)	-	H*	H* L-L%	s
4	EN0ab	need (loan)	-	L*	H* L-L%	s
5	EN0ab	need (loan)	-	%H L*	H* L-L%	s
6	EN0ab	have (marmalade)	-	-	L* H-, L* H-H%	y-n-q
7	EN0ab	have (marmalade)	-	L*	L*	y-n-q
8	EN0ab	have (marmalade)	-	-	L* H-, L* H-H%	y-n-q

the automatic tone marking system. The linguists can also mark a longer variety of tone marks than the automatic system. For example, the accent tone marks L* and L*+H, and the boundary tone marks L- and H- never occur in the CMU-COM dataset. Following the dataset from the OSULL, the possible tone marks, occurring at the actor part, are shown in figure 6.4. Considering this dataset, if boundary tone mark appears at the actor part then the strong tone emphasis must be at the actor part. Consequently, the focus need to be at the actor part. The possible tone marks are more complicated than the possible tone marks from the CMU-COM dataset as shown in figure 6.2. Therefore, the tone marks diagram at the actor part in OSULL's dataset can cover the tone marks diagram at actor part in the CMU-COM dataset The structure in figure 6.2 is the subset of the structure in figure 6.4. The diagram in figure 6.4 shows that at least ten different tone marks (including their combinations but excluding the repeatable tone marks) can occur at the actor part. They are H*, H* L-L%, H* H-H%, H* L-, H* H-, !H*, L*, L* H-, L*+H, and ⊘.

For the actee part, the linguists also mark more various tone marks than the automatic tone marking system. Thirty seven different tone marks occur at the actee part in the OSULL's dataset including single tones and their combinations. The possible tone marks, occurring at the actee part, are depicted in figure 6.5. The linguists can mark tone marks better than the automatic tone marking system. They can select more accurate tone marks than the automatic system. The combinations of tone marks at the actee part are complicated than at the actor part. Some tone marks at actee part in OSULL's dataset, such as L*, L*+H, L*+!H*, L-,and H- never appear in the CMU-COM dataset. The diagram in figure 6.5 show that at

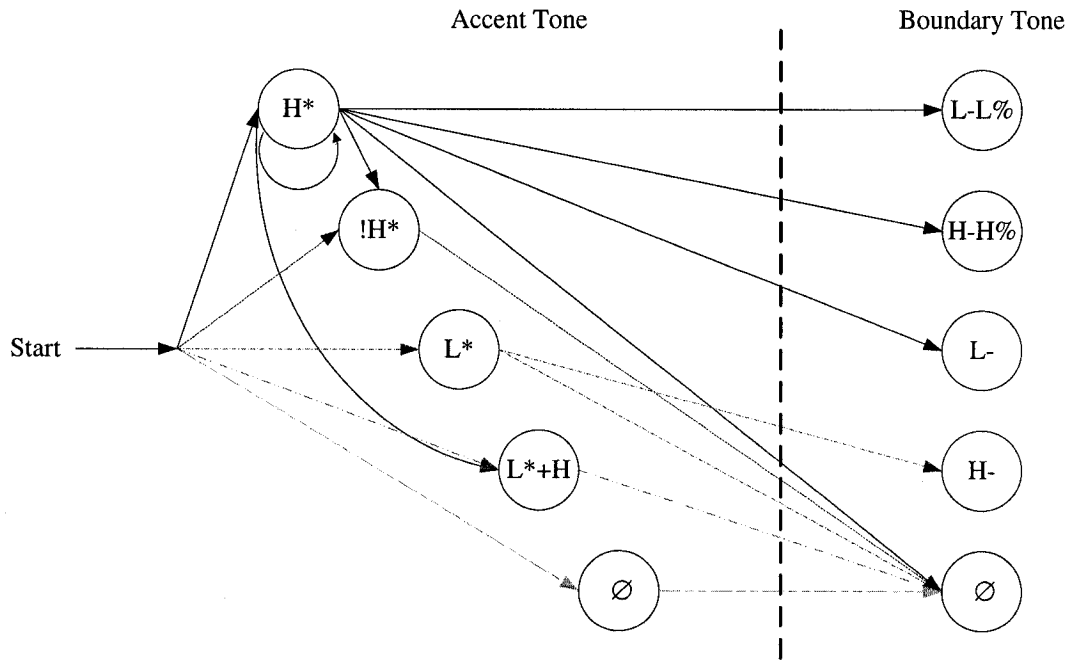


Figure 6.4: Tone marks at actor part in the OSULL's dataset

least twenty seven different tone marks (include their combinations but excluding the repeatable tone marks) can occur at the actor part. They are H-H%, L-H%, L-L%, L-, H-, H* H-H%, H* L-H%, H* L-L%, H* L-, H* H-, L+H* H-H%, L+H* L-H%, L+H* L-L%, L+H* L-, !H* H-H%, !H* L-H%, !H* L-L%, !H* L-, L*+H L-H%, L*+H L-L%, L* H-H%, L* L-L%, L* H-, L* L-H% and ∅. The tone mark diagram in figure 6.5 can cover the tone mark diagram in figure 6.3 which represents the tone marks at actee part of the CMU-COM dataset.

Some phenomena of marking tones are observed in both datasets. For instance, the boundary tones (H- and L-) occur in front of comma, semicolon, and conjunction (and, or, but, and so on) while the boundary tones (H% and L%) occur at the last word before full stop. However, these boundary tones can be placed at the words that the speaker wants the listener to recognize the content by emphasizing the strong utterance. Most boundary tones are used to compare the different components in a sentence and connect between phrases or sentences, such as "if ... then ...". The high accent tones, such as H* and L+H*, is used to emphasize the content at that word, or speaker agrees or wants to support the previous content. On the other hand, the low tones, such as L* and L*+H, are marked when speaker does not want to emphasize

the content. Sometime the low tones are used when speaker disagrees or argues with hearer.

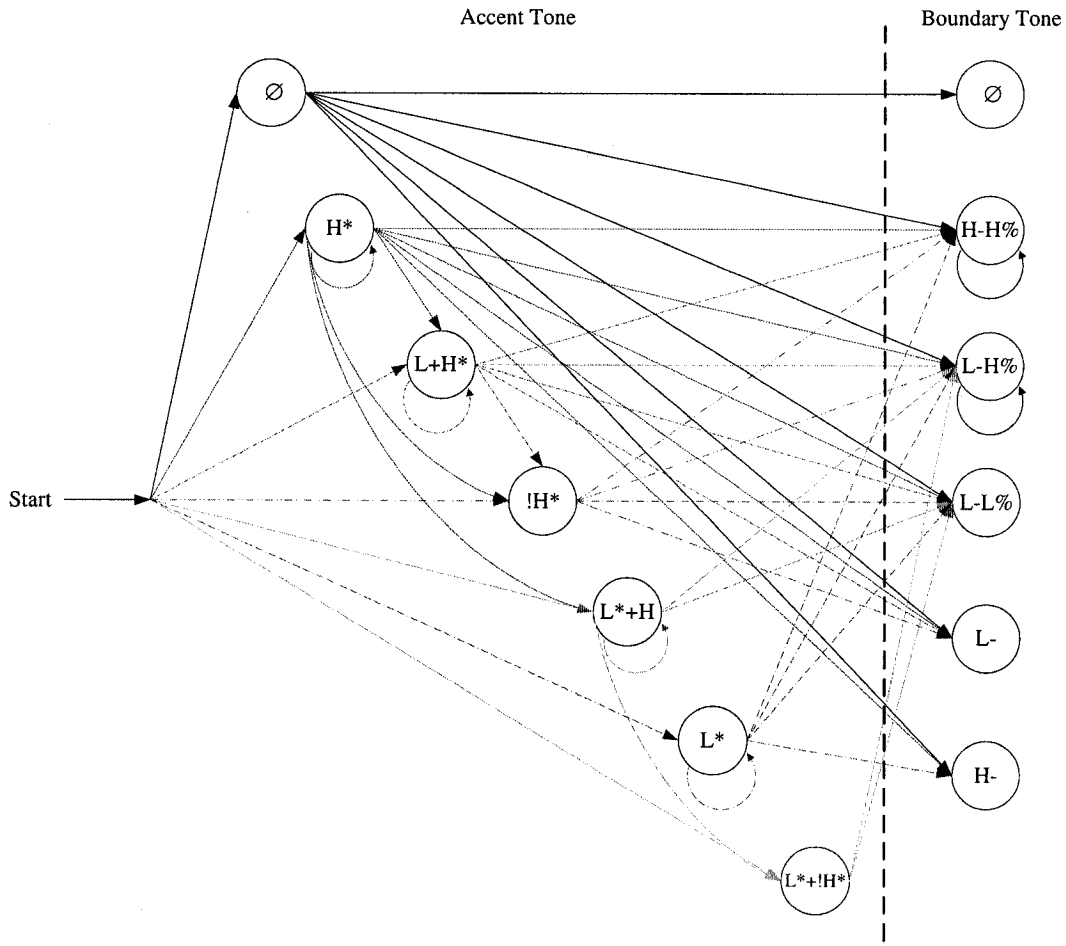


Figure 6.5: Tone marks at actee part in the OSULL's dataset

6.2.3 Tone Patterns Based on Intonational Theory

In the intonational structure, the tone phenomena are considered to help designing the tone patterns for the FET system. In this section, the ToBI annotation system is used to describe the tone patterns and their phenomena based on [74]. The ToBI representation is comprised from the low tone (L), high tone (H), and low-high tone combination (LH) called bitonal tone. ToBI is composed of two type of tones: accent and boundary tones. Accent tone is marked at any accented syllable and accent tones for ToBI include H* (peak accent), L* (low accent), L*+H (scooped accent), L+H*

(rising peak accent). The bitonal pitch accents H*+L and H+L* are eliminated from the ToBI system because these pitch accents, which are downstep pitch accents, are transformed to be the combinations of L or H with the pitch accent !H* (downstep accents) represented as H+!H* and L+!H*. The downstep pitch accent is described in the next paragraph. The boundary tone or phrasal tone is assigned at any intermediate or intonation phrase. The representations of boundary tones are L-, H-, L-L%, H-H%, L-H%, and H-L%. The extra phrasal tone is %H which can be marked at initial boundary or the beginning of the intonational phrase.

The tone phenomena are used to explain the intonation patterns or pitch contours. In general, utterance trend declines in fundamental frequency and this frequency declination can be represented by the tone phenomenon, called *DOWNTRENDS* [74]. For this phenomenon, the pitch contour declines to the low level of the pitch range. One of the *DOWNTRENDS* phenomena is the downstep pitch accents, such as !H*, and L+!H*. Describing in [37], some examples of downstep pitches are shown below:

- A downstepped H* is given as !H* for the succession of downstepped peaks. This is the repetition of rising peak pitch accents. For instance L+H* L+H* L-L% is re-defined to L+H* L+!H* L-L% in ToBI

- A high-level pitch is followed by downstep pitch accent declared by H+!H*, which is called "staircase". For instance, H* H+L* L-L% is defined as H* H+!H* L-L% in ToBI.

- H*+L is a downstep trigger, which is called "calling contour" For instance, H*+L H* L-L% is H* !H* L-L% in ToBI.

Usually, the downstep pitch accents appear after the peak accents or the stresses. They are frequently marked behind the focus part or after the peak accents to increase the tone emphasis at the focus part. The downstep pitch accents can be deaccentted to reduce the tone emphasis of the focus part. For the intonational phrase, the downstep boundary tones L-L% and H-L% appear in most affirmative and declarative sentences.

6.3 Relationships of Speech Acts and Tone Marks for Each Focus Part

Tone marks for each focus part are investigated to retrieve a set of tone patterns of each focus part. Since the focus parts only are not enough to analyze the tone patterns then the focus part with the sentence types and the positions of words or a list of words are served to determine the tone patterns. These patterns are recognized following the automatic tone annotation dataset from the CMU-COM dataset and the manual dataset from OSULL. Section 6.3.1 expresses tone marks which occur at each focus part and the relations of tone marks with the positions of words and sentence types. The relationships of tone patterns with different speech act categories depending on the focus parts, sentence types, and the positions of words are described in the section 6.3.2

6.3.1 Tone Marks Depending on Focus Parts

In determining the relationships between speech acts and focus parts, some common patterns are recognized to mark tones in a sentence. For example, the tone mark `L-L%`, analyzed as low phrase tone (`L-`) to low boundary tone (`L%`), is marked at the last word of a sentence for most affirmative sentences. The tone marks `H-` (high phrase tone) and `L-` are marked at the last word before conjunction (such as "and", "or", "but", and so on) or are marked at the last word of the current phrase (following the next phrase). The tone mark `H*` (high accent tone) is used to emphasize content at a word or a group of words in a sentence. If we want strong emphasis at a word or a group of words then the tone mark `L+H*` (rising accent tone) is selected for strong emphasis instead of `H*`. The tone mark conditions are concluded in table 6.3.

Table 6.3: Positions and tone marks on a sentence

Conditions	Tone Marks	Position
Affirmative sentence	L-L%	at the end
Interrogative sentence	H-H%	at the end
Conjunction (and, or, but, etc)	H- / L-	before conjunction
Phrase1, Phrase2, ...	H- / L-	before conjunction
Emphasis / Strong emphasis	H* / L+H*	any words

We consider each category and its subcategories of speech acts for each dataset. The relationships of speech acts and tone marks are investigated in the OSULL's

dataset and the CMU-COM dataset. To compare between two datasets, the variations of actions in a sentence in the manual dataset is lower than the automatic dataset. For the OSULL's dataset, most actions are found in three speech act categories: thinking (*DE*), enaction model (*EN*) and competition and corporation model (*KA*), while five speech act categories are discourse model (*DI*), *EN*, experience and text model (*ET*), *KA* and thematic phrases model (*TV*) for the CMU-COM dataset. In this section, the main subgroups of speech act codes are: intending (*EN0ab*), want (*DE8b*), victory (*KA4a*), and thematic phases model (*TV*), which have enough information to explore the tone patterns. The actor part is marked only with the accent tone or no tone marks. However, sometime the boundary tones `L-` or `H-` can be marked following the accent tones at the actor part if the speaker wants hearer to focus at the actor part. The actee part can be marked by the boundary tones, such as `L-L%` and `H-H%`, or combinations of accent tones and boundary tones. The act part is marked by the accent tones or nothing. If a sentence does not have an actee part, then the act part can be marked by boundary tone (`L-L%` or `H-H%`), the combinations of accent and boundary tone (`L-L%` or `H-H%`), or no tone marks.

6.3.2 Tone Patterns with the Different Speech Acts

For example, in the speech act code *EN0ab* ("intending"), if the speaker focuses on what the actor wants to do, the actee part is the most important part in the sentence. The components of accent tone (*Accent–Tone*) and boundary tone (*Bound–Tone*) in the actee part can be repeated, as shown in the tone pattern (6.1).

$$Actee\_tone \leftarrow ([Accent - Tone] + Bound - Tone)^n \tag{6.1}$$

Note: n is the number of phrases, and the variable in square bracket is optional

The tone patterns are a combination of accent and boundary tones. The accent tone labels for the actee parts are `L*` or `L+H*` for affirmative sentences, and `H*` or `L+H*` for interrogative sentences. For the boundary tone, the possible tone marks are `L-` or `L-L%` for affirmative sentences and `H-` or `H-H%` for interrogative sentences. The tone patterns of the actee part for *EN0ab* are shown in tone patterns (6.2) and (6.3). From the datasets, the tone patterns are found for affirmative and interrogative sentences.

The tone patterns for these sentences are repeatable until ending with the tone marks L-L% or H-H%.

For the affirmative sentence

$$Actee\_tone \leftarrow (\texttt{L*} \vee (\texttt{L+H*}) + \texttt{L-})^{n-1} + ([\texttt{L*} \vee (\texttt{L+H*})] + \texttt{L-L\%}) \tag{6.2}$$

For the interrogative sentence

$$Actee\_tone \leftarrow (\texttt{H*} \vee (\texttt{L+H*}) + \texttt{H-})^{n-1} + ([\texttt{H*} \vee (\texttt{L+H*})] + \texttt{H-H\%}) \tag{6.3}$$

Note: n is the number of phrases and the variables in square bracket are optional

For the speech act code *DE8b* ("want"), the speaker informs the listener that the speaker wants something from listener. Therefore, the actee part is emphasized by tone marks. The relevant tone marks are the accent tone H* and the boundary tones L-L% and H-H%, both occurring at the actee part in the datasets. The tone patterns are shown in tone patterns (6.4) and (6.5) and they use simple tone patterns for the affirmative and interrogative sentences.

For the affirmation sentence

$$Actee\_tone \leftarrow (\texttt{H*}) + (\texttt{L-L\%}) \tag{6.4}$$

For the interrogative sentence

$$Actee\_tone \leftarrow (\texttt{H*}) + (\texttt{H-H\%}) \tag{6.5}$$

For the speech act code *KA4a* ("victory"), the focus must be at both the actor and actee parts. The relevant tone marks are the accent tone marks H* and L+H* and the boundary tone marks L-L% and H-H%. The tone patterns are shown in tone pattern (6.6) for actor part, and the patterns (6.7) and (6.8) for the actee part of affirmative and interrogative sentences.

Actor Part

For affirmative and interrogative sentences

$$Actor\_tone \leftarrow (\texttt{H*}) \vee (\texttt{L+H*}) \tag{6.6}$$

Actee Part

For the affirmation sentence

$$Actee\_tone \leftarrow ([\texttt{H*} \vee (\texttt{L+H*})] + \texttt{L-})^{n-1} + (\texttt{L-L\%}) \tag{6.7}$$

For the interrogative sentence

$$Actee\_tone \leftarrow ([\texttt{H*} \vee (\texttt{L+H*})] + \texttt{H-})^{n-1} + (\texttt{H-H\%}) \tag{6.8}$$

Summary of the relationships of speech acts and tone marks grouping by focus parts are shown in table 6.4. Since the example sentence has focus at actee part, speech act code is *EN0ab*, and the sentence mood is affirmative sentence (*aff*), the tone marks are defined for a set of words in the actee part as `L+H* L-L%`, following table 6.4.

Table 6.4: Tone constraints

Code	Focus Type	Sentence Type	Condition
En0ab	Actee	Aff	Actee_tone ← $([L^* \vee (L+H^*)]+L-)^{n-1}+([L^* \vee (L+H^*)]+L\text{-}L\%)$
		Int	Actee_tone ← $([H^* \vee (L+H^*)]+H-)^{n-1}+([H^* \vee (L+H^*)]+H\text{-}H\%)$
DE8b	Actee	Aff	Actee_tone ← H*+L-L%
		Int	Actee_tone ← H*+H-H%
KA4a	Actor	Aff	Actor_tone ← $H^* \vee (L+H^*)$
		Int	Actor_tone ← $H^* \vee (L+H^*)$

6.4 Structures for the Relationships of Speech Acts and Prosodic Marks Based on Focus Parts

The FET feature structure constrains the relationships of focus parts with the speaker's intentions and prosodic marks. The FET structure is composed of two main structures: the speech act structure (*SPAct*), and the focus information (*Focus-Info*) structure, as shown in figure 6.6.

The *SPCode*, derived from the speech act classification by Ballmer and Brennenstuhl [71], is defined inside the *SPAct* structure (figure 6.7(a)). Mostly, the *SPCode* is inferred from the main action or main verb of a sentence. Each code identifies a category of the speaker's intention. For example, the speech act code *EN2b*, which

$$\begin{bmatrix} SPAct & \begin{bmatrix} SPCode & code \\ STMood & mood \\ FCGroup & group \end{bmatrix} \\ Focus\text{-}Info & \left\{ \begin{bmatrix} Focus\text{-}Part & actor \\ FET\text{-}obj & \langle\rangle \\ Prosody & \langle\rangle \end{bmatrix}, \begin{bmatrix} Focus\text{-}Part & actee \\ FET\text{-}obj & \langle\rangle \\ Prosody & \langle\rangle \end{bmatrix}, \begin{bmatrix} Focus\text{-}Part & act \\ FET\text{-}obj & \langle\rangle \\ Prosody & \langle\rangle \end{bmatrix} \right\} \end{bmatrix}$$

Figure 6.6: *Focus-Info* inside the FET structure

means "asking for", is assigned to these verbs such as beg, request, require, invite. Another feature in the *SPAct* structure, the sentence moods (*STMood*) is obtained from *E.Mood* feature in the MRS representation and represents the type of sentence such as affirmative sentence. The last feature in the *SPAct* structure is the focus criteria (*FCGroup*). This feature is derived from the table 5.3. In the example of *SPAct* structure depicted in figure 6.7(b), that *SPCode* is *EN0ab* ("intending"), *STMood* is interrogative sentence, and *FCGroup* is "G", which indicates that the focus of sentence at actee part can be wide focus (*w-focus*) or single focus (*s-focus*).

$$SPAct: \begin{bmatrix} SPCode & code \\ STMood & mood \\ FCGroup & group \end{bmatrix} \quad (a) \qquad SPAct: \begin{bmatrix} SPCode & EN0ab \\ STMood & interrogative \\ FCGroup & G \end{bmatrix} \quad (b)$$

Figure 6.7: The speech act feature structure: (a) *SPAct* structure and (b) an example of *SPAct* structure

The *Focus-Info* structure indicates what focus part (actor, act or actee part) must be emphasized by tone, and how the prosodic information can be related to focus information. The *Focus-Info* structure contains the focus parts with their focus features, such as *FCType*, and *Prosody* structure. The *Focus-Info* structure is shown in figure 6.8.

Considering the *Focus-Info* structure for each focus part, the *FCType* can be assigned focus: the wide focus (*w-focus*), the single focus (*s-focus*), or no focus. The *List-obj* contains words or lists of words from a sentence, while the *Relation* represents what part in a sentence is the focus part (actor, act, or actee part). The *Index* is used to indicate its structure and *RIndex* indicates the related *Focus-Info* structure of the other focus parts. The *FET-obj* contains the groups of words which must be

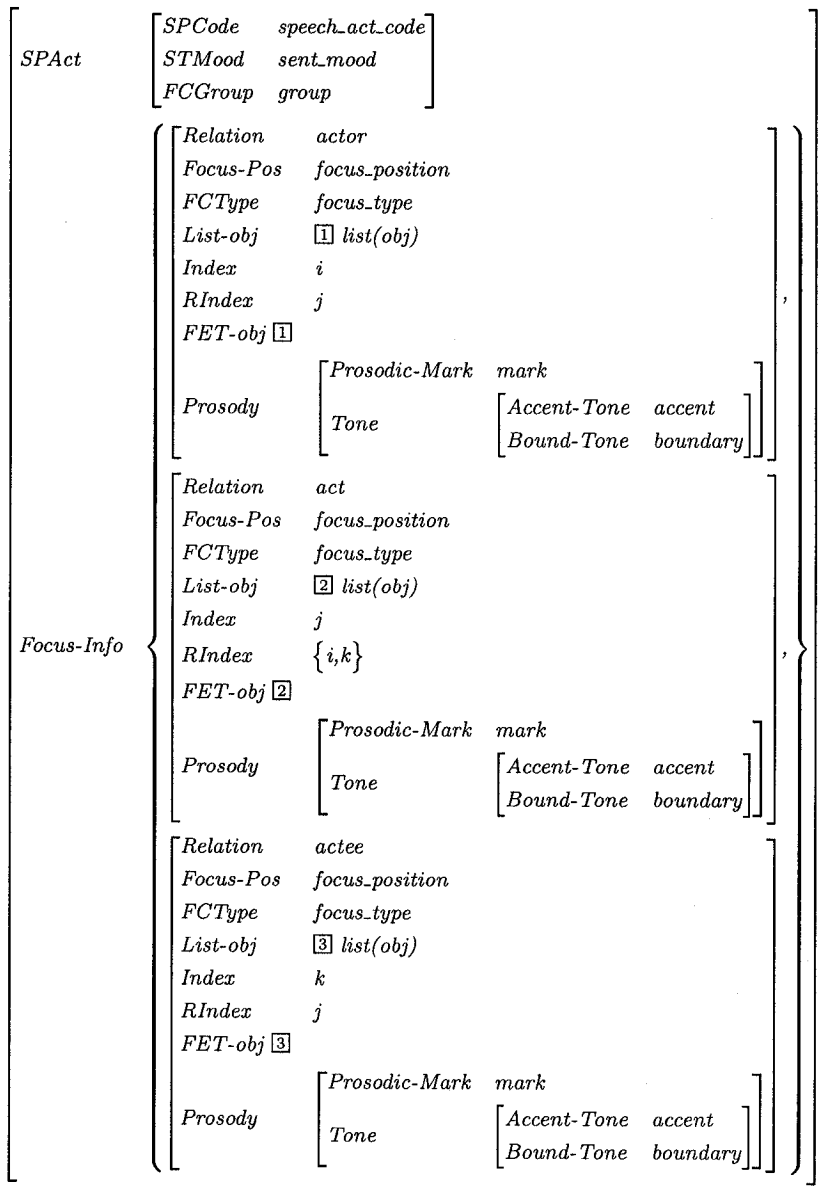


Figure 6.8: Focus information structure

emphasized by tone. If the focus part is no focus then the *FET-obj* can be the same as *List-obj*.

The *Prosody* structure includes the prosodic information and refers the *FET-obj* from the *Focus-info* structure. The *Focus-Info* structure assigns which positions in a sentence must be labelled by tone marks while the *Prosody* structure declares what tone marks can be labelled at the focus position. The *Prosodic-Mark* structure shows a type of prosodic marks while the *Tone* structure represents ToBI marks as a set of accent and boundary tone marks. These tone marks are labeled for each *FET-obj*.

Mapping between prosodic marks and a set of accent-boundary tone marks is described in section 5.4.2 and is shown in table 5.4. The *Prosody* structure is designed to map between *Prosodic-Mark* and ToBI mark in *Tone* structures, as illustrated in figure 6.9(a). For the *Focus-Info* structure, the *FET-obj* feature contains the list of words that must be emphasized by tone. Since, the *FET-obj* links to *Prosodic-Mark* and *Tone* structure, then the prosodic marks are selected by analyzing the relationships between tone marks and speech acts code following section 6.3.2. The example of this mapping is shown in figure 6.9(b). In this example, the *FET-obj* is <a,book> which is marked by the prosodic mark *Em_Lg-break*. The *Em_Lg-break* represents the emphasized accent tone with long break of boundary tone and is mapped to accent tone H* and boundary tone L-L%. In the figure 6.10, the *Prosodic-Mapping* structure represents information in table 5.4.

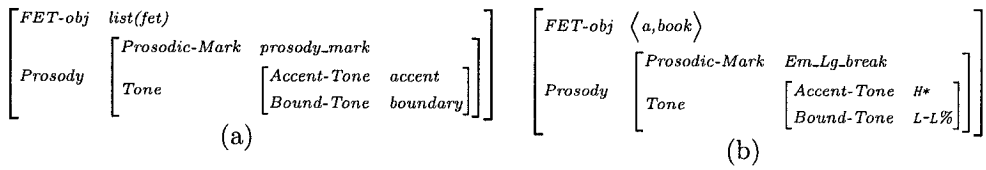


Figure 6.9: Mapping feature structure (a) the structure for mapping between prosodic information and accent-boundary tone marks, and (b) an example of this mapping

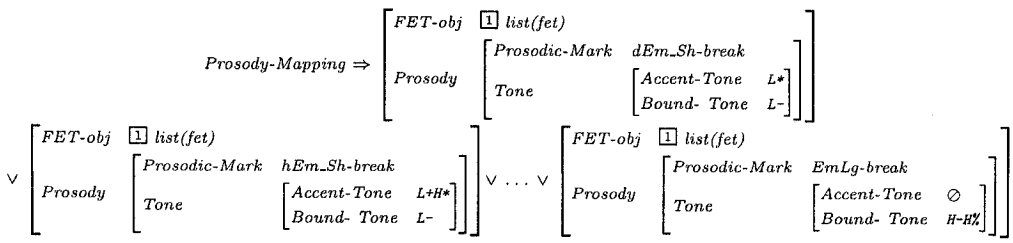


Figure 6.10: Prosodic mapping structure

The *FET* structure is designed to represent the relations of focus, prosodic and tone domains. The focus parts have actor, act, and actee parts. Sometimes each part can have more than a group of words as shown in figure 6.11. The groups of words are represented by $a_1, a_2, \ldots, a_n, b_1, b_2, \ldots, b_m, c_1, c_2, \ldots, c_o$. At each part, the *FET-obj*, which is a list of groups of words, is referred to the *Prosody* structure. Following the prosodic structure in figure 6.10, only a group of words from prosodic information can be mapped to tone information. Therefore a condition is defined to split the list of

groups of words into an individual list of prosodic structures such that each structure contains only a group of words illustrated in figure 6.11. This condition is shown in figure 6.12(a). The *FET-obj* contains a multiple list of words, which require splitting the list of words. As a result of splitting, the *Focus-Info* structure includes the list of prosodic structure as illustrated in figure 6.12(b).

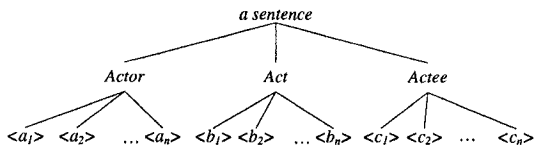


Figure 6.11: Several groups of words of actor, act and actee parts

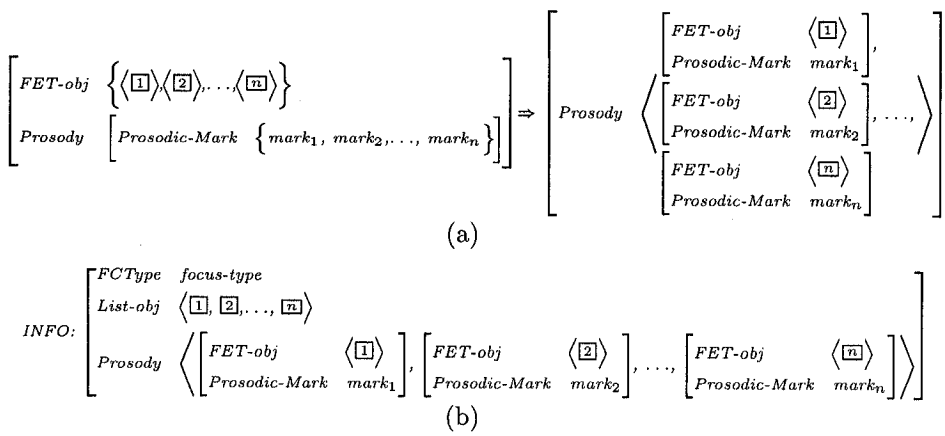


Figure 6.12: The conditional structure: (a) condition to split the list of *FET-obj*, and (b) the complete structure after split the list

An example of the information structure of the sentence "Mary bought a book" is shown in the figure 6.13. The prosodic marks can be referred to tone marks following the table 5.4. The tone marks `H* L-L%` are defined at the actee part "a book" as a result.

6.5 Summary

We investigated the prosodic patterns from manually annotated dataset, automatically annotated dataset, and intonation theory. We explored the prosodic phenomena in the human utterances. In this chapter, three speech act codes, *EN0ab* (intending), *DE8b* (want), and *KA4a* (victory), are selected for our studies of the relationships

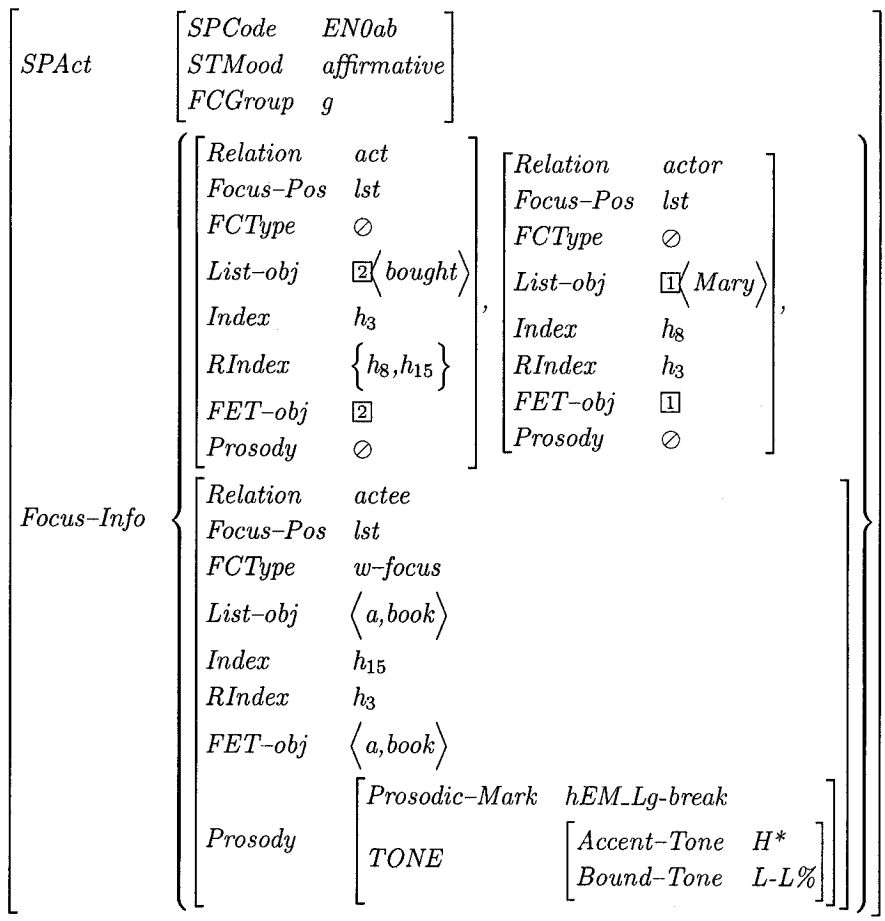


Figure 6.13: Information structure of "Mary bought a book"

of speech acts and tones for each focus part. These relationships are described in section 6.3. The structures for the relationships are designed inside the FET structure. These structures contain the speaker's intention information, sentence types, and prosodic features. They are explained in section 6.4.

Chapter 7

Focus to Emphasize Tone Structure for the Linguistic Knowledge Building System

The LKB system is an HPSG parser requiring grammar, feature structures, and constraints to parse a sentence. Within the FET system, we design the FET subgrammar, including the type hierarchy, focus structures, prosodic structures, rules, and constraints. This subgrammar is accepted by the LKB system to analyze prosodic features, and the system annotates tone marks, depending on a focus part in a sentence, as a result. In this chapter, we describe how the focus words are presentend by the LKB system with our FET subgrammar. The focus words are represented by the FC structure described in chapter 4 and are designed to be compatible with the LKB system. The focus word structure is explained in section 7.1. The details of the FET subgrammar containing typed hierarchy, its structure, and components are described in section 7.2. Inside the FET subgrammar, three main groups of features for the FET structure in section 7.3 are **focus–value**, **prosody–value**, and *feat–struc*. **focus–value** consists of a set of focus features such as focus type, focus group, and focus part. Description of the focus features is given in section 7.3.1. **prosody–value** contains the prosodic features and ToBI marks. It is used to constrain the relations of speech acts and tone marks for each focus part, which is described in section 7.3.2. *feat–struc* constrains the focus and prosodic features, based on the tone patterns with the focus parts and it is described in section 7.3.3.

7.1 Transformation of Focus Content Structure to Focus Words in the LKB System

The LKB system with ERG parses a sentence and generates the MRS representation. By scanning each object inside the MRS representation, all reference numbers are mapped to their objects. Every connection which is related to this object and this reference number is recorded. Only necessary information is extracted to generate the FC structure represented as a set of focus words described in chapter 4. These focus words are generated to correspond to the LKB system. For a sentence, a speech

act code is referred to a main action or main verb and a focus group is selected by a speaker from table 5.3.

Based on the FC structure in figure 7.1, each focus word includes a focus part as shown in figure 7.2. A focus word structure (*focus-word*) contains the focus group, speech act code, sentence mood and focus position in a focus part. These features are part of the argument *AGRS*. For example, *AGRS* of focus word "Kim" is labeled by *ls-actor_G-aff-enoab*. The focus group is defined as group G (see table 5.3) and the speech acts code is *En0ab* (intending). The sentence mood referring from MRS is the affirmative sentence (*aff*) and focus position is the last node (*ls*). Each focus word includes the feature constraints of HEAD, SPR and COMPS, which are described in section 7.3.3. The feature structure ORTH is used to declare the orthography of the focus word. In figure 7.2, "Kim" is an actor part, while "bought" is an act part. The words "a" and "flower" are the actee parts.

$$\left\{ \begin{bmatrix} relation & actor \\ list\text{-}obj & \langle Kim \rangle \\ index & h_8 \\ rindex & h_3 \end{bmatrix}, \begin{bmatrix} relation & actee \\ list\text{-}obj & \langle a, flower \rangle \\ index & h_{15} \\ rindex & h_3 \end{bmatrix}, \begin{bmatrix} relation & act \\ list\text{-}obj & \langle bought \rangle \\ index & h_3 \\ rindex & \{h_8, h_{15}\} \end{bmatrix} \right\}$$

Figure 7.1: Focus content structure of "Kim bought a flower"

7.2 Focus to Emphasize Tone Subgrammar

In FET system, a set of focus words is provided to the LKB system with the FET subgrammar. This subgrammar contains the constraints, rules, type hierarchy, a set of features, and their structures for the FET analysis. The type hierarchy allows for inheritance of constraints. The FET type hierarchy is shown in figure 7.3. Three main groups of features are: **focus–value**, **prosodic–value** and *feat–struc* to control the focus constraints. **focus–value** represents the focus structures. It is composed of five features: focus criterion (*fcgroup*), focus type (*fctype*), focus part (*focus*), focus position (*focus–pos*), and checking whether a tone mark can be marked at a word (*tone–mark*). **prosody–value** is a group of prosodic features. Four prosodic

```
Kim := focus-word &
  ORTH    "Kim",
  HEAD    actor-part & [ AGRS ls-actor_G-aff-en0ab ],
  SPR     <  >,
  COMPS   <  > ].

bought := focus-word &
  ORTH    "bought",
  HEAD    act-part & [ AGRS ls-act_G-aff-en0ab ],
  SPR     < [ HEAD actor-part & [ AGRS ls-actor_G-aff-en0ab ] ] >,
  COMPS   < focus-phrase & [ HEAD actee-part &
          [ AGRS ls-actee_G-aff-en0ab ] ] > ].

a := focus-word &
  ORTH    "a",
  HEAD    actee-part & [ AGRS pv-actee_G-aff-en0ab ],
  SPR     <  >,
  COMPS   <  > ].

flower := focus-word &
  ORTH    "flower",
  HEAD    actee-part & [ AGRS ls-actee_G-aff-en0ab ],
  SPR     < [ HEAD actee-part & [ AGRS pv-actee_G-aff-en0ab ] ] >,
  COMPS   < focus-phrase & [ HEAD actee-part &
          [ AGRS ls-actee_G-aff-en0ab ] ] > ].
```

Figure 7.2: Focus words

features are sentence mood (*stmood*), speech act code (*spcode*), accent tone (*accent–tone*), and boundary tone (*bound–tone*). *feat–struc* contains the core FET structure that constrains the relationships between focus and prosodic features. The *feat–struc* consists of six feature structures: *prosody–mark*, *prosody*, *focus–struc*, *focus–part*, *focus–cat* and *addtone* (checking whether a tone mark can be marked at a focus part) The details of these features are described in the next section.

Following the LKB system conventions, the typed feature structures connecting objects consists of the top type (**top**) which is at the top of typed hierarchy, the list type (**list**), the non–empty list (**ne–list**), and the empty list (**null**). These types and its constraints are declared in the structures (7.1), (7.2), and (7.3). **ne–list** is a daughter of **list** which is a daughter of **top**. The constraint on type

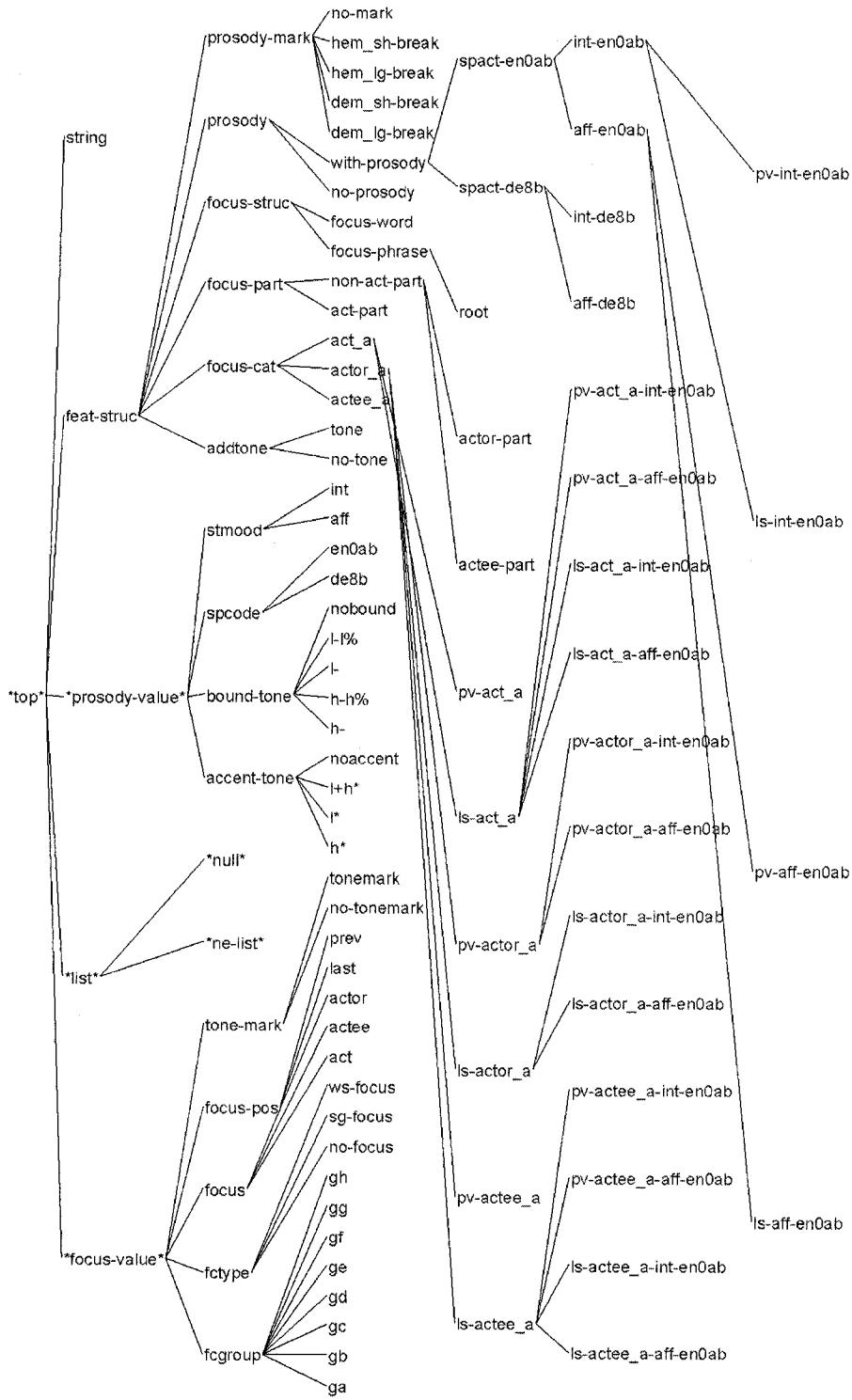


Figure 7.3: FET type hierarchy

ne-list has two features *FIRST* and *REST*. It can be represented by the graph in figure 7.4. The arcs are labeled as *FIRST* and *REST* while the destination points are the structures of *ne-list*, *list*, and *top*. The value of *FIRST* is *top* which can be unified with any feature structure. The value of *REST* is *list* and it can be unified with one of its subtyped feature structures.

$$*list* := *top* \tag{7.1}$$

$$*null* := *list* \tag{7.2}$$

$$*ne\text{-}list* := *list* \;\&\; \begin{bmatrix} \text{FIRST} & *top* \\ \text{LAST} & *list* \end{bmatrix} \tag{7.3}$$

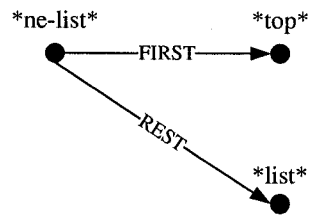


Figure 7.4: Graph of *ne-list* typed feature structure

7.3 Focus to Emphasize Tone Feature Structure

The basis of the FET structure is derived from the descriptions given in chapter 4, 5 and 6. As mentioned in the previous section, design of the FET structure is separated into three main groups: *focus-value*, *prosodic-value*, and *feat-struc*. *focus-value*, is explained in the section 7.3.1. The subfeatures of *focus-value* declare variables of focus features following the knowledge in chapter 5. *prosody-value* includes four features which are described in section 7.3.2. The subfeatures of *prosody-value* represent the variables of prosodic features as reported in chapter 6. *feat-struc* constrains a set of features between focus and prosodic features. These constraints map the focus information to annotate tone marks. The details of *feat-struc* are given in section 7.3.3.

7.3.1 Focus Structure

focus-value contains the focus information and is classified into five features: focus group (*fcgroup*), focus type (*fctype*), focus part (focus), focus position (*focus-pos*), and tone mark (*tone-mark*).

- *fcgroup* is represented by table 5.3.
- *fctype* can be wide focus (*w-focus*), single focus (*s-focus*), or no focus (*no-focus*)
- *focus* is the focus part (actor, act, or actee).
- *focus-pos* defines the position of focus word whether the focus word is at the end of a list of words. *focus-pos* is represented by *last* at the end of a list of words for each focus part otherwise the *focus-pos* is represented by *prev.*
- *tone-mark* is used for checking whether tone can be marked on a focus word and is represented by *tonemark* or *no-tonemark.*

7.3.2 Prosodic Structure

prosodic-value includes four prosodic features: sentence mood (stmood), speech act code (*spcode*), accent tone (*accent-tone*), and boundary tone (*bound-tone*).

- *stmood* represents a sentence type such as affirmative sentence (*aff*), or interrogative sentence (*int*).
- *spcode* represents the speaker's intention of a sentence by a code. These codes are adopted from the Baller [71]. For example, *EN0ab* is "intending" and *DE8b* is "want".
- *accent-tone* represents accent tone for ToBI marks such as `H*` and `L*`
- *bound-tone* represents the boundary tone of ToBI marks such as `H-` and L%.

7.3.3 Constraints of the FET Stucture

The focus structure (*feat–struc*) is a group of constraints to control the relations between focus and prosodic features to generate tone marks. *feat–struc* is composed of six main feature structures: *focus–struc, focus–part, focus–cat, prosody,, prosody–mark* and *addtone*. *focus–struc*, as shown in structure (7.4), consists of HEAD, specifier (SPR) and complement (COMPS) [58]. *focus–struc* is assigned to be a subfeature of focus word (*focus–word*) and focus phrase (*focus-phrase*) structures. Inside the *focus–struc*, HEAD refers to *focus–part* which is shown in structure (7.5). SPR and COMP are used to constrain the components of previous nodes and following nodes in a sentence respectively. Each *focus–part* contains focus and prosodic structures.

$$\textit{focus-struc} := \textit{feat-struc} \;\&\; \begin{bmatrix} \text{HEAD} & \textit{focus-part} \\ \text{SPR} & *\textit{list}* \\ \text{COMPS} & *\textit{list}* \end{bmatrix} \tag{7.4}$$

focus–part structure classifies between *act* and *non–act* part structures. *non–act* part structure is separated into actor or actee part structure as shown in structures (7.8) and (7.9).

$$\textit{focus-part} := \textit{feat-struc} \;\&\; \begin{bmatrix} \text{FOCUS} & \textit{focus} \\ \text{ARG1} & \textit{focus-cat} \end{bmatrix} \tag{7.5}$$

$$\textit{act-part} := \textit{focus-part} \;\&\; \begin{bmatrix} \text{FOCUS} & \textit{act} \end{bmatrix} \tag{7.6}$$

$$\textit{non-act-part} := \textit{focus-part} \tag{7.7}$$

$$\textit{actor-part} := \textit{focus-part} \;\&\; \begin{bmatrix} \text{FOCUS} & \textit{actor} \end{bmatrix} \tag{7.8}$$

$$\textit{actee-part} := \textit{focus-part} \;\&\; \begin{bmatrix} \text{FOCUS} & \textit{actee} \end{bmatrix} \tag{7.9}$$

Inside *focus–part*, the focus category (*focus–cat*) structure is a set of constraints which are the combinations of a focus part and a focus group such as *act_g*, *actor_g*, *actee_g*, and so on. The focus features are classified following the possible *focus–cat* for the FET structure. The *focus–cat* constrains the actor, act and actee parts. The *focus–cat* contains both the focus and prosodic features as a set of subfeatures of the FET structure. This structure consists of focus position (*focus-pos*), focus criterion (*fcgroup*), focus type (*fctype*), adding tone (*addtone*), and prosodic structure (*prosody*) as shown in structure (7.10).

$$\textit{focus-cat} := \textit{feat-struc} \;\&\; \begin{bmatrix} \text{FOCUS-POS} & \textit{focus-pos} \\ \text{FCGROUP} & \textit{fcgroup} \\ \text{FCTYPE} & \textit{fctype} \\ \text{ADDTONE} & \textit{addtone} \\ \text{PROSODY} & \textit{prosody} \end{bmatrix} \tag{7.10}$$

prosody structure consists of these features: sentence mood, speech act code, and a set of prosodic mark structures. It controls the prosodic marks following the FET constraints. These constraints depend on the relationships of focus with speech acts and intonation patterns. *prosody* is shown in structure (7.11). The *prosody–mark* structure represents accent and boundary tones, by using ToBI representation which is illustrated in structure (7.12).

$$\textit{prosody} := \textit{feat-struc} \;\&\; \begin{bmatrix} \text{STMOOD} & \textit{stmood} \\ \text{SPCODE} & \textit{spcode} \\ \text{PROSODY-MARK1} & \textit{prosody-mark} \\ \text{PROSODY-MARK2} & \textit{prosody-mark} \end{bmatrix} \tag{7.11}$$

$$\textit{prosody-mark} := \textit{feat-struc} \;\&\; \begin{bmatrix} \text{ACCENT-TONE} & \textit{accent-tone} \\ \text{BOUND-TONE} & \textit{bound-tone} \end{bmatrix} \tag{7.12}$$

focus–phrase in structure (7.13) inherits the *focus–struc* with the argument ARGS. The ARGS constrains the type *list*. The focus rules parse the *focus-phrase* with their constraints and define whether tone can be marked at a word in each focus part.

$$\textit{focus-phrase} := \textit{focus-struc} \;\&\; \left[\text{ARGS} \quad \textit{*list*}\right] \tag{7.13}$$

focus–word inherits the *focus–struc* with orthography of a word (ORTH) as string. The *focus–word*, as shown in structure (7.14), represents the FC structure and corresponds to the LKB system.

$$\textit{focus-word} := \textit{focus-struc} \;\&\; \left[\text{ORTH} \quad \textit{string}\right] \tag{7.14}$$

7.3.4 Focus to Emphasize Tone Rules

For the FET rules, two types of focus rules are *head-complement* and *head-specifier* rules. These rules are similar to the simple grammar rules which are explained in [58]. These rules are typed feature structures and they are used to connect words, or phrases to make further phrases. These rules contain four main feature structures: HEAD, SPR, COMPS, and ARGS. HEAD represents the phrase structure of mother's phrase which is connected to the daughter's phrase structures, called head-complement (COMPS) and head specifier (SPR). These connections must follow the agreements of phrase structure (ARGS). Three head-complement rules are defined for the FET system. These head-complement rules can cover co-occurences between two phrases that the daughter phrase is connected behind the mother phrase. For example, noun-preposition phrase agreement is the connection between noun and preposition phrases. *head-complement-rule-0* is processed when the mother phrase has no daughter phrase. *head-complement-rule-1* is used when one daughter phrase is connected to the mother phrase while *head-complement-rule-2* is used if there are more than one connection between mother and daughter phrases and these connections must be linked sequentially.

$$\textit{head-complement-rule-0} := \textit{focus-phrase} \;\&\; \begin{bmatrix} \text{HEAD} & \boxed{0} \\ \text{SPR} & \boxed{a} \\ \text{COMPS} & \langle\rangle \\ \text{ARGS} & \left\langle \textit{focus-word} \;\&\; \begin{bmatrix} \text{HEAD} & \boxed{0} \\ \text{SPR} & \boxed{a} \\ \text{COMPS} & \langle\rangle \end{bmatrix} \right\rangle \end{bmatrix} \tag{7.15}$$

$$\textit{head-complement-rule-1} := \textit{focus-phrase} \;\&\; \begin{bmatrix} \text{HEAD} & \boxed{0} \\ \text{SPR} & \boxed{a} \\ \text{COMPS} & \langle\rangle \\ \text{ARGS} & \left\langle \textit{focus-word} \;\&\; \begin{bmatrix} \text{HEAD} & \boxed{0} \\ \text{SPR} & \boxed{a} \\ \text{COMPS} & \langle \boxed{1} \rangle \end{bmatrix}, \boxed{1} \right\rangle \end{bmatrix} \tag{7.16}$$

$$\textit{head-complement-rule-2} := \textit{focus-phrase} \;\&\; \begin{bmatrix} \text{HEAD} & \boxed{0} \\ \text{SPR} & \boxed{a} \\ \text{COMPS} & \langle\rangle \\ \text{ARGS} & \left\langle \textit{focus-word} \;\&\; \begin{bmatrix} \text{HEAD} & \boxed{0} \\ \text{SPR} & \boxed{a} \\ \text{COMPS} & \langle \boxed{1}, \boxed{2} \rangle \end{bmatrix}, \boxed{1}, \boxed{2} \right\rangle \end{bmatrix} \tag{7.17}$$

head-specifier-rule is processed in the same way as determiner-noun agreement in syntactic analysis. An example is the connection of the noun phrase of "A flower", which links a determiner and a noun. In the FET system, the *head-specifier-rule* is processed when the current focus word is connected to the previous focus word and this connection must agree with the ARGS constraint.

$$\textit{head-specifier-rule} := \textit{focus-phrase} \;\&\; \begin{bmatrix} \text{HEAD} & \boxed{0} \\ \text{SPR} & \langle\rangle \\ \text{COMPS} & \boxed{a} \\ \text{ARGS} & \left\langle \textit{focus-phrase} \;\&\; \boxed{1} \;\&\; \left[\text{SPR}\ \langle\rangle\right],\ \textit{focus-phrase} \;\&\; \begin{bmatrix} \text{HEAD} & \boxed{0} \\ \text{SPR} & \boxed{1} \\ \text{COMPS} & \langle \boxed{a} \rangle \end{bmatrix} \right\rangle \end{bmatrix} \tag{7.18}$$

Using these rules, the example sentence "Mary bought a flower" is parsed and the result is parsing tree as shown in figure 7.5 and the complete FET structure including the focus and prosodic information is shown in appendix A.1.

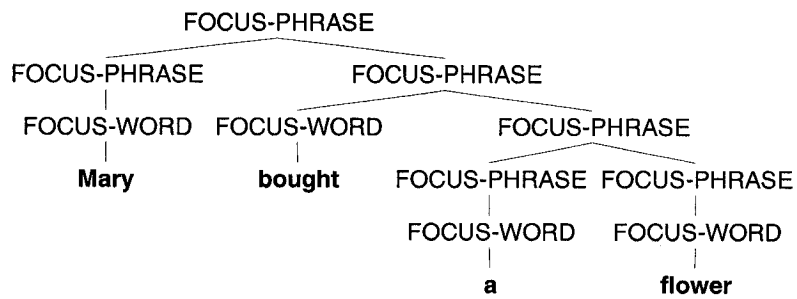


Figure 7.5: FET structure of the word "Mary"

7.4 Summary

We designed the FET subgrammar for our FET anlysis. The subgrammar must be compatible with the LKB system. The LKB system with the FET subgrammar can parse a sentence and its result is the complete FET structure annotated by ToBI marks. The FET subgrammar consists of type hierarchy, focus words, rules, focus constraints, and focus and prosodic typed feature structures as described in section 7.2. The details of FET structure for the LKB system are described in section 7.3.

Chapter 8

Implementation of the FET System

The implementation of the FET system is described in sequential steps by presenting the details of each stage. Briefly, the system is separated into three parts: preprocessing, the FET analysis, an postprocessing. In section 8.1, overview of the FET system is introduced following the diagram in figure 8.1. The preprocessing stage in section 8.2 produces the focus information in form of a set of focus words. The LKB system with ERG grammar parses a sentence and obtains the MRS representation. This representation is transformed to a set of focus words, which are provided to the next stage. The FET analysis in section 8.3 uses the LKB system with the FET subgrammar to parse a sentence with focus information to annotate prosodic marks on each word. The basis of the FET analysis is explained in chapters 5 and 6 and the FET subgrammar has been described in chapter 7. The result is the FET structure including prosodic marks. The last process extracts the prosodic marks from the FET structure and modifies prosody following their marks. This postprocessing process is described in the section 8.4.

8.1 Overview of the FET System for Prosodic Generation

The FET system generates the FET structure depending on the FET analysis. The FET structure is constrained by the speaker's intentions and focus parts. The diagram of the FET system is shown in figure 8.1 and an overview of the FET analysis based on the LKB system is explained below.

Inputs are a sentence and its focus group, provided by a user. In figure 8.1, the example sentence is "Kim bought a flower" and the focus group is "G" (see table 5.3). The FET system is composed of four main steps.

In the first step, the LKB system with the ERG [3] parses the sentence, analyzes the syntactic and semantic structures, and generates the MRS [4] representation. This step occurs before invoking the FET analysis.

In the second step, the MRS structure is scanned and any of components and the relations among them obtained from the preprocessing step are collected. Only the

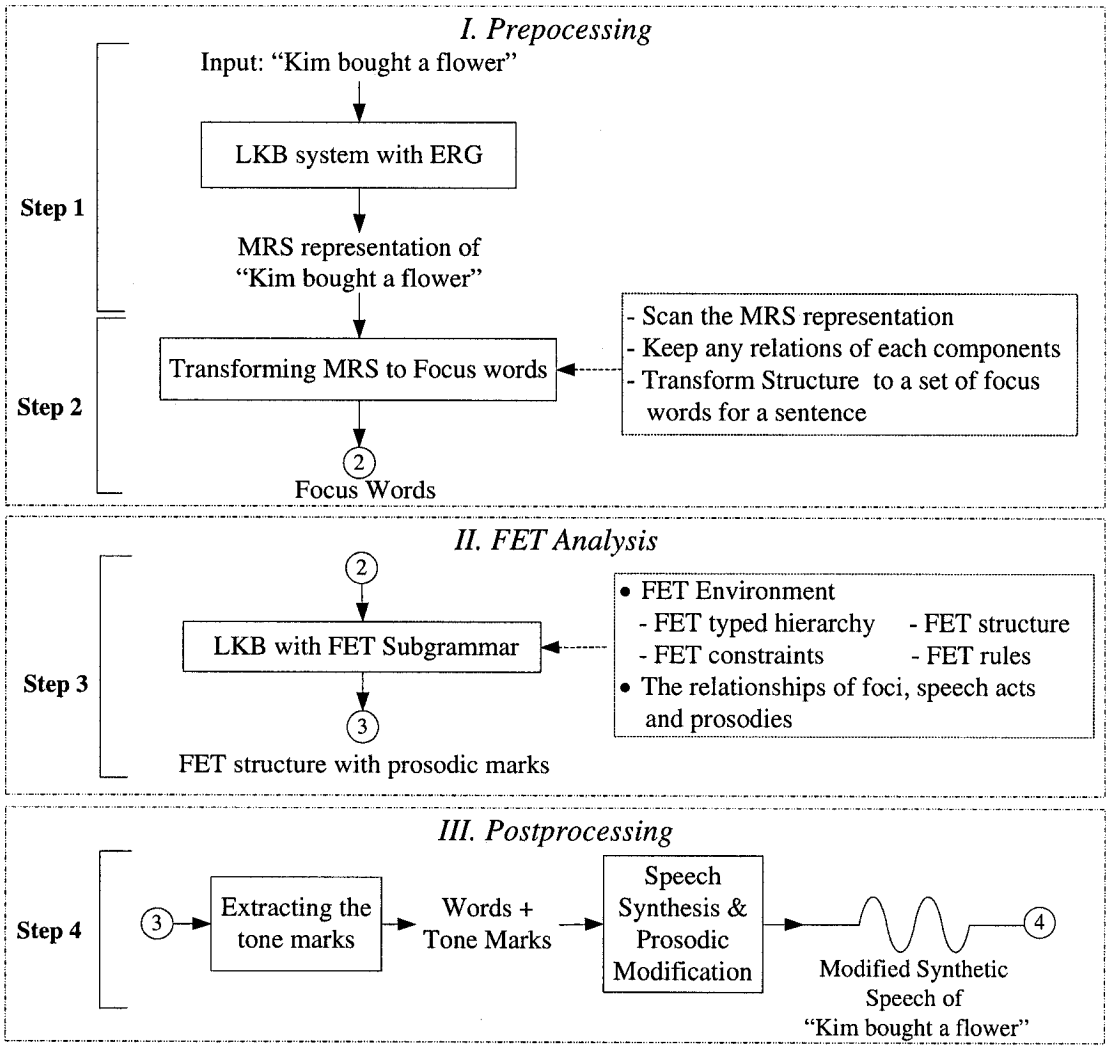


Figure 8.1: Diagram of the FET system

required information, such as sentence mood, selected from the MRS representation, a speech act code referring to a main verb, or main action of the sentence are assigned. The MRS structure is transformed to a set of focus words. These focus words are an input for the FET analysis..

In the third step, the FET analysis can generates the FET structure including the prosodic components. Using FET subgrammar, the focus words are provided to the LKB system with the FET subgrammar. This subgrammar consists of the FET typed hierarchy, constraints, rules, and their structures for the focus and prosodic features. Since the LKB system with FET subgrammar analyze the focus relations

corresponding to speech acts and sentence moods, the system completes the FET structure by generating a set of appropriate prosodic structures containing tone marks as a result.

In the last step the words and their prosodic marks such as ToBI representations [5] are extracted from the FET structure. Only our required prosodic fields are extracted from the FET structure. These fields are a set of words and their tone marks for a sentence. The set of words with tone marks is used to modify prosody of synthetic speech by using *Praat* [6] with our prosodic modification module. The output is an audio file of the sentence with modified prosody.

8.2 Preprocessing

In the first step, the system obtains an input sentence from user. In this example, "Kim bought a flower for her mother" is our input. The LKB system, which is an HPSG parser, with the ERG grammar parses the sentence. The LKB system provides a result in form of a syntactic tree and semantic structure called the MRS representation which is shown in figure 8.2.

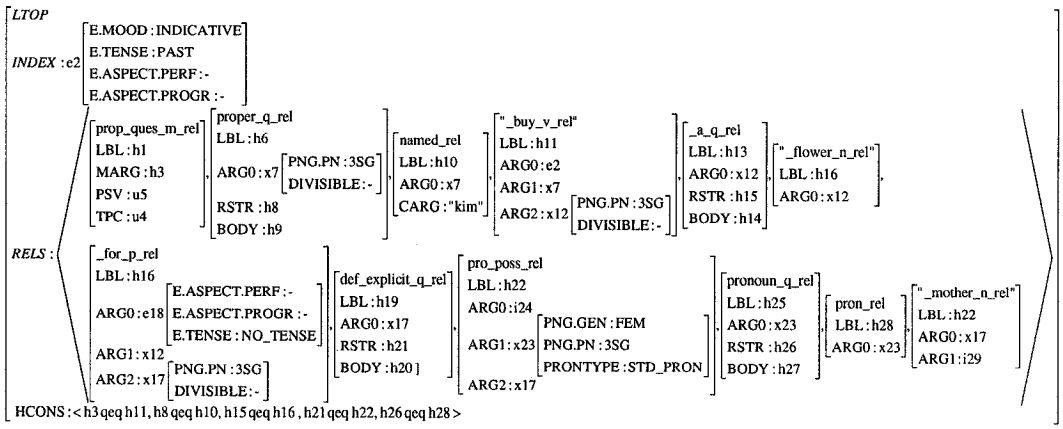


Figure 8.2: MRS of "Kim bought a flower for her mother"

The MRS representation is transformed to the AVM, as illustrated in figure 8.3. This AVM is used to generate the semantic structure and retrieve focus information, and it is issued to generate the FC structure. In figure 8.3, the AVM shows the relation between two arguments: ARG1 (actor part) and ARG2 (actee part). These arguments are connected by ARG0 (act part). The act part is represented by the

main verb "buy" while the actor and actee parts are represented by "Kim" and "a flower for her mother", respectively.

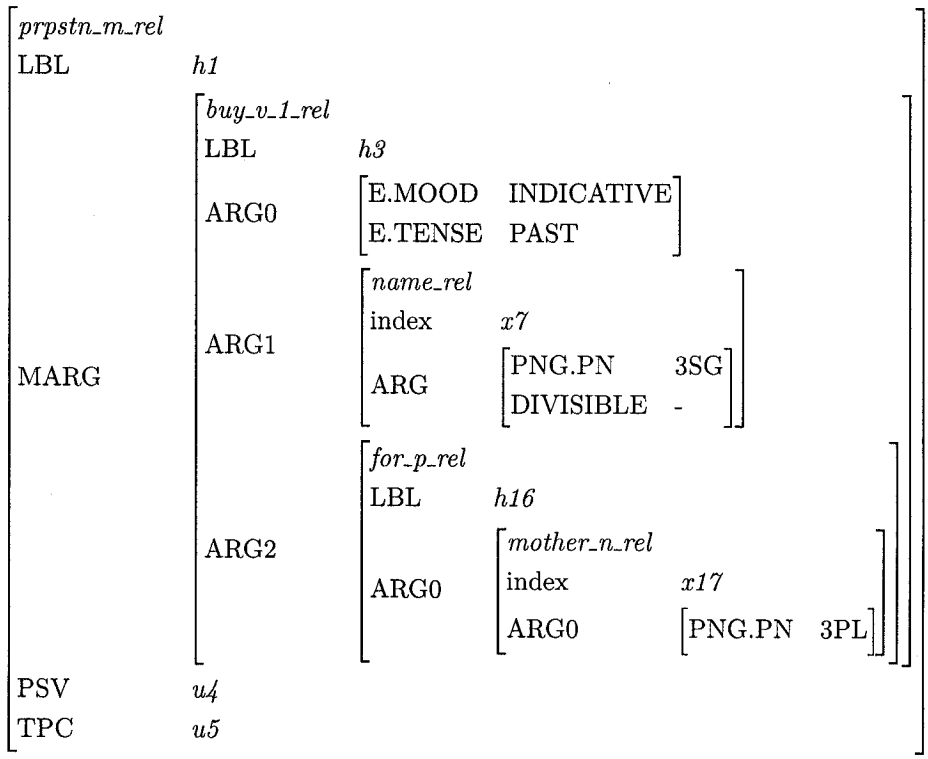


Figure 8.3: AVM of "Kim bought a flower for her mother"

The FC structure is transformed to a set of focus words described in chapter 4. The focus word structures are compatible with the unification-based formalism of the LKB system and they are ready to be provided to the FET analysis step. The focus words of the sentence "Kim bought a flower for her mother" are shown in figure 8.4. There are seven focus words for this sentence. A focus word structure contains four main features: ORTH, HEAD, SPR, and COMPS. ORTH represents the orthography of a word. HEAD, SPR, and COMPS inherit from focus structure (*focus-struc*). Inside *focus-struc*, HEAD identifies the focus components of a focus word. SPR is called *specifier* and constrains the components of the previous node which can be a word or list of words. COMPS is called *complement.* It constrains focus components of the following node. In this figure, the focus parts of the sentence "Kim bought a flower for her mother" are $\{[bought]_{act}, [Kim]_{actor}, [[a,flower], [for,her,mother]]_{actee}\}$. For example, at the focus word "bought", HEAD is *act–part* with the following focus components: focus position is last position (*ls*), focus group is "*G*", sentence type

is interrogative sentence (*int*) and speech act code is intending (*EN0ab*). SPR and COMPS is constrained by *actor–part* and *actee–part* with the same components.

8.3 FET Analysis

The focus words, which are assigned speech act code and focus group for each word, are provided to the LKB system with the FET subgrammar. The FET subgrammar consists of the FET structure, typed hierarchy, rules and constraints. Since the LKB system with FET subgrammar can analyze the focus relations corresponding to speech acts and sentence types, the system generates the appropriate prosodic marks for each word in a sentence. The result of this analysis is the FET structure with the prosodic components such as tone marks which are shown in the last section. To understand the FET structure, an example of the FET structure with its explanation is shown in figure 8.5. The focus word "Kim" is composed of four main features: *ORTH, HEAD, SPR,* and *COMPS. ORTH* represents the orthography of "Kim". *HEAD* is declared as *actor–part.* The focus components of HEAD are focus position (*FOCUS-POS*), focus group (*FCGROUP*), focus type (*FCTYPE*) and prosodic structure (*PROSODY*). The *PROSODY* structure contains sentence type (*STMOOD*), speech act code (*SPCODE*), accent tone (*ACCENT–TONE*) and boundary tone (*BOUND–TONE*). *SPR* and *COMPS* are declared as *NULL* in this example. The descriptions of these features and their structures appear in chapter 7. The complete FET structure of "Kim bought a flower for her mother" for the LKB system with the FET subgrammar is shown in appendix A.2.

8.4 Postprocessing

In the next step, the words with tone marks are extracted from the FET structure as shown in figure 8.6. The Festival speech synthesis [75] is employed to generate the synthetic sound in form of a wave file for the sentence and the prosody is modified by using Praat with our prosodic modification module described in the next paragraph. For example, a wave file with textgrid is shown in figure 8.7. The textgrid represents a set of words of a sentence which are annotated with tone marks for a sentence. These words and their tone marks in textgrid are derived from the FET structure

```
Kim := focus-word & [
     ORTH    "Kim",
     HEAD    actor-part & [ AGR1 ls-actor_G-int-en0ab ]
     SPR     <> ,
     COMPS   <> .]
bought := focus-word & [
     ORTH    "bought",
     HEAD    act-part & [ AGR1 ls-act_G-int-en0ab ],
     SPR     < [HEAD actor-part &
               [ AGR1 ls-actor_G-int-en0ab ] ] >,
     COMPS   < focus-phrase & [HEAD actee-part &
               [ AGR1 ls-actee_G-int-en0ab ]] > ].
a := focus-word & [
     ORTH    "a",
     HEAD    actee-part & [ AGR1 pv-actee_G-int-en0ab ],
     SPR     <> ,
     COMPS   <> ].
flower := focus-word & [
     ORTH    "flower",
     HEAD    actee-part & [ AGR1 ls-actee_G-int-en0ab ],
     SPR     < [ HEAD actee-part &
               [ AGR1 pv-actee_G-int-en0ab ] ] >,
     COMPS   < focus-phrase & [HEAD actee-part &
               [ AGR1 ls-actee_G-int-en0ab ] ] > ].
for := focus-word & [
     ORTH    "for",
     HEAD    actee-part & [AGR1 pv-actee_G-int-en0ab ],
     SPR     <>,
     COMPS   <> ].
her := focus-word & [
     ORTH    "her",
     HEAD    actee-part & [AGR1 pv-actee_G-int-en0ab],
     SPR     < [ HEAD actee-part &
               [ AGR1 pv-actee_G-int-en0ab ] ] >,
     COMPS   <> ].
mother := focus-word & [
     ORTH    "mother",
     HEAD    actee-part & [AGR1 ls-actee_G-int-en0ab],
     SPR     < [ HEAD actee-part &
               [ AGR1 pv-actee_G-int-en0ab ] ] >,
     COMPS   <> ].
```

Figure 8.4: Focus words of "Kim bought a flower"

```
[ focus-word
 HEAD: [ actor-part
         FOCUS: ACTOR
         AGR1: [ ls-actor_g-int-en0ab
                 FOCUS-POS: LAST
                 FCGROUP: G
                 FCTYPE: WS-FOCUS
                 PROSODY: [ int-en0ab-no-prosody
                            STMOOD: INT
                            SPCODE: EN0AB
                            PROSODY-SET: [ no-prosody
                                           PROSODY-MARK1: [ no-mark
                                                            ACCENT-TONE: NOACCENT
                                                            BOUND-TONE: NOBOUND ] ]
                                           PROSODYMARK2: [ no-mark
                                                           ACCENT-TONE: NOACCENT
                                                           BOUND-TONE: NOBOUND ]
         ] ] ] ] ] ] ] ] ]
SPR: *NULL*
COMPS: *NULL*
ORTH: Kim ]
```

Figure 8.5: FET structure of the word "Kim"

in the previous step. The wave file is the synthetic speech generated by the Festival speech system.

ORTH:	Kim	ORTH:	bought
Focus:	actor–part	Focus:	act–part
ACCENT_TONE1:	NOACCENT	ACCENT_TONE1:	NOACCENT
BOUND_TONE1:	NOBOUND	BOUND_TONE1:	NOBOUND
ACCENT_TONE2:	NOACCENT	ACCENT_TONE2:	NOACCENT
BOUND_TONE2:	NOBOUND	BOUND_TONE2:	NOBOUND
ORTH:	a	ORTH:	flower
Focus:	actee–part	Focus:	actee–part
ACCENT_TONE1:	NOACCENT	ACCENT_TONE1:	L+H*
BOUND_TONE1:	NOBOUND	BOUND_TONE1:	H-
ACCENT_TONE2:	NOACCENT	ACCENT_TONE2:	NOACCENT
BOUND_TONE2:	NOBOUND	BOUND_TONE2:	NOBOUND
ORTH:	for	ORTH:	her
Focus:	actee–part	Focus:	actee–part
ACCENT_TONE1:	NOACCENT	ACCENT_TONE1:	NOACCENT
BOUND_TONE1:	NOBOUND	BOUND_TONE1:	NOBOUND
ACCENT_TONE2:	NOACCENT	ACCENT_TONE2:	NOACCENT
BOUND_TONE2:	NOBOUND	BOUND_TONE2:	NOBOUND
ORTH: mother			
Focus:	actee–part		
ACCENT_TONE1:	L+H*		
BOUND_TONE1:	H-H%		
ACCENT_TONE2:	NOACCENT		
BOUND_TONE2:	NOBOUND		

Figure 8.6: Words annotated with tone marks of "Kim bought a flower for her mother"

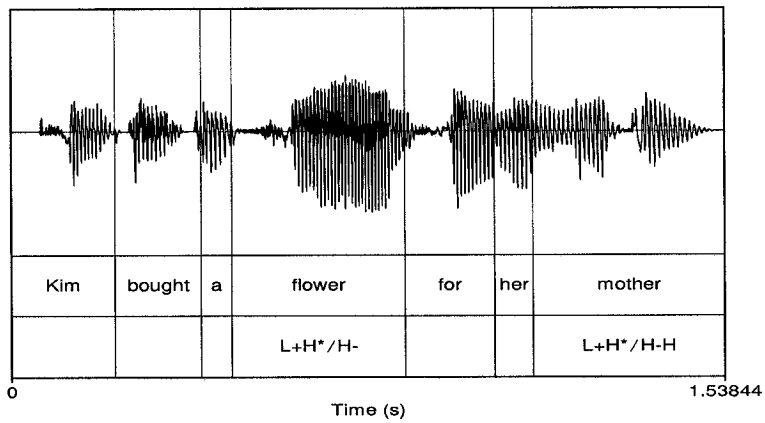


Figure 8.7: Waveform and textgrid of the sentence "Kim bought a flower for her mother"

To modify prosody, a set of prosodic features is based on three domains: frequency, time and intensity domains. The frequency domain relates to tone, fundamental frequency (f_0), and pitch contour. The features in time domain are duration and break. In the intensity domain, the loudness is represented by the intensity contour. To emphasize prosody following ToBI marks, the four main features; tone, duration, break, and loudness, are modified for each focus part.

8.4.1 Tone Modification

For tone modification, the initial values of frequency range and frequency levels are defined to limit frequency area of tone modification. This range varies depending on the speaker characteristics such as male or female and child or adult. Based on ToBI system, four frequency levels: baseline, below baseline, midline, and topline are assigned to control the tone modification. For example, in this work, frequency range is defined between 95 Hz and 145 Hz for male voice. The baseline, below baseline, midline, and topline are 115 Hz, 95 Hz, 125 Hz, and 145 Hz, respectively, as shown in figure 8.8. These frequency values depended on the frequency levels of the speaker in the synthetic waveform. The high tone marks (H*, H-) lie between midline and topline while the low tone marks (L*, L-) lie between midline and baseline. The rising tone (L+H*) starts from below baseline and changes to topline. The comparison between the original pitch contour and pitch contour after modifying tone is illustrated in

figure 8.9. At the word "mother" in figure 8.9(a), the original pitch contour lies between baseline and below baseline. The pitch contour of "mother" in figure 8.9(b) is drawn from baseline to below baseline then to topline following tone mark L+H* H-H%.

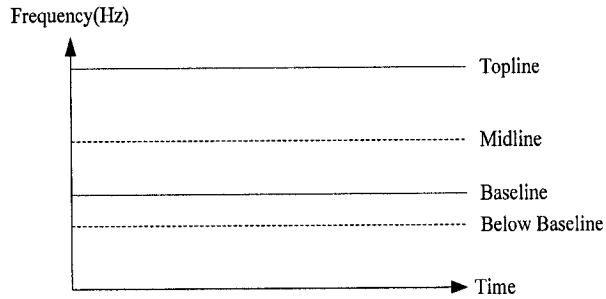


Figure 8.8: Range of frequency

8.4.2 Duration Modification

Duration modification can improve the listener's recognition at a focus word. When the speech duration at a focus word is increased (slow down the speech rate), the listener can recognize the focus word better. The duration is adjusted, depending on the length of focus words. If a focus word is a short word, such as a mono-syllable word, then the proportion of increasing duration needs to be greater than the proportion of increasing duration of multi-syllable words. For example, in this modification, the duration of words can be expanded twice of the original duration of the short word marked with sentence break and 1.5 time of the original duration of the short word marked with phrase break. This example shows the comparison between the original duration (figure 8.10(a)) and the modified duration (figure 8.10(b)) of the word "flower" in figure 8.10.

8.4.3 Break Insertion

Break insertion depends on the boundary tone marks. Three levels of breaks are no break, phrase break, and sentence break based on the ToBI system. The sentence break is the longest break which is usually placed at the end of a sentence. The phrase break is a short break. It can be inserted between phrases such as noun and

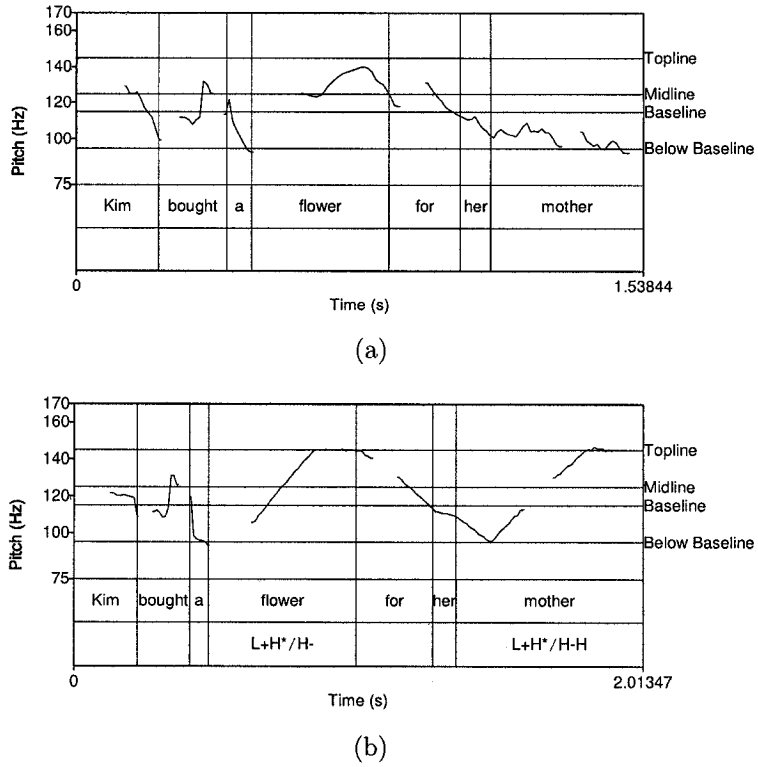
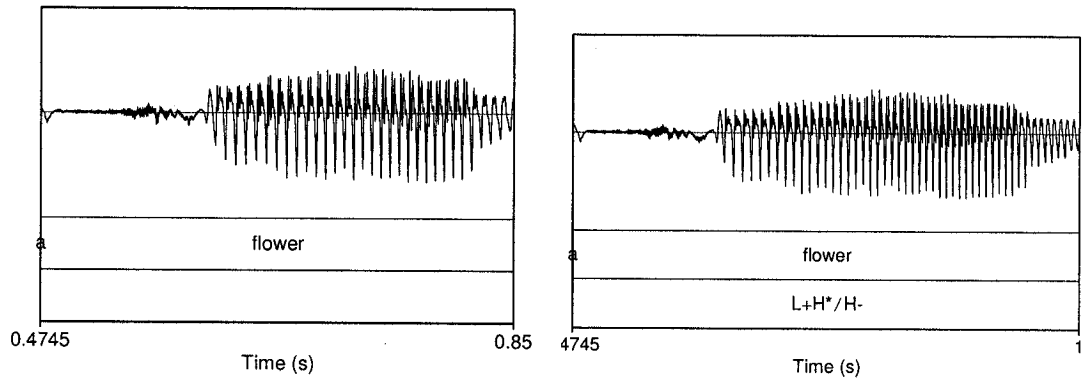


Figure 8.9: Comparison between (a) original pitch contour and (b) modified pitch contour

preposition phrases or between the conjunctions "and", "or", and so on. For example, the sentence and phrase breaks are defined as 0.5 and 0.3 seconds respectively in this work. The phrase break can be inserted at the end of boundary tones or between focus parts. Since break insertion reduces the smoothness of synthetic sound, applying cosine window function can help to the sounds which are connected to the breaks as shown in figure 8.11. The window convolutes a number of samples at the end of sound. This convolution can smooth the loudness at the end of sound before break.

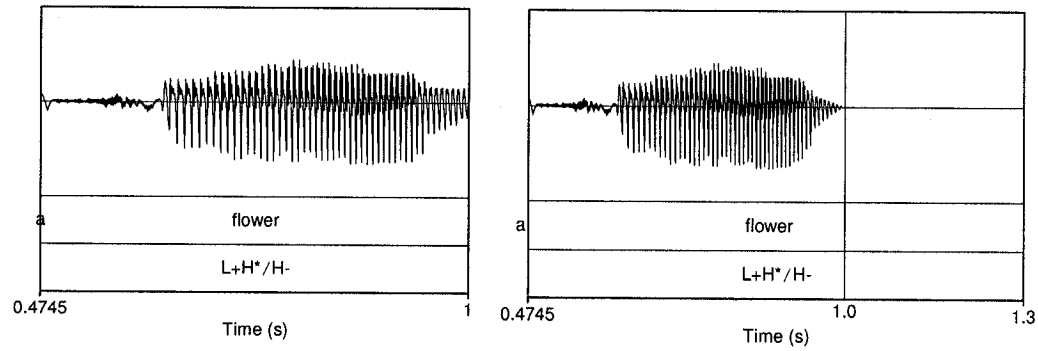
8.4.4 Loudness Adjustment

The loudness must be increased at the focus words which are marked by emphasized tone. Listener has the better recognition at these focus words when the loudness is increased. Similar to using the intensity contour to adjust the loudness, the loudness of a focus word is increased by multiplying the amplitude with a proportion higher than one of the original amplitude. In this work, two levels of loudness are assigned



(a) (b)

Figure 8.10: Comparison of the duration between (a) original waveform and (b) modified waveform



(a) (b)

Figure 8.11: Smoothing amplitude and inserting break (a) original waveform and (b) modified waveform

for the emphasized tones (such as H*) and high emphasized tone (such as L+H*). For example, the level of emphasized tone can be defined with 1.5 times of the original amplitude while the level of highly emphasized tone can be defined with 1.75 times of the original amplitude. The comparison of the intensity (loudness) contour between the original sound and modified sound of the word "flower" is shown in figure 8.12. The maximum intensity of the original sound is 82.53 dB in figure 8.12(a) and the maximum intensity of modified sound is 86.41 dB in figure 8.12(b).

Finally, the output is a wave file of the sentence with modified prosody which is controlled by the tone marks. The waveform of synthetic speech with modified tone is illustrated in figure 8.13 and pitch contour of this speech is shown in figure 8.14.

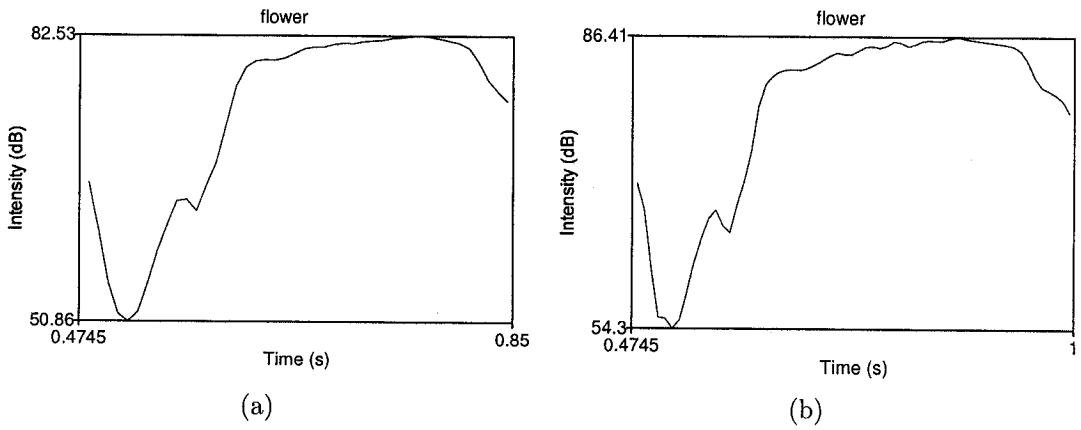


Figure 8.12: Comparison between (a) original intensity contour and (b) modified intensity contour

Considering the pitch contour of the modified speech, the slope changes obviously from low to high at [*a,flower*] following the marks `L+H* H-`. The slope drops after [*flower*] or at the beginning of [*for,her,mother*] and then raises to high tone until the end of [*for,her,mother*] following the marks `L+H* H-H%`. This pitch contour is controlled by the tone marks deriving from the FET structure.

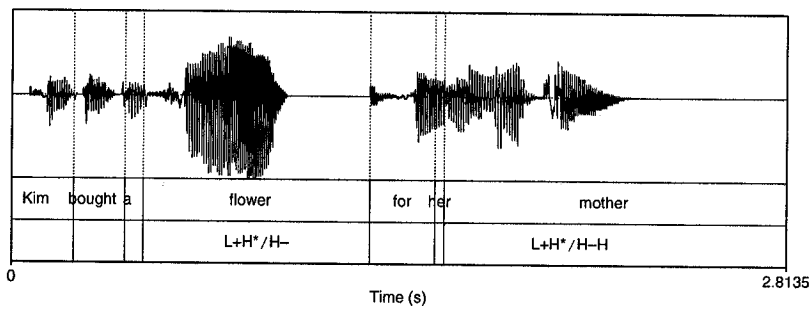


Figure 8.13: Waveform of "Kim bought a flower for her mother"

8.5 Summary

In this chapter, we implemented the FET system as a unification-based system for prosodic generation. The FET system consists of three main steps: preprocessing step, FET analysis step and post processing step. In the preprocessing step, we

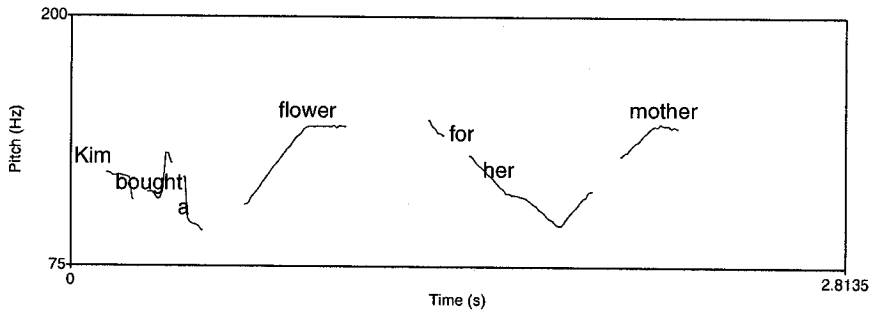


Figure 8.14: Pitch contour annotated with tone marks of "Kim bought a flower for her mother"

transformed the MRS representation, deriving from the LKB system with ERG, to a set of focus words provided to the next step. In the second step, the LKB system with the FET subgrammar parses a sentence and its result is the complete FET structure annotated with the tone marks. In the last step, a set of words and their tone marks are extracted from the FET structure. Since these words are sent to generate the synthetic speech then tone marks are used to modify prosody of this sentence by using our prosodic modification module. The result is a wave file of the sentence with modified prosody.

Chapter 9

Evaluation of the FET System

Two evaluations are proposed (i) perceptual evaluation of focus conveyed by emphasizing tone, and (ii) comparison of prosodic annotation. The first evaluation is a subjective evaluation. It proposes to determine whether the prosody annotated by the FET system can convey the focus contents to the listeners. The synthetic speech was modified to follow the prosodic marks needed to evaluate whether the sounds can convey focus by emphasizing prosody. A listening test is performed to measure the listener's preferences. The details of perceptual evaluation plans are described in the next section. The experimental results of perceptual evaluation are analyzed in section 9.2. The second evaluation is the comparison of the prosodic annotation between the FET system and the CMU-COM dataset. For the second evaluation, the details and experimental results are explained in section 9.3.

9.1 Design of Experiment for Perceptual Evaluation

Listening tests are performed for our perceptual evaluation. Experimental design for perceptual evaluation is composed of two main parts. The first part in section 9.1.1 is to evaluate the listener's preference between the sounds with focus and without focus. The second part is the evaluation of focus conveyed by prosodic emphasis described in section 9.1.2. Each part contains two dialogues for our listening tests. These dialogues are a conversation about a travel reservation between a travel agent and a caller. These dialogues are selected from the air travel domain in the ATIS project [76].

9.1.1 Design of Experiment to Analyze the Preference of Sounds with and without Focus

The purpose of this evaluation is to compare the sounds of the sentences with focus by emphasizing prosody and sentences without focus. Another purpose is to test whether the listener can recognize the focus part in a sentence and can select the sound that makes the most sense in the sentence. Modifying the prosody of sounds is

controlled by prosodic marks generated from our FET system. In the test, listeners are given a dialogue. They compare the sound utterances with and without modified prosody for the same sentence. Four sounds of the same sentence are played and the listeners select utterance that makes the most sense and is suitable for the dialogue. The test dialogues are selected from the travel reservation domain. For example, the dialogue in figure 9.1 is the conversation between Peter, a travel agent, and Tom, who is a caller. Listeners listen to the sounds of sentences from this dialogue in order and they select the most suitable sounds for the situation from the multiple choices that they are given e.g., (A)-(D) in figure 9.1.

Dialogue A: Travel Reservation

Line 01	Peter:	Halifax Travel. Peter speaking.
Line 02	Tom:	Hello, I am Tom.
Line 03	Tom:	I would like to make some travel arrangements?
Line 04	Peter:	What day did you wish to travel?
Line 05	Tom:	This is May 12th.
Line 06	Peter:	Fly from Toronto.

At this point, listener must select the most suitable sound utterance from one of four choices for this sentence

(A)	Fly from Toronto.
(B)	Fly from Toronto?
(C)	Fly from $[\textbf{Toronto}]_F$.
(D)	Fly from $[\textbf{Toronto?}]_F$
Answer (1):	_____

Line 07	Tom:	Fly from Toronto.
Line 08	Peter:	And going where?
Line 09	Tom:	going to Boston.
Line 10	Tom:	I believe the United flight.

At this point, listener must select the most suitable sound utterance from one of four choices for this sentence

(A)	I believe the United flight.
(B)	I believe the United flight?
(C)	I believe $[\textbf{the United flight}]_F$.
(D)	I believe $[\textbf{the United flight?}]_F$
Answer (2):	_____

Line 11	Peter:	That is eight a.m.
Line 12	Tom:	Right.

Figure 9.1: Dialogue A: Example dialogue for comparison between the sounds with and without focus

In figure 9.1, dialogue A includes two test sentences (line 06 and 10). Each sentence have four choices of sounds with different prosodies. These choices can be divided into two types: (i) question or non-question sentence and (ii) sound with focus, which is emphasized by prosody, or without focus. Choice A is the sound of non-question sentence without focus while Choice B is the sound of question sentence without focus in the sentence. Choice C and D include focus in the sentence, but choice D

is a question sentence while choice C is a non-question sentence. Considering these choices, we compare whether listeners can recognize focus and select the sentence type that makes the most sense to the situation in the dialogue.

9.1.2 Design of Experiment to Analyze Focus Conveyed by Emphasizing Prosody

The purpose is to evaluate whether subject can recognize the different focus parts emphasized by prosody and whether the prosody can convey our expected contents to the listeners. In this test, the listeners choose the appropriate focus part, which makes the most sense in the dialogue. From four choices, listeners must select a suitable sound which emphasized prosody at the correct focus part for the sentence in a given dialogue. The listeners receive the dialogue about the travel reservation. They must recognize the tone emphasis at different focus parts in the same sentence and choose one sound which is appropriate to the situation in the dialogue.

For example, in dialogue B in figure 9.2, travel agent "Peter" and caller "Tom" were talking about travel reservation. Listeners must listen to this dialogue and select one of four choices of sounds at test sentences. In each test sentence, the sounds of these choices have tone emphasis at the different focus parts. Listener must select the most suitable sounds corresponding to the dialogue's situation from four multiple choices provided for each test sentence.

In figure 9.2, there is one test sentences at line 08. The test sentence has four choices of sounds which have tone emphasis at actor, act and actee parts. For choice A, the sound does not have the tone emphasis at any focus part which is used as a control group. Choice B is the sound emphasized by tone at actor part. For choices C and D, the sounds are emphasized by tone at the act and actee parts orderly. This test evaluates whether listeners can recognize the different focus parts and select the most suitable focus for a situation. Since our purpose is to explore the effect of emphasizing prosody at the different focus parts, then result of this evaluation will confirm whether the listener can recognize the focus conveyed by prosody.

Dialogue B: Travel Reservation

Line 01	Peter:	Halifax Travel. May I help you?
Line 02	Tom:	Yes, this is Tom.
Line 03	Tom:	I would like to make some travel reservation.
Line 04	Peter:	What do you need to do?
Line 05	Tom:	In August. I need to go to Southbend, Indiana.
Line 06	Tom:	And I want to fly on the United flight.
Line 07	Peter:	One second.
Line 08	Peter:	What time do you want to leave?

At this point, listener must select the most suitable sound utterance from one of four choices for this sentence

(A)	What time do you want to leave?
(B)	**[What time]**$_F$ do you want to leave?
(C)	What time do **[you]**$_F$ want to leave?
(D)	What time do you **[want to leave]**$_F$?

Answer (3): _____

Line 09	Tom:	I want to go from Boston to Southbend on the United flight 262
Line 10	Peter:	OK. That leaves at 4:50 p.m.

Figure 9.2: Dialogue B: Example dialogue for comparison among sounds of different focus parts

9.2 Experimental Results of Perceptual Evaluation

The listening tests have been performed following our design of experiments in section 9.1. The results are collected from a group of participants. In this section, the results are analyzed to measure whether subject can recognize focus emphasized by tone. This evaluation is calculated by using the Analysis of Variance (ANOVA) method. This evaluation method considers the significance of effects which are focus parts.

9.2.1 Analysis of Difference between Dialogues

In table 9.1, the dialogues 1 and 2 in the first section of Appendix B are used to compare between the sentences with and without focus for affirmative and interrogative sentences. The second section is used to compare the different focus parts on a sentence. The subjects or listeners must choose the sounds that make the most sense for the sentence. The number of correct answers of each subject for each dialogue is shown in table 9.1. The *correct answer* is the sound, selected by the subject, which matches our expected sound from one of multiple choices. There are 17 subjects participating in this perceptual evaluation. Each subject makes 20 answers from multiple choice questions and has 20-30 minutes to complete the experiment. There are five answers for each dialogue. These subjects are volunteers recruited from the computer science students. They include native and non-native English speakers the number of

participants varies from 8 to 30 subjects [77, 78] depending on the focus group, difficulty, and duration of the experiments. Although there are seventeen subjects in our experiment, each subject answers twenty questions which amount to a large number of measurements. We also need to consider the distribution of data which can affect our conclusions by using the ANOVA method. Therefore the normal probability plots are drawn to confirm the normal distribution of data.

Table 9.1: Number of correct answers from 17 subjects

Subject	No. of Correct Answer						
	Section 1		Results of	Section 2		Results of	Total
	Dialogue 1	Dialogue2	Section1	Dialogue 3	Dialogue4	Section 2	Result
1	5	5	10	4	5	9	19
2	3	4	7	2	2	4	11
3	5	4	9	2	1	3	12
4	4	3	7	4	5	9	16
5	4	4	8	5	3	8	16
6	2	5	7	4	5	9	16
7	3	5	8	0	0	0	8
8	4	5	9	2	1	3	12
9	4	4	8	0	1	1	9
10	3	4	7	0	3	3	10
11	3	5	8	4	2	6	14
12	3	4	7	0	0	0	7
13	4	5	9	3	2	5	14
14	5	4	9	2	3	5	14
15	5	4	9	5	4	9	18
16	3	4	7	3	2	5	12
17	3	5	8	5	2	7	15

Considering the number of correct answers, there is a possibility that the different dialogues in the same section can affect the decision of subjects to choose the correct answer. A subject's decision may be affected by the sound quality, and faulty prosodic modification. We need to prove that no difference between dialogues in the same section is statistically significant.

Experimental Results of the Difference between Dialogues 1 and 2

Comparing between dialogues 1 and 2, the number of correct answers for each subject is shown in Table 9.1. Based on our calculation, the ANOVA table in figure 9.3, shows that the F ratio of dialogues is $5.82 > F_{0.05,1,32} = 4.17$. We conclude that the different dialogues affect the number of correct answers. Following the individual

```
One-way ANOVA: tol_d1, tol_d2
tol_d1 = No.  of correct answer in dialogue 1
tol_d2 = No.  of correct answer in dialogue 2

Source  DF      SS     MS    F      P
Factor   1   3.559  3.559  5.87  0.021
Error   32  19.412  0.607
Total   33  22.971

S = 0.7789   R-Sq = 15.49%    R-Sq(adj) = 12.85%

Individual 95% CIs For Mean Based on Pooled StDev
Level    N    Mean    StDev   -------+---------+---------+---------+--
tol_d1  17  3.7059  0.9196    (---------*--------)
tol_d2  17  4.3529  0.6063                      (---------*--------)
                               -------+---------+---------+---------+--
                                     3.60      4.00      4.40      4.80
Pooled StDev = 0.7789

Tukey 95% Simultaneous Confidence Intervals
All Pairwise Comparisons
Individual confidence level = 95.00%

tol_d1 subtracted from:
      Lower  Center   Upper      -+---------+---------+---------+--------
tol_d2 0.1029 0.6471 1.1912                   (----------*----------)
                                 -+---------+---------+---------+--------
                                -0.50       0.00      0.50      1.00
```

Figure 9.3: ANOVA table, individual 95% CIs, and Turkey test for the comparison between dialogues 1 and 2

95% confidence interval (CI), the intervals of dialogue 1 and 2 slightly overlap so that the population means of these dialogues may not be different. However, when we compare dialogues 1 and 2 by using Turkey's test [79], the interval of pair of dialogues does not include zero. The pair of population means (between dialogue 1 and 2) are significantly different, which supports our conclusion. Furthermore, the normal probability plot in figure 9.4 lies on the straight line which also supports the difference between dialogues 1 and 2.

There is some evidence that the difference between dialogues affects the number of correct answers. From our observation, dialogue 1 contains two ambiguous sentences which affect the selection of some subjects while dialogue 2 has no ambiguous sentences. For instance, "Again on the United flight" is difficult to decide whether it is a question sentence. More investigation and discussion of the differences in dialogues 1 and 2 is described in the section 9.2.2.

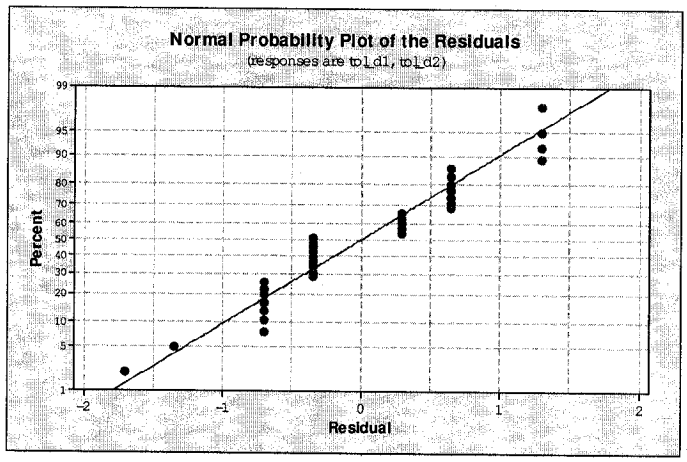


Figure 9.4: Normal plot of dialogues 1 and 2

Experimental Results of the Difference between Dialogues 3 and 4

When comparing between dialogues 3 and 4 in Appendix B, our assumption is that there is no statistical difference between dialogues. To prove the assumption, the ANOVA table in figure 9.5 is calculated and we conclude that the difference between dialogues is not statistically significant because the F ratio of dialogues is $0.16 < F_{0.05,1,32} = 4.17$. This assumption is confirmed by using the individual 95% confident interval and Turkey's test. The intervals of dialogues 3 and 4 overlap for the individual CIs and the interval of pair of dialogues includes zero for the Turkey's test. These results and the normal probability plot in figure 9.6, which lies on the straight line, support the assumption that there is no statistically significant difference between dialogues 3 and 4.

9.2.2 Analysis of Difference among Choices of Sounds

In this section, we need to examine whether there is statistical difference among four multiple choices A, B, C, and D. The objective of this evaluation is to conclude whether listeners can recognize the focus parts emphasized by prosody and can choose the focus part that makes the most sense in a sentence according to our system. In the previous section, there is a statistical difference between dialogues 1 and 2 while there is no difference between dialogues 3 and 4. Therefore, we consider the dialogues 1 and 2 separately, while the dialogues 3 and 4 can be calculated together.

```
One-way ANOVA: tol_d3, tol_d4
tol_d3 = No.  of correct answer in dialogue 3
tol_d4 = No.  of correct answer in dialogue 4

Source  DF     SS    MS     F      P
Factor   1   0.47  0.47  0.16  0.695
Error   32  96.00  3.00
Total   33  96.47
S = 1.732   R-Sq = 0.49%   R-Sq(adj) = 0.00%

Individual 95% CIs For Mean Based on Pooled StDev
Level    N   Mean  StDev  ---------+---------+---------+---------+
tol_d3  17  2.647  1.835       (-----------------*-----------------)
tol_d4  17  2.412  1.622  (-----------------*-----------------)
                          ---------+---------+---------+---------+
                                 2.00      2.50      3.00      3.50
Pooled StDev = 1.732

Tukey 95% Simultaneous Confidence Intervals
All Pairwise Comparisons
Individual confidence level = 95.00%
tol_d3 subtracted from:
        Lower  Center  Upper    -+---------+---------+---------+--------
tol_d4 -1.445 -0.235  0.975    (-----------------*-----------------)
                               -+---------+---------+---------+--------
                             -1.40     -0.70      0.00      0.70
```

Figure 9.5: ANOVA table, individual 95% CIs, and Turkey test for the comparison between dialogues 3 and 4

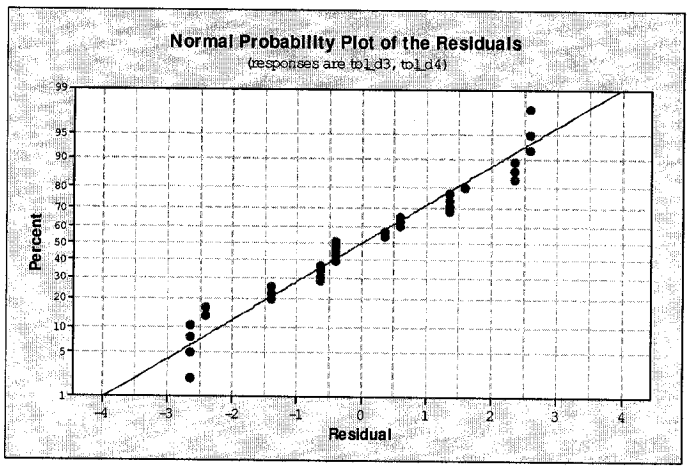


Figure 9.6: Normal plot of dialogues 3 and 4

Experimental Results of the Difference among Choices of Sounds in Dialogue 1

In dialogue 1 and 2, the choices A, B, C, and D are the representations of "non-question sentence without focus", "question sentence without focus", "non-question sentence with focus", and "question sentence with focus" respectively. We want to

investigate the decision of subjects to select these choices, i.e. choosing between sentence with or without focus. Therefore, we analyze whether there is a statistical difference between these choices. Table 9.2 shows the number of selected choices A, B, C, and D for each subject from five answers.

Table 9.2: Number of selecting choices A, B, C, and D for each subject for dialogue 1

Subject No.	No. of Choices			
	A	B	C	D
1	0	0	2	3
2	1	0	1	3
3	1	1	1	2
4	0	0	1	4
5	2	1	1	1
6	0	1	1	3
7	4	1	0	0
8	0	2	1	2
9	1	2	0	2
10	2	1	0	2
11	1	1	1	2
12	3	1	1	0
13	0	3	1	1
14	2	0	0	3
15	0	2	2	1
16	1	1	1	2
17	1	1	1	2
Total	19	18	15	33

Considering the ANOVA table in figure 9.7, the F ratio is $4.21 > F_{0.05,3,64} = 3.68$. We conclude that the difference among choices A, B, C, and D is statistically significant. For the individual 95% confident interval, the interval of D does not overlap the intervals of B and C. Our conclusion is that the difference between population means of the pair of the question sentence with focus (D) and question sentence without focus (B), and the pair of non-question sentence with focus (C) and question sentence with focus (D) are statistically significant. These explain that the subjects can recognize the sound differences comparing between B and C, and between C and D. For the Turkey's test, the result also supports our conclusion from individual 95% confident interval. Since, the intervals of these pairs: between B and C, and between C and D, do not include zero, there are statistically significant differences between these

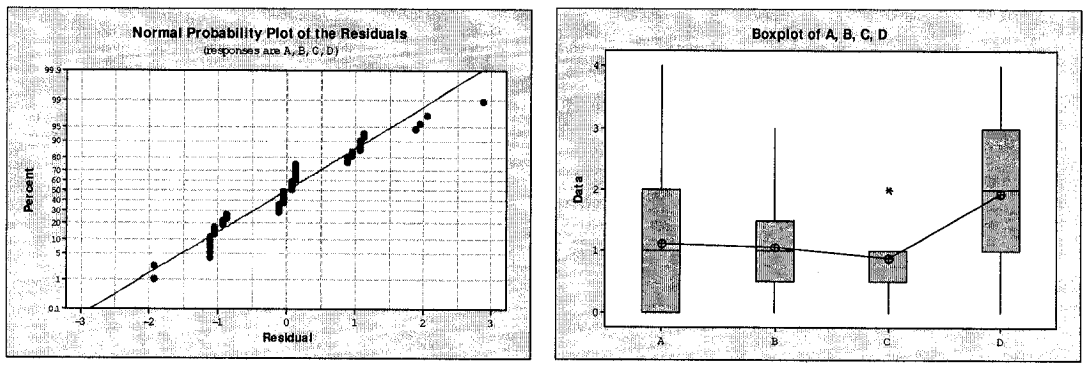
pairs of choices. To confirm the conclusion, we determine the normal probability plot which lies on the straight line. The plot supports our conclusion of the differences among choices. In the box plot, mean of choice D has the highest frequency and this data corresponds to the number of correct answers that choice D has the highest frequency of number of correct answers in dialogue 1. We conclude that most subjects can select the correct answers and recognize the differences between pairs of choices. Following our observation, the reason why choice A has the second highest frequency is that sometimes the subjects intend to select the correct answer but they change their mind and select choice A instead because of the low sound quality occurring by prosodic modification of those choices.

Experimental Results of the Difference among Choices of Sounds in Dialogue 2

For the dialogue 2, the number of subject's answers selecting choice A, B, C, and D is shown in table 9.3. Based on section 9.1, we know that dialogue 2 is different from dialogue 1. Based on the ANOVA table in figure 9.9, the F ratio is $10.84 > F_{0.05,3,64} = 3.68$ so that the difference among choices A, B, C, and D is statistically significant. Considering the difference between choices, the results of the individual 95% confident intervals permit the conclusion that there are statistically significant differences between choices A and B, A and C, and A and D. This conclusion is supported by the Turkey's test. We can conclude that the sound of choice A (non-question sentence without focus) is statistically different from the other choices: B (question sentence without focus), C (non-question sentence with focus) and D (question sentence with focus). To confirm our conclusion, we observe that the normal probability plot lies on a straight line so it supports the difference among choices as statistically significant. Considering the box plot, mean of choice A is the highest number and the second highest number is choice C. From these data, we can explain that many subjects select choice A as their most preferred answer because many subjects are not satisfied with the low quality of sounds and the sounds with vague prosody, based on subject's comments for dialogue 2. Therefore subjects select the sound of choice A which represents no prosodic modification. However, some subjects argue that the sounds are acceptable for dialogue 2. These subjects select choice C as the second

```
One-way ANOVA: Res versus ch
ch:  choice of A, B, C, and D
Res:  No.of answer choice A, B, C, and D

Source  DF      SS     MS     F      P
ch       3  11.338  3.779  4.21  0.009
Error   64  57.412  0.897
Total   67  68.750
S = 0.9471   R-Sq = 16.49%   R-Sq(adj) = 12.58%

Individual 95% CIs For Mean Based on Pooled StDev
Level   N    Mean   StDev  --+---------+---------+---------+-------
A      17  1.1176  1.1663       (---------*---------)
B      17  1.0588  0.8269      (---------*---------)
C      17  0.8824  0.6002  (---------*---------)
D      17  1.9412  1.0880                          (---------*---------)
                           --+---------+---------+---------+-------
                          0.50      1.00      1.50      2.00
Pooled StDev = 0.9471

Tukey 95% Simultaneous Confidence Intervals
All Pairwise Comparisons among Levels of ch
Individual confidence level = 98.95%

ch = A subtracted from:
ch    Lower   Center   Upper  ---------+---------+---------+---------+
B   -0.9157  -0.0588  0.7980            (--------*--------)
C   -1.0921  -0.2353  0.6215          (--------*--------)
D   -0.0333   0.8235  1.6804                    (--------*--------)
                           ---------+---------+---------+---------+
                                  -1.0       0.0       1.0       2.0
ch = B subtracted from:
ch    Lower   Center   Upper  ---------+---------+---------+---------+
C   -1.0333  -0.1765  0.6804          (--------*--------)
D    0.0255   0.8824  1.7392                    (--------*--------)
                           ---------+---------+---------+---------+
                                  -1.0       0.0       1.0       2.0
ch = C subtracted from:
ch   Lower  Center   Upper    ---------+---------+---------+---------+
D   0.2020  1.0588  1.9157                      (--------*--------)
                           ---------+---------+---------+---------+
                                  -1.0       0.0       1.0       2.0
```

Figure 9.7: ANOVA table, individual 95% CIs, and Turkey test for the comparison among choices in dialogue 1



(a) (b)

Figure 9.8: Graphs of (a) normal plot (b) box plot for dialogue 1

highest frequency and choice C is the highest frequency of number of correct answers. Our conclusion is that subjects select choice A, which is a non-question sentence without focus, for their most preferred answer. The subjects can recognize the differences between pairs of sound but they do not prefer the sound with focus emphasized by prosody for the second dialogue.

Table 9.3: Number of selecting choices A, B, C, and D of each subject for dialogue 2

Subject No.	No. of Choices			
	A	B	C	D
1	0	0	4	1
2	1	0	4	0
3	0	0	0	5
4	2	0	2	1
5	2	1	1	1
6	0	0	4	1
7	3	0	1	1
8	4	0	0	1
9	4	0	1	0
10	3	0	2	0
11	3	1	1	0
12	5	0	0	0
13	3	0	1	1
14	2	2	1	0
15	5	0	0	0
16	4	0	1	0
17	3	1	1	0
Total	44	5	24	12

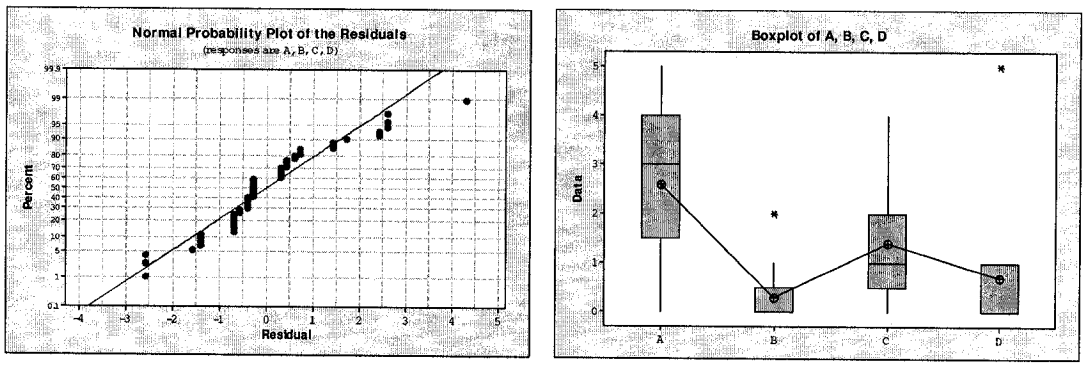
Experimental Results of the Difference among Choices of Sounds in Dialogue 3 and 4

Because there is no statistically significant difference between dialogues 3 and 4, the comparison among choices A, B, C, and D in dialogues 3 and 4 can be considered together. The number of subject's answers of choices A, B, C, and D is shown table 9.4. Choice A represents the sentence without focus while choices B, C, and D represent the sentence with the focus at the actor, act, and actee parts, respectively. We need to prove that there are statistically significant differences among these choices. From the ANOVA table in figure 9.11, the F ratio is equal to $6.66 > F_{0.05,3,64} = 3.68$ so that

```
One-way ANOVA: Res versus ch
ch:  choice of A, B, C, and D
Res:  No.of answer choice A, B, C, and D

Source  DF      SS     MS      F      P
ch       3   51.46  17.15  10.84  0.000
Error   64  101.29   1.58
Total   67  152.75
S = 1.258   R-Sq = 33.69%   R-Sq(adj) = 30.58%

Individual 95% CIs For Mean Based on Pooled StDev
Level   N   Mean  StDev  ---+---------+---------+---------+------
A      17  2.588  1.622                          (-----*-----)
B      17  0.294  0.588  (-----*-----)
C      17  1.412  1.372             (-----*-----)
D      17  0.706  1.213       (-----*-----)
                           ---+---------+---------+---------+------
                            0.0       1.0       2.0       3.0
Pooled StDev = 1.258

Tukey 95% Simultaneous Confidence Intervals
All Pairwise Comparisons among Levels of ch
Individual confidence level = 98.95%

ch = A subtracted from:
ch   Lower  Center   Upper  ---+---------+---------+---------+------
b   -3.432  -2.294  -1.156  (-------*------)
c   -2.315  -1.176  -0.038          (-------*-------)
d   -3.020  -1.882  -0.744      (------*-------)
                           ---+---------+---------+---------+------
                            -3.0      -1.5       0.0       1.5
ch = B subtracted from:
ch   Lower  Center  Upper  ---+---------+---------+---------+------
C   -0.020   1.118  2.256                          (------*-------)
D   -0.726   0.412  1.550                    (-------*------)
                           ---+---------+---------+---------+------
                            -3.0      -1.5       0.0       1.5
ch = C subtracted from:
ch   Lower  Center  Upper  ---+---------+---------+---------+------
D   -1.844  -0.706  0.432             (------*-------)
                           ---+---------+---------+---------+------
                            -3.0      -1.5       0.0       1.5
```

Figure 9.9: ANOVA table, individual 95% CIs, and Turkey test for the comparison among choices in dialogue 2

the difference among choices is statistically significant. For the confidence interval, there is no overlapping between choices B and D and between choices C and D. We can conclude that the subject can recognize the sound difference between focus at actor part and actee part and between the focus at act part and actee part. However, there is no sound difference between focus at actor and act parts. This description is supported by the Turkey's test. Between choices B and D and between choices C and D, their mean intervals do not include zero so these pairs have statistical differences. To confirm this conclusion, we consider the normal probability plot which lies on the straight line. The plot supports our conclusion of the difference among



(a) (b)

Figure 9.10: Graphs of (a) normal plot (b) box plot for dialogue 2

choices. Regarding the box plot, choice D has the highest mean corresponding to the highest number of correct answers which is also choice D. We observe that most subjects select the correct answers and recognize the sounds emphasized by prosody at the actee part. The second highest mean is choice A. Some subjects, who intend to select choice B, C, or D, decide to choose choice A and do not want to choose the other choices instead of A because they are not satisfied with prosody of those sounds. Based on the subject's comments, the subjects recognize that choice D is the correct answer, but they still decide to have choice A as their answer. They do not want to select the other choices B and C because the sounds of these choices cannot make a sense in these dialogues. Our conclusion for dialogue 3 and 4 is that the majority of subjects can recognize the difference among sounds of sentences with the different focus parts and select the correct sounds that make the most senses in the dialogue.

9.2.3 Conclusion of Perceptual Evaluation

Our perceptual evaluations consist of two sections: comparison of the sounds with and without focus and comparison of the sounds among the different focus parts. In the first section, the difference between dialogues 1 and 2 is statistically significant based on the ANOVA method calculated from the number of subject's answers. We observe that the number of subject's answers of the sounds with focus in dialogue 1 is distinguishably higher than the number of subject's answers in dialogue 2. The dialogue 2 has negative feedback from participants and affects our experimental results.

Table 9.4: Number of selecting choices A, B, C, and D of each subject for dialogue 3 and 4

Subject No.	No. of Choices			
	A	B	C	D
1	0	0	4	1
2	1	0	4	0
3	0	0	0	5
4	2	0	2	1
5	2	1	1	1
6	0	0	4	1
7	3	0	1	1
8	4	0	0	1
9	4	0	1	0
10	3	0	2	0
11	3	1	1	0
12	5	0	0	0
13	3	0	1	1
14	2	2	1	0
15	5	0	0	0
16	4	0	1	0
17	3	1	1	0
Total	44	5	24	12

This dialogue needs to be considered separately from dialogue 1. From the results in dialogue 1, the subjects recognize focus in the sentence and select the sound with focus for their most preference. On the other hand, the results of the dialogue 2 suggest that the sounds with focus are not the subject's preferences because of the low quality and inconsistency of prosodic modification, which is a cause of fault utterances of some sounds.

In the second section, there is a statistically significant difference among different focus parts for dialogues 3 and 4. Most subjects recognize the different focus parts in the same sentence. They can select the focus part which makes the most sense in the sentence correctly. Therefore, the prosody can convey focus content to the listener.

9.3 Evaluation of Prosodic Annotation

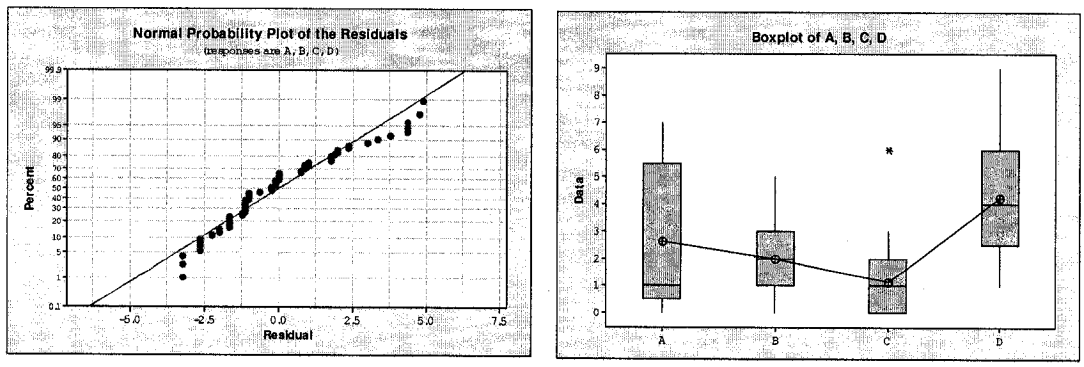
Finding the accuracy of prosodic annotation is a difficult task. One of the main obstacles is building the gold standard of prosodic annotation corpus. There is no

```
One-way ANOVA: Res versus ch
ch:  choice of A, B, C, and D
Res:  No.of answer choice A, B, C, and D

Source  DF      SS     MS     F      P
ch       3   88.29  29.43  6.66  0.001
Error   64  282.71   4.42
Total   67  371.00
S = 2.102   R-Sq = 23.80%   R-Sq(adj) = 20.23%

Individual 95% CIs For Mean Based on Pooled StDev
Level   N   Mean  StDev  ---------+---------+---------+---------+
A      17  2.647  2.783           (------*-----)
B      17  2.000  1.414        (-----*------)
C      17  1.118  1.576  (-----*------)
D      17  4.235  2.333                     (------*------)
                          ---------+---------+---------+---------+
                               1.5       3.0       4.5       6.0
Pooled StDev = 2.102

Tukey 95% Simultaneous Confidence Intervals
All Pairwise Comparisons among Levels of ch
Individual confidence level = 98.95%

ch = A subtracted from:
ch Lower Center  Upper  -------+---------+---------+---------+--
B   -2.548 -0.647  1.254          (-----*-----)
C   -3.431 -1.529  0.372        (-----*-----)
D   -0.313  1.588  3.490                  (-----*------)
                         -------+---------+---------+---------+--
                             -3.0       0.0       3.0       6.0
ch = B subtracted from:
ch   Lower  Center  Upper  -------+---------+---------+---------+--
C   -2.784 -0.882  1.019          (-----*-----)
D    0.334  2.235  4.137                   (-----*------)
                          -------+---------+---------+---------+--
                              -3.0       0.0       3.0       6.0
ch = C subtracted from:
ch  Lower  Center  Upper  -------+---------+---------+---------+--
D   1.216  3.118  5.019                    (-----*------)
                          -------+---------+---------+---------+--
                              -3.0       0.0       3.0       6.0
```

Figure 9.11: ANOVA table, individual 95% CIs, and Turkey test for the comparison among choices in dialogue 3

gold standard for the prosodic annotation. Normally, the researchers compare the results from their annotation system with the annotation performed by linguists or experts. However, different linguists or experts may annotate different prosodic marks at the same position in the sentence. They annotate the prosodic marks depending on what they heard and sometimes they may not hear the same prosody or may hear partially different prosodies which also affects the prosodic annotation.

In this evaluation, the annotation by the FET system is compared with the annotation dataset from the CMU Communicator (CMU-COM), which is the reference dataset to measure performance. The sentences in this dataset are derived from the



(a) (b)

Figure 9.12: Graphs of (a) normal plot (b) box plot for dialogue 3

travel reservation dialogues. One hundred sentences with their ToBI annotations are collected for the evaluation. Sixty-one sentences of one hundred sentences can be parsed by the LKB system using the ERG grammar that provides the MRS as the results. The MRS representations are transformed to a set of focus words for each sentence and then these 61 sentences are parsed by the LKB with the FET subgrammar. The results are the sentences with ToBI annotation on each word.

9.3.1 CMU Communicator KAL Limited Domain

The CMU-COM dataset uses a limited domain synthesis for the dialogue system. The corpus is gathered from an automated telephone based dialogue system for booking flight information, called the Communicator. This corpus consists of five hundred sentences including the ToBI marks, part-of-speech, time features and frequency features for each syllable. Five hundred sentences are selected based on the most frequent utterances in the telephone-based dialogue system gathered over a 3 month period. Some example sentences from this dataset are shown in table 9.5.

Table 9.5: Example sentences from the CMU-COM dataset

No.	Sentence
1	Where are you leaving from?
2	Are you a registered user?
3	This is the end of the instructions
4	I didn't catch that.
5	Are you satisfied with this itinerary?

9.3.2 Design of Experiment for Prosodic Annotation

For each word, if the annotation by the FET system matches the annotation from the CMU-COM, then the number of matched annotation labels will be counted. The words without prosodic marks are labeled with "0" while the words with prosodic marks are labeled with ToBI representations. The comparison between two prosodic annotations is shown in Appendix D. Because of no gold standard for the prosodic annotation, this comparison is used to measure our annotation by the FET system with the reference dataset, which is the CMU-COM dataset.

There are two main sections of this evaluations: (i) the evaluation without tone mark's alignment and (ii) the evaluation with the tone mark's alignment. For the first section, the evaluation is processed by matching the same annotation word by word in each sentence and counting the number of matches.

For the second section, the evaluation with the tone marks's alignment is processed by aligning the tone annotations by the FET system before comparing with the annotation from the CMU-COM dataset. An example of alignment is shown in table 9.6. The example sentence is "Are you a registered user?". If the accent tone marks of the FET system are shifted right at word "a" and "registered", the annotation after shifting is shown in the row "FET after shift". Comparing two annotations, the annotation by the FET system after shift is the same as the annotation from the CMU-COM.

The reason for using tone mark's alignment is to match the tone marks between two systems. Although the tone annotations by the FET system looks different from the annotation of the CMU-COM, by only shifting the accent tone marks to the previous word, both annotations govern the same pitch contour and same boundary and the sounds after modifying prosody are not different. For example, the pitch contour of the annotation of {a/0, registered/H*, user/L+H* H-H} from the CMU-COM is similar to the pitch contour of the annotation of {a/H*, registered/L+H*, user/H-H%} and the boundary of pitch contour governs from "a" to "user" even though there are no annotations at "a" for the CMU-COM. After the tone mark alignments, the number of words, whose annotations are matched by comparing between the FET system and the CMU-COM, are counted and the counts are divided by the total number of words.

Table 9.6: Comparison of tone annotations of the FET system and CMU-COM dataset

Sentence	Are	you	a	registered	user
FET	0	H*	H* →	L+H* →	H-H%
FET after shift	0	H*	0	H*	L+H* H-H%
CMU Communicator	0	H*	0	H*	L+H* H-H%

9.3.3 Evaluation Results of Prosodic Annotation

For the first section, the result is calculated by dividing the number of words with matching annotations by the total number of words. The total number of words is 455 words of the 61 sentences. The number of matched annotations are 263 words and the number of unmatched annotations is 192 words. Summary of this evaluation is shown in table 9.7. The percentage of matched tone marked is 57.8%, comparing between the tone annotation from the CMU-COM dataset and the annotation by the FET system. When comparing tone annotations, the utterances of H* and L+H* may not be distinguished by hearer. This conclusion is discussed in section 9.3.4. For example, in table 9.6, "registered" can be marked with L+H* instead of H* in the row "FET". If one of them can match with the annotation of "registered" from the CMU-COM dataset, then their annotations are matched.

Table 9.7: Summary of the evaluation without tone mark's alignment

	Sum of Words	Percentage
Matched tone marks	263	57.802
Unmatched tone marks	192	42.198
Total No. of words	455	

The second section of evaluation is the comparison between tone annotations by the FET system with tone alignments, and from the CMU-COM dataset. The total number of words is 455 words of 61 sentences. The number of matched annotations is 342 words while unmatched annotations are 113 words. The percentage of matched annotation is equal to 75.164 percents. The summary of this section is shown in table 9.8.

Without the gold standard of the prosodic annotation, we cannot find the true accuracy of the annotation by the FET system. We compare the annotation of our

Table 9.8: Summary of the evaluation with tone mark's alignment

	Sum of Words	Accuracy(%)
Matched tone marks	342	75.165
Unmatched tone marks	113	24.835
Total No. of words	455	

system with CMU-COM, which is our reference annotation dataset, generated by a machine learning system. The experimental result is the similarity measurement between the FET system and the CMU-COM dataset.

Considering the unmatched results in table 9.8, approximately 25% of tone mark comparison is not matched. Based on our observation, most of the unmatched labels occur in a group of prepositions (PP), pronouns (PR), and determiners (DET). We discuss why two systems label different tone marks for the annotation in this group. Many DET, PR, and PP in the CMU-COM are not marked with any annotation. The FET system has more sensitive tone annotation on the PP, PR, and DET than the CMU-COM dataset. For example, the tone annotations of the sentence "Where are you leaving from" are shown in table 9.9. A difference is at the word "you". The annotation by FET system has tone mark H* while there is no tone mark for the CMU-COM dataset. Since the FET system considers "you", that needs to be focus of actee part, "you" is marked with the emphasized tones H* or L+H*. On the other hand, "you" is not considered as the theme of sentence for the CMU-COM dataset so this word is no tone mark.

Table 9.9: Comparison the annotations of sentence "Where are you leaving from" between the FET system and the CMU-COM dataset

Sentence	Where	are	you	leaving	from
FET	0	0	L+H*/H*	L+H*/H*	L-L%
CMU	0	0	0	H*	L-L%

9.3.4 Discussion between H* and L+H*

In the FET system, the H* and L+H* are not distinguished. The reason is that the pitch contour of H* looks like the pitch contour of L+H*. The pitch contours of H*

is slightly different from the pitch contour of L+H*. The main difference between H* and L+H* is that the pitch contour of L+H* have a delayed rising and an extended peak. The difference of pitch between H* and L+H* may not be distinguished by most people's hearing perception, except by an expert. There are different opinions about L+H*; i.e. Steedman [80] defined L+H* accent which is marked for a theme part in the sentence, Ladd [67] explains that the status of L+H* is a phonologically distinct entity.

In the FET system, the L+H* accent is defined as the high emphasis, which is slightly more tone emphasis than the H* accent. Both L+H* and H* are also defined for a focus part in a sentence. Another accent, L*+H which appears rarely in the dataset, have a similar pitch contour to H* and L+H* but this accent is not included to the prosodic structure in the FET system. The reason is that the L*+H has currently insufficient evidence to distinguish L*+H from L+H* and H*. The researchers [67, 80, 5] agree that the pitch contours of H*, L+H* and L*+H are almost the same and are not easily distinguishable.

9.4 Summary

Following the experimental results, our FET system can annotate the ToBI marks with 75% of similarity when compared with the CMU-Com dataset. The perceptual evaluation is performed by the listening tests and their results support our assumption that the prosody can convey focus content and the listeners can recognize these focuses emphasized by prosody.

Chapter 10

Conclusion

We proposed the FET structure to represent the relationship between focus and prosody of our analysis. The summary of the FET structure and its environment is given in section 10.1. Our future work is introduced in section 10.2.

10.1 Conclusions

The contributions of our research consist of three main parts: analyzing the relationships of focus and prosody, designing focus to emphasize tone system and evaluating the performance of the FET system. In the first part, we have investigated the relationships of focus and prosody. Our analysis includes focus analysis and analysis of the speaker's intention and intonation patterns. Focus analysis is used to define speaker's intention and focus components; i.e. focus parts and focus types. In chapter 4, we analyzed the syntactic and semantic features using the LKB system with ERG. The LKB system is a unification-based parser and the result of parsing is the MRS representation. We used this representation to define focus parts (actor, act, and actee) for a sentence. For the focus type, each focus part can be labeled as wide focus or single focus. The focus type assignment is controlled by the speaker's intention and sentence types, and it must follow our focus constraints. The details of focus analysis are presented in chapter 5. For the relationships of speaker's intention and intonation patterns, we studied the prosodic phenomena and the intonation patterns based on the linguistic theory as explained in chapter 6. Furthermore, we collected the intonation patterns from the CMU-COM dataset. These patterns are annotated with the ToBI representation. We filtered these patterns, based on the speaker's intention and sentence types, to design our prosodic structures. The CMU-COM dataset is gathered from the travel reservation domain and we focus on three types of speaker's intentions in our analysis.

The FET structure is a unification-based formalism for focus analysis. Designing the FET structures for LKB, described in chapter 7, is based on analysis of relationships between focus and prosody. This structure contains the focus and prosodic

feature structures, and the structures of relationships mapping between these features. The focus structure includes the focus part, focus type, and sentence type of a sentence. The prosodic feature structure is used to represent prosodic phenomena. We also design the environment for the FET system called the FET subgrammar, including a set of FET constraints, typed hierarchy, focus words, focus rules, and so on. The implementation of the FET system is demonstrated by an example, which is shown in chapter 8. This implementation begins with parsing the sentence using LKB with the ERG which represents the preprocessing step. In the next step, the LKB with the FET subgrammar parses the sentence with a set of focus words. As a result, the FET structure is generated and it includes the focus information, prosodic information, and ToBI marks of the sentence. In the last step, the Festival speech synthesis generates the synthetic speech for this sentence and we modified the prosody following the ToBI marks using the Praat system with our prosodic modification module.

The last part of our work is (i) the perceptual evaluation and (ii) the evaluation for the prosodic annotation described in chapter 9. We performed the listening test to evaluate whether a listener can recognize focus content by emphasizing tone in a sentence. Four dialogues in travel reservation domain are prepared for this listening test. From the experimental results, the listener can recognize the focus parts among the different tone emphasis from three of four dialogues. For the evaluation for the prosodic annotation, since there is no gold standard to measure performance of prosodic annotation systems, it would be difficult to build one. We compare our results with the tone annotation of the CMU-COM dataset The comparison results show approximately 75% of similarity between these systems.

Below is the summary of our contributions

- **Building the** FET **Subgrammar** In this system, we have designed the FET structure to represent relationships of focus and prosodic domains. The FET structure is unique and is used for analyzing this relationship between domains. The design of FET structure is compatible with the unification-based formalism for the LKB system. The FET structure includes focus structures such as focus parts and focus types, speaker's intention feature and the prosodic feature structures.

- **Reducing the ambiguity and improving the recognition rate in speaker utterance.** The critical content of modified prosody following the prosodic annotation by the FET system, is easy to recognize in a sentence because the focus part in the sentence is emphasized. Understanding these contents by hearing their utterances also increases because of the tone emphasis at the focus content. It is very useful for increasing the recognition rate in communication with a low sound quality, such as the telephone quality.

- **Performing the perceptual evaluation and evaluation of prosodic annotation.** The listening test has been performed in our evaluation. The objective of the evaluation is whether listeners can recognize the different foci in a sentence by emphasizing prosody at the focus part. Our experimental results show that most listeners can recognize the focus part in the sentence. They can select the utterance that makes the most sense in the dialogue correctly from three of four dialogues. For the prosodic annotation, we compared the annotation of our system with the annotation from the CMU-COM. The results of our system is 75% of similarity of the annotation from the CMU-COM.

10.2 Discussion and Future Works

We have investigated the concept of focus in the speech utterance especially in relationships of focus, speaker's intention, and prosody. We have implemented and evaluated the FET system as the proof-of-concept system for a number of sentences in the travel reservation domain. Our approach is limited by our use of LKB with ERG and as such demonstrates proof-of-concept. The parser's performance depends on the number of grammar rules and lexicon entries of the ERG. The FET system as the unification-based sub–grammar now works separately from the LKB system with ERG. In this thesis, we designed the FET subgrammar to support a set of sentences controlled by sentence types and speech act types. Our experiment is a domain–dependent implementation such that we control the number of focus, speaker's intention and prosodic features. The experimental results has shown a statistical significance at 95% interval confidence of conveying focus by emphasizing prosody.

For our future work, we plan to investigate the following list.

- **Extending the FET structure.** The analysis of relationships of focus, speaker's intention and the prosodic features need to have more investigation to support various sentences. The large number of sentences annotated with prosodic marks must be gathered to recognize various intonation patterns depending on speech act groups, and sentence types.

- **Integrating the FET structure as a sub–grammar of the ERG.** This is a complicated process to include a new feature structure into the ERG and we need to insure that the FET subgrammar will not decrease the performance of the LKB parser with ERG. This integration will increase the knowledge-based structures for the constraint-based parsing system. The benefit of this integration is that the LKB parser can analyze not only syntactic and semantic features but also focus, speaker's intention and prosodic features.

- **Extend perceptual evaluation.** The evaluation will be performed to collect feedback from participants. Their feedback will help us to improve the performance and reduce some errors in prosodic generation for the FET system. For instance, based on the participant's comments in chapter 9, we plan to reduce the inconsistency in prosodic generation which is a cause of fault utterance. This inconsistency occurs because of (i) prosodic annotation and (ii) the prosodic modification algorithm and the investigation of this inconsistency will be in our future work.

Finally, our FET system is constructed as a small parser by using the LKB system to annotate the prosodic marks based on focus and speaker's intention information. For our design, the FET subgrammar is the core structure and it contains the FET structures, grammar rules, focus words and so on. This subgrammar can be used to analyze the relationships of focus and speaker's intention to find the intonation patterns. Our evaluation shows that there is statistical significance in conveying focus of prosody by emphasizing tone to hearer.

Bibliography

[1] M. Haji-Abdolhosseini. A constraint-based approach to information structure and prosody correspondence. In Stefan Mller, editor, *Proceedings of The HPSG-2003 Conference*, pages 143–162. CSLI Publisher, 2003.

[2] E. Klien. Prosodic constituency in hpsg. In R. Cann, C. Grover, and P. Miller, editors, *Grammatical Interfaces in HPSG*, pages 169–200. CSLI Publisher, 2000.

[3] Ann Copestake. *Implementing Typed Feature Structure Grammars.* CSLI Publisher, Stanford, CA, 2002.

[4] Ann Copestake, D. Flickinger, R.Malouf, S. Riehemann, and I.A. Sag. Translation using minimal recursion semantics. *In Proceeding of the The 6th International Conference on Theoretical and Methodological Issues in Machine Translation (TMI-95)*, 1995.

[5] K. Silverman, M. E. Beckman, J. Pitrelli, M. Ostendorf, C. Wightman, P. Price, J. Pierrehumbert, and J. Hirschberg. ToBI: a standard for labelling English prosody. In *Proceedings of the 1992 International Conference on Spoken Language Processing(ICSLP'92)*, volume 2, pages 867–870, Banff, Canada, October 1992.

[6] Paul Boersma and David Weenink. *Praat: Doing phonetics by computer (Version 4.4.32).* Institute of Phonetic Sciences, University of Amsterdam, September 2006.

[7] Lalita Narupiyakul, Vlado Keselj, Nick Cercone, and Booncharoen Sirinaovakul. Focus to emphasize tone analysis for prosodic generation. *An International Journal: Computers & Mathematics with Applications*, 2007. (To appear).

[8] Lalita Narupiyakul, Nick Cercone, Vlado Keselj, and Booncharoen Sirinaovakul. Focus and speech act in prosodic analysis for spoken language generation. In *The 6th Symposium on Natural Language Processing, SNLP'2005*, Chiang Rai, Thailand, December 2005.

[9] Lalita Narupiyakul. Focus to emphasize tone structures for prosodic analysis in spoken language generation. In *ACL*, Sydney, Australia, July 2006. The Association for Computer Linguistics.

[10] Hiroya Fujisaki. Prosody, models, and spontaneous speech. In N. Campbell Y. Sagisaka and N. Higuchi, editors, *Computing Prosody*, New York, USA, 1997. Springer.

[11] J.L.G. Baart. *Focus Syntax and Accent Placement.* PhD thesis, University of Leiden, 1987.

[12] J.R. Davis and J. Hirschberg. Assigning intonational features in synthesized spoken directions. In *Proceedings of the 26th Annual Meeting of the Association for Computational Linguistics (ACL88)*, pages 187–193, Buffalo, USA, 1988.

[13] D. Klatt. Software for a cascade/parallel formant synthesizer. *Journal of the Acoustical Society of America*, 67:971–995, 1987.

[14] E. Moulines and F. Charpentier. Pitch-synchronous waveform processing techniques for text-to-speech synthesis using diphones. *Speech Communication*, 9(5): 453–467, 1990.

[15] F. Charpentier and MG. Stella. Diphone synthesis using an Overlap-Add technique for speech waveforms concatenation. In *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing*, pages 2015–2018, 1986.

[16] A. Hunt and A. Black. Unit selection in a concatenative speech synthesis system using a large speech database. In *Proceedings of ICASSP 96*, volume 1, pages 373–376, Atlanta, Georgia, USA, 1996.

[17] L. Breiman. *Classification and Regression Trees.* Wadsworth, Pacific Grove, California, 1984.

[18] Thierry Dutoit. *An Introduction to Text-to-Speech Synthesis.* Kluwer Academic Publishers, Norwell, MA, USA, 2001. ISBN 1402003692.

[19] John F. Pitrelli, R. Bakis, E. M. Eide, R. Fernandez, W. Hamza, and M. A. Picheny. The ibm expressive text-to-speech synthesis system for american english. *IEEE Transactions on Audio, Speech & Language Processing*, 14(4):1099–1108, 2006.

[20] H.H. Clark and S.E. Haviland. Comprehension and the given-new contract. In R.O. Freedle, editor, *Discourse Processes: Advances in Research and Theory*, volume 1, page 140. Ablex, 1977.

[21] E.F. Prince. Toward a taxonomy of given-new information. In P. Cole, editor, *Radical Pragmatics*, pages 223–255. Academic Press, 1981.

[22] B.J. Grosz and C.L. Sidner. Attention, intentions, and the structure of discourse. *Computational Linguistics*, 12(3):175–204, 1986.

[23] A. Dirksen. Accenting and deaccenting: A declarative approach. In *Proceedings of 14th International Conference on Computational Linguistics (COLING'92)*, pages 865–869, Nantes, France, July 1992.

[24] J.J. Venditti. *Discourse Structure and Attentional Salience Effects on Japanese Intonation.* PhD thesis, Ohio State University, 2000.

[25] Hiroya Fujisaki and H. Sudo. A generative model of the prosody of connected speech in Japanese. *Annual Report of Engineering Research Institute*, 30:75–80, 1971.

[26] K.R. McKeown. Discourse strategies for generating natural language text. *Artificial Intelligence*, 27(1):1–42, 1985.

[27] K. McKeown and S. Pan. Prosody modeling in concept-to-speech generation: Methodological issues. *Philosophical Transactions of The Royal Soceity, Series A: Mathematical, Physical and Engineering Sciences*, 358(1769):1225–1431, 1999.

[28] E. Klabbers, J. Odijk, J. R. de Pijper, and M. Theune. Goalgetter: From teletext to speech. In *IPO Annual Progress Report 31*, pages 66–75, Eindhoven, 1996.

[29] E. Andr, G. Herzog, and T. Rist. On the simultaneous interpretation of real world image sequences and their natural language description: The system SOCCER. In Y. Kodratoff, editor, *Proceedings of the 8th European Conference on Artificial Intelligence (ECAI'88)*, pages 449–454, London, 1988. Pitmann Publisher.

[30] K. Tanaka, K. Hasida, and I. Noda. Reactive content selection in the generation of real-time soccer comentary. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th iIternational Conference on Computational Linguistics (COLING-ACL'98)*, pages 1282–1288, Montreal Canada, 1998.

[31] V. Zue, S. Seneff, J.R. Glass, J. Polifroni, C. Pao, T.J. Hazen, and L. Hetherington. Jupiter: A telephone-based conversational interface for weather information. In *IEEE Trans Acoustics, Speech and Signal Processing*, volume 8 of *1*, pages 85–96, January 2000.

[32] S. McGlashan, D.C. Burnett, J. Carter, P. Danielsen, J. Ferrans, A. Hunt, B. Lucas, B. Porter, K. Rehor, and S. Tryphonas. Voice extensible markup language (VoiceXML) version 2.0. Technical report, W3C Candidate Recommendation, March 2004.

[33] J.E. Cahn. The generation of affect in synthesized speech. *Journal of the American Voice I/O Society*, 8(1):1–9, July 1990.

[34] Corporate Author International Phonetic Association. *Handbook of the International Phonetic Association: A guide to the use of the International Phonetic Alphabet.* Cambridge University Press, July 1999. ISBN 0521637511.

[35] D.C. Burnett, M.R. Walker, and A. Hunt. Speech synthesis markup language version 1.0. Technical report, W3C Candidate Recommendation, December 2003.

[36] A. Hunt. Japanese speech markup language. Technical report, World Wide Web Consortium, June 2003.

[37] M. Beckman and J. Hirschberg. The ToBI annotation conventions. Unpublished manuscript, 1993.

[38] J. Terken and J. Hirschberg. Deaccentuation of words representing given information: Effects of persistence of grammatical function and surface position. *Language and Speech*, 37(2):125145, 1994.

[39] C. Nakatani and J. Chu-Carroll. Using dialogue representations for concept-to-speech generation. In *Proceedings of the ANLP-NAACL Workshop on Conversational Systems*, Seattle, USA, 2000.

[40] C. Raymond Perrault and James Allen. A plan-based analysis of indirect speech acts. *American Journal of Computational Linguistics*, 6:167182., 1980.

[41] Klaus von Heusinger. *Intonation and Information Structure: The Representation of Focus in Phonology and Semantics. H.* Universitt Konstanz., 1999.

[42] John Langshaw Austin. Performative utterances. In J. O. Urmson and G. J. Warnock, editors, *Philosophical Papers*, 1961.

[43] Philip R. Cohen and C. Raymond Perrault. Elements of a plan-based theory of speech acts. *Cognitive Science 3*, 3(3):177–212, 1979.

[44] James F. Allen and C. Raymond Perrault. Plans, inference, and indirect speech acts. In *17th Annual Meeting of the Association for Computational Linguistics*, 1979.

[45] Ann Copestake and D. Flickinger. An open-source grammar development environment and broad-coverage English grammar using HPSG. *In Proceedings of the Second Conference on Language Resources and Evaluation (LREC-2000)*, 1997.

[46] M. Theune. Contrast in concept-to-speech generation. *Computer Speech & Language*, 16:491–531, July-October 2002.

[47] D.K. Roy. Learning visually grounded words and syntax for a scene description task. *Computer Speech and Language*, 16:353–385, 2002.

[48] P. Taylor. Concept-to-speech synthesis by phonological structure matching. *Philosophical Transactions of The Royal Soceity, Series A: Mathematical, Physical and Engineering Sciences*, 358(1769):1225–1431, 1999.

[49] S. Pan, K. McKeown, and J. Hirschberg. Exploring features from natural language generation for prosody modeling. *Computer Speech & Language*, 16:457–490, July-October 2002.

[50] W. Cohen. Fast effective rule induction. In *Machine Learning: Proceedings of the Twelfth International Conference (ML'95)*, 1995.

[51] M.A. Walker and O.C. Rambow. Spoken language generation. *Computer Speech & Language*, 16:273–281, July–October 2002.

[52] R.E. Schapire. A brief introduction to boosting. In *Proceedings of the Sixteenth International Joint Conference on Artificial Intelligence*, 1999.

[53] A.H. Oh and A.I. Rudnicky. Stochastic natural language generation for spoken dialog systems. *Computer Speech & Language*, 16:387–407, July-October 2002.

[54] Ash Asudeh and Ewan Klein. Shape conditions and phonological context. In Frank van Eynde, Lars Hellan, and Dorothee Beermann, editors, *The 8th. International HPSG Conference*, Stanford, CA, 2002. CSLI Publications.

[55] Carnegie Mellon University Language Technologies Institute. CMU communicator KAL limited domain., October 2005. URL `http://www.festvox.org`.

[56] I. Bulyko and M. Ostendorf. Efficient integrated response generation from multiple targets using weighted finite state transducers. *Computer Speech & Language*, 16:533–550, July-October 2002.

[57] I.A. Sag C. Pollard. *Head-driven Phrase Structure Grammar.* Studies in Contemporary Linguistics. University of Chicago Press, Chicago, 1993.

[58] T. Wasow I.A. Sag. *Syntactic Theory: A Formal Introduction.* CSLI Publications, 1999.

[59] N. Campbell. Specifying affect and emotion for expressive speech synthesis. In A. Gelbukh, editor, *Computational Linguistics and Intelligent Text Processing, Proc. CICLing-2004. Lecture Notes in Computer Science*, Korea, Februry 2004. Springer-Verlag.

[60] Kevin Knight. Unification: A multidisciplinary survey. *ACM Comput. Surv.*, 21 (1):93–124, 1989.

[61] J. A. Robinson. A machine-oriented logic based on the resolution principle. *Journal of ACM*, 12(1):23–41, 1965.

[62] Mike Paterson and Mark N. Wegman. Linear unification. In *STOC*, pages 181–186, 1976.

[63] Alberto Martelli and Ugo Montanari. An efficient unification algorithm. *ACM Trans. Program. Lang. Syst.*, 4(2):258–282, 1982.

[64] M. Steedman. Information-structural semantics for English intonation, 2001.

[65] Mark Steedman. Information and syntax in spoken language systems. In *Proceedings of the workshop on Speech and Natural Language (HLT '89)*, pages 222–227, Morristown, NJ, USA, 1989. Association for Computational Linguistics.

[66] Elisabeth Selkirk. On prosodic structure and its relation to syntactic structure. In T. Fretheim, editor, *Nordic Prosody II*, pages 111–140, Trondheim: Tapir, 1981.

[67] Robert Ladd. *The Structure of Intonational Meaning. Evidence from English.* Indiana University Press, 1980.

[68] Anand S. Rao and Michael P. Georgeff. Decision procedures for bdi logics. *Journal Logic and Computation*, 8(3):293–342, 1998.

[69] Wallace Chafe. *Givenness, Contrastiveness, Definiteness, Subjects, Topics, and Point of View.* Subject and Topic. New York: Academic Press, 1976.

[70] Michael Halliday. Notes on transitivity and theme in English Part 1 and 2. *Journal of Linguistics*, 3:199–244, 1967.

[71] T. T. Ballmer and W. Brennenstuhl. *Speech act classification: A study of the lexical analysis of English speech activity verbs*, volume 8 of *Series in language and communication.* Springer, Berlin, 1981.

[72] Mary E. Beckman and Gayle Ayers Elam. Guidelines for ToBI labelling. Ohio State University, March 1997.

[73] Sphinx III. Sphinx Speech Group, School of Computer Science, Carnegie Mellon University, 2006. URL `http://cmusphinx.sourceforge.net/`.

[74] Carlos Gussenhoven. *The Phonology of Tone and Intonation (Research Surveys in Linguistics).* Cambridge University Press, July 2004. ISBN 0521012007.

[75] A.W. Black, P. Taylor, and R. Caley. *Festival speech synthesis system.* Centre for Speech Technology Research, University of Edinburgh, UK, 1.95 edition, July 2004.

[76] J. Kowtko and P. Price. *SRI Transcripts derived from audiotape conversations made at SRI Inter.* SRI Inter., Menlo Park, CA., 1994.

[77] Y. Morlec, V. Auberg, and G. Bailly. Evaluation of automatic generation of prosody with a superposition model, 1995.

[78] Robert A. J. Clark. *Generating Synthetic Pitch Contours Using Prosodic Structure.* PhD thesis, The University of Edinburgh, 2003.

[79] D.C. Montgomery. *Design and Analysis of Experiments.* John Wiley, NY US, 5 edition, June 2000.

[80] M. Steedman. Information-structural semantics for English intonation. In *Proceedings of LSA Summer Institute Workshop on Topic and Focus*, pages 294–301, Santa Barbara, USA, 2003.

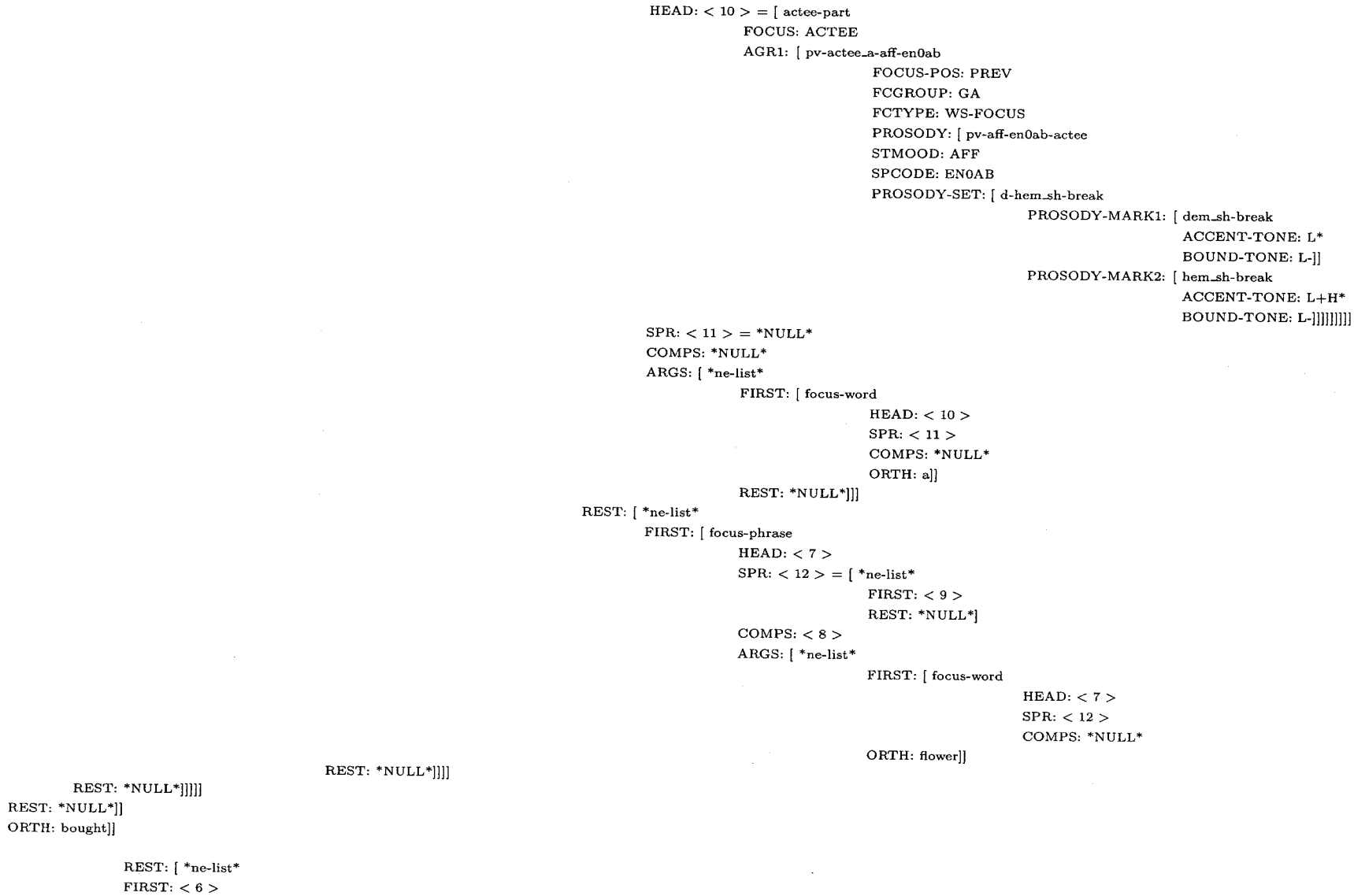
Appendix A

FET Sturucture in the LKB System

A.1 FET Structure of "Mary bought a flower"

```
[ focus-phrase
 HEAD: < 0 > = [ act-part
                 FOCUS: ACT
                 AGR1: [ ls-act_a-aff-en0ab
                         FOCUS-POS: LAST
                         FCGROUP: GA
                         FCTYPE: SG-FOCUS
                         PROSODY: [ aff-en0ab-no-prosody
                                    STMOOD: AFF
                                    SPCODE: EN0AB
                                    PROSODY-SET: [ no-prosody
                                                   PROSODY-MARK1: [ no-mark
                                                                    ACCENT-TONE: NOACCENT
                                                                    BOUND-TONE: NOBOUND]]
                                                   PROSODY-MARK2: [ no-mark
                                                                    ACCENT-TONE: NOACCENT
                                                                    BOUND-TONE: NOBOUND]]]]]]]]
 SPR: *NULL*
 COMPS: < 1 > = *NULL*
 ARGS: [ *ne-list*
                 FIRST: < 2 > = [ focus-phrase
                                  HEAD: ¡3¿ = [ actor-part
                                                FOCUS: ACTOR
                                                AGR1: [ ls-actor_a-aff-en0ab
                                                        FOCUS-POS: LAST
                                                        FCGROUP: GA
                                                        FCTYPE: WS-FOCUS
                                                        PROSODY: [ aff-en0ab-no-prosody
                                                                   STMOOD: AFF
                                                                   SPCODE: EN0AB
                                                                   PROSODY-SET: [ no-prosody
                                                                                  PROSODY-MARK1: [ no-mark
                                                                                                   ACCENT-TONE: NOACCENT
                                                                                                   BOUND-TONE: NOBOUND]]
                                                                                  PROSODY-MARK2: [ no-mark
```

```
                                                            ACCENT-TONE: NOACCENT
                                                            BOUND-TONE: NOBOUND]]]]]]]]]

SPR: < 4 >= *NULL*
COMPS: *NULL*
ARGS: [ *ne-list*
      FIRST: [ focus-word
             HEAD: < 3 >
             SPR: < 4 >
             COMPS: *NULL*
             ORTH: Mary]]
      REST: *NULL*]]]
REST: [ *ne-list*
      FIRST: [ focus-phrase
             HEAD: < 0 >
             SPR: < 5 > = [ *ne-list*
                          FIRST: < 2 >
                          REST: *NULL*]
             COMPS: < 1 >
             ARGS: [ *ne-list*
                   FIRST: [ focus-word
                          HEAD: < 0 >
                          SPR: < 5 >
                          COMPS: [ *ne-list*
                                 FIRST: < 6 > = [ focus-phrase
                                                HEAD: < 7 > = [ actee-part
                                                              FOCUS: ACTEE
                                                              AGR1: [ ls-actee_a-aff-en0ab
                                                                     FOCUS-POS: LAST
                                                                     FCGROUP: GA
                                                                     FCTYPE: WS-FOCUS
                                                                     PROSODY: [ ls-aff-en0ab-actee
                                                                              STMOOD: AFF
                                                                              SPCODE: EN0AB
                                                                              PROSODY-SET: [ n-hem_lg-break
                                                                                           PROSODY-MARK1: [ hem_lg-break
                                                                                                          ACCENT-TONE: L+H*
                                                                                                          BOUND-TONE: L-L%]]
                                                                                           PROSODY-MARK2: [ no-mark
                                                                                                          ACCENT-TONE: NOACCENT
                                                                                                          BOUND-TONE: NOBOUND]]]]]]]]]
                                                SPR: *NULL*
                                                COMPS: < 8 > = *NULL*
                                                ARGS: [ *ne-list*
                                                      FIRST: < 9 > = [ focus-phrase
```


HEAD: < 10 > = [actee-part
FOCUS: ACTEE
AGR1: [pv-actee_a-aff-en0ab
FOCUS-POS: PREV
FCGROUP: GA
FCTYPE: WS-FOCUS
PROSODY: [pv-aff-en0ab-actee
STMOOD: AFF
SPCODE: EN0AB
PROSODY-SET: [d-hem_sh-break
PROSODY-MARK1: [dem_sh-break
ACCENT-TONE: L*
BOUND-TONE: L-]]
PROSODY-MARK2: [hem_sh-break
ACCENT-TONE: L+H*
BOUND-TONE: L-]]]]]]]]
SPR: < 11 > = *NULL*
COMPS: *NULL*
ARGS: [*ne-list*
FIRST: [focus-word
HEAD: < 10 >
SPR: < 11 >
COMPS: *NULL*
ORTH: a]]
REST: *NULL*]]]
REST: [*ne-list*
FIRST: [focus-phrase
HEAD: < 7 >
SPR: < 12 > = [*ne-list*
FIRST: < 9 >
REST: *NULL*]
COMPS: < 8 >
ARGS: [*ne-list*
FIRST: [focus-word
HEAD: < 7 >
SPR: < 12 >
COMPS: *NULL*
ORTH: flower]]
REST: *NULL*]]]]
REST: *NULL*]]]]]
REST: *NULL*]]
ORTH: bought]]
REST: [*ne-list*
FIRST: < 6 >

```
REST: *NULL*]]]]]]
REST: *NULL*]]]]
```

A.2 FET Structure of "Kim bought a flower for her mother"

```
[ focus-phrase
 HEAD: < 0 > = [ act-part
                 FOCUS: ACT
                 AGR1: [ ls-act_a-aff-en0ab
                         FOCUS-POS: LAST
                         FCGROUP: GA
                         FCTYPE: SG-FOCUS
                         PROSODY: [ aff-en0ab-no-prosody
                                    STMOOD: AFF
                                    SPCODE: EN0AB
                                    PROSODY-SET: [ no-prosody
                                                   PROSODY-MARK1: [ no-mark
                                                                    ACCENT-TONE: NOACCENT
                                                                    BOUND-TONE: NOBOUND ] ]
                                                   PROSODY-MARK2: [ no-mark
                                                                    ACCENT-TONE: NOACCENT
                                                                    BOUND-TONE: NOBOUND ] ] ] ] ] ] ] ] ]


  SPR: *NULL*
  COMPS: < 1 > = = *NULL*
  ARGS: [ *ne-list*
          FIRST: < 2 > = [ focus-phrase
                           HEAD: < 3 > = [ actor-part
                                           FOCUS: ACTOR
                                           AGR1: [ ls-actor_a-aff-en0ab
                                                   FOCUS-POS: LAST
                                                   FCGROUP: GA
                                                   FCTYPE: WS-FOCUS
                                                   PROSODY: [ aff-en0ab-no-prosody
                                                              STMOOD: AFF
                                                              SPCODE: EN0AB
                                                              PROSODY-SET: [ no-prosody
                                                                             PROSODY-MARK1: [ no-mark
                                                                                              ACCENT-TONE: NOACCENT
                                                                                              BOUND-TONE: NOBOUND ] ]
                                                                             PROSODY-MARK2: [ no-mark
                                                                                              ACCENT-TONE: NOACCENT
                                                                                              BOUND-TONE: NOBOUND ] ] ] ] ] ] ] ] ]
                                                   SPR: < 4 > = = *NULL*
                                                   COMPS: *NULL*
                                                   ARGS: [ *ne-list*
```

```
                    FIRST: [ focus-word
                                HEAD: < 3 >
                                SPR: < 4 >
                                COMPS: *NULL*
                                ORTH: Kim ] ]
                    REST: *NULL* ] ] ]

REST: [ *ne-list*
        FIRST: [ focus-phrase
                HEAD: < 0 >
                SPR: < 5 > = [ *ne-list*
                            FIRST: < 2 >
                            REST: *NULL* ]
                COMPS: < 1 >
                ARGS: [ *ne-list*
                            FIRST: [ focus-word
                                    HEAD: < 0 >
                                    SPR: < 5 >
                                    COMPS: [ *ne-list*
                                            FIRST: < 6 > = [ focus-phrase
                                                    HEAD: < 7 > = [ actee-part
                                                            FOCUS: ACTEE
                                                            AGR1: [ ls-actee_a-aff-en0ab
                                                                    FOCUS-POS: LAST
                                                                    FCGROUP: GA
                                                                    FCTYPE: WS-FOCUS
                                                                    PROSODY: [ ls-aff-en0ab-actee
                                                                            STMOOD: AFF
                                                                            SPCODE: EN0AB
                                                                            PROSODY-SET: [ n-hem_lg-break
                                                                                    PROSODY-MARK1: [ hem_lg-break
                                                                                    ACCENT-TONE: L+H*
                                                                                    BOUND-TONE: L-L% ] ]
                                                                                    PROSODY-MARK2: [ no-mark
                                                                                    ACCENT-TONE: NOACCENT
                                                                                    BOUND-TONE: NOBOUND ] ] ] ] ] ] ] ] ]
                                                    SPR: *NULL*
                                                    COMPS: < 8 > = *NULL*
                                                    ARGS: [ *ne-list*
                                                            FIRST: <9> = [ focus-phrase
                                                            HEAD: <10> = [ actee-part
                                                                    FOCUS: ACTEE
                                                                    AGR1: [ pv-actee_a-aff-en0ab
                                                                            FOCUS-POS: PREV
```

```
                                                                FCGROUP: GA
                                                                FCTYPE: WS-FOCUS
                                                                PROSODY: [ pv-aff-en0ab-actee
                                                                        STMOOD: AFF
                                                                        SPCODE: EN0AB
                                                                        PROSODY-SET: [ d-hem_sh-break
                                                                                PROSODY-MARK1: [ dem_sh-break
                                                                                ACCENT-TONE: L*
                                                                                BOUND-TONE: L- ] ]
                                                                                PROSODY-MARK2: [ hem_sh-break
                                                                                ACCENT-TONE: L+H*
                                                                                BOUND-TONE: L- ] ] ] ] ] ] ] ] ]
                                                SPR: <11> = *NULL*
                                                COMPS: *NULL*
                                                ARGS: [ *ne-list*
                                                        FIRST: [ focus-word
                                                                HEAD: < 10 >
                                                                SPR: < 11 >
                                                                COMPS: *NULL*
                                                                ORTH: a ] ]
                                                        REST: *NULL* ] ] ]

REST: [ *ne-list*
        FIRST: [ focus-phrase
                HEAD: < 7 >
                SPR: < 12 > = [ *ne-list*
                                FIRST: < 9 >
                                REST: *NULL* ]
                COMPS: < 8 >
                ARGS: [ *ne-list*
                                FIRST: [ focus-word
                                        HEAD: < 7 >
                                        SPR: < 12 >
                                        COMPS: [ *ne-list*
                                                FIRST: < 13 > = [ focus-phrase
                                                HEAD: < 14 > = [ actee-part
                                                FOCUS: ACTEE
                                                AGR1: [ ls-actee_a-aff-en0ab
                                                        FOCUS-POS: LAST
                                                        FCGROUP: GA
                                                        FCTYPE: WS-FOCUS
                                                        PROSODY: [ ls-aff-en0ab-actee
                                                                STMOOD: AFF
                                                                SPCODE: EN0AB
```

PROSODY-SET: [n-hem_lg-break
PROSODY-MARK1: [hem_lg-break
ACCENT-TONE: L+H*
BOUND-TONE: L-L%]]
PROSODY-MARK2: [no-mark
ACCENT-TONE: NOACCENT
BOUND-TONE: NOBOUND]]]]]]]]]

SPR: *NULL*
COMPS: < 15 > = *NULL*
ARGS: [*ne-list*
FIRST: < 16 > = [focus-phrase
HEAD: < 17 > = [actee-part
FOCUS: ACTEE
AGR1: [pv-actee_a-aff-en0ab
FOCUS-POS: PREV
FCGROUP: GA
FCTYPE: WS-FOCUS
PROSODY: [pv-aff-en0ab-actee
STMOOD: AFF
SPCODE: EN0AB
PROSODY-SET: [d-hem_sh-break
PROSODY-MARK1: [dem_sh-break
ACCENT-TONE: L*
BOUND-TONE: L-]]
PROSODY-MARK2: [hem_sh-break
ACCENT-TONE: L+H*
BOUND-TONE: L-]]]]]]]]]
SPR: *NULL*
COMPS: < 18 > = *NULL*
ARGS: [*ne-list*
FIRST: < 19 > = [focus-phrase
HEAD: ¡20¿ = [actee-part
FOCUS: ACTEE
AGR1: [pv-actee_a-aff-en0ab
FOCUS-POS: PREV
FCGROUP: GA
FCTYPE: WS-FOCUS
PROSODY: [pv-aff-en0ab-actee
STMOOD: AFF
SPCODE: EN0AB
PROSODY-SET: [d-hem_sh-break
PROSODY-MARK1: [dem_sh-break
ACCENT-TONE: L*

```
                                                                                                BOUND-TONE: L- ] ]
                                                                                                PROSODY-MARK2: [ hem_sh-break
                                                                                                ACCENT-TONE: L+H*
                                                                                                BOUND-TONE: L- ]]]]]]]]
                                                                SPR: < 21 > = *NULL*
                                                                COMPS: *NULL*
                                                                ARGS: [ *ne-list*
                                                                        FIRST: [ focus-word
                                                                                HEAD: < 20 >
                                                                                SPR: < 21 >
                                                                                COMPS: *NULL*
                                                                                ORTH: for ] ]
                                                                        REST: *NULL* ] ] ]
                                                        REST: [ *ne-list*
                                                                FIRST: [ focus-phrase
                                                                        HEAD: < 17 >
                                                                        SPR: < 22 > = [ *ne-list*
                                                                                FIRST: < 19 >
                                                                                REST: *NULL* ]
                                                                        COMPS: < 18 >
                                                                        ARGS: [ *ne-list*
                                                                                FIRST: [ focus-word
                                                                                                HEAD: < 17 >
                                                                                                SPR: < 22 >
                                                                                                COMPS: *NULL*
                                                                                                ORTH: her ] ]
                                                                                REST: *NULL* ] ] ] ]
                                                                        REST: *NULL* ] ] ] ] ]
                REST: [ *ne-list*
                        FIRST: [ focus-phrase
                                HEAD: < 14 >
                                SPR: < 23 > = [ *ne-list*
                                        FIRST: < 16 >
                                        REST: *NULL* ]
                                COMPS: < 15 >
                                ARGS: [ *ne-list*
                                        FIRST: [ focus-word
                                                HEAD: < 14 >
                                                SPR: < 23 >
                                                                COMPS: *NULL*
                                                                ORTH: mother ] ]
                                        REST: *NULL* ] ] ] ]
                                        REST: *NULL* ] ] ] ] ]
                        REST: *NULL* ] ]
```

```
                    ORTH: flower ] ]

REST: [ *ne-list*
        FIRST: < 13 >
        REST: *NULL* ] ] ] ] ] ]
REST: *NULL* ] ] ] ] ]
REST: *NULL* ] ]
ORTH: bought ] ]
REST: [ *ne-list*
        FIRST: < 6 >
        REST: *NULL* ] ] ] ] ] ]
      REST: *NULL* ] ] ] ] ]
```

Appendix B

Travel Reservation Dialogues

B.1 Dialogue 1

Table B.1: Dialogue 1

No.	Speaker	Name	Sentence
	Travel Agent	Peter	Halifax Travel. Peter Speaking
	Caller	Tom	Hello. I am Tom. I would like to make some travel arrangement?
	Travel Agent	Peter	What day did you wish to travel?
	Caller	Tom	This is May 12th.
1	Travel Agent	Peter	(a) Fly from Indiana. (b) Fly from Indiana? (c) Fly from Indiana. (d) Fly from Indiana?
	Caller	Tom	Fly from Indiana.
	Travel Agent	Peter	And going where?
	Caller	Tom	Going to Boston. There is an eight o'clock.
2	Caller	Tom	(a) I beleive the United flight. (b) I beleive the United flight? (c) I beleive the United flight. (d) I beleive the United flight?
3	Travel Agent	Peter	(a) That is eight a.m. (b) That is eight a.m.? (c) That is eight a.m. (d) That is eight a.m.?
	Caller	Tom	Right.
	Travel Agent	Peter	I got that. Returning on what day?
	Caller	Tom	Same day.
	Travel Agent	Peter	What time?
	Caller	Tom	Six o'clock
4	Travel Agent	Peter	(a) Again on the United flight. (b) Again on the United flight? (c) Again on the United flight. (d) Again on the United flight?
	Caller	Tom	O.K.
	Travel Agent	Peter	We have confirmed on th United flight 115.
	Caller	Tom	O.K.
	Travel Agent	Peter	The fare on this is $198.
5	Caller	Tom	(a) You are kidding. (b) You are kidding? (c) You are kidding. (d) You are kidding?
	Caller	Tom	What happended to $78 fates, or those sort of things?
	Travel Agent	Peter	Those you need to stay over a Saturday night.
	Caller	Tom	Well, I cannot do that.
	Travel Agent	Peter	Yes.
	Caller	Tom	Or, O.K.
	Travel Agent	Peter	Very good. Thank you.
	Caller	Tom	Thank you.

B.2 Dialogue 2

Table B.2: Dialogue 2

No.	Speaker	Name	Sentence
	Caller	Tom	This is Tom. I need to make some travel arrangement, please
	Travel Agent	Peter	What is the passenger's last name?
6	Caller	Tom	(a) It is Thomas. (b) It is Thomas? (c) It is Thomas. (d) It is Thomas?
	Travel Agent	Peter	And, first name.
7	Caller	Tom	(a) is Max. (b) is Max? (c) is Max. (d) is Max?
	Travel Agent	Peter	What day would he like to travel?
	Caller	Tom	He needs to go on Tuesday May 19th.
	Travel Agent	Peter	O.K.
8	Caller	Tom	(a) Fly from Boston to Toronto. (b) Fly from Boston to Toronto? (c) Fly from Boston to Toronto. (d) Fly from Boston to Toronto?
	Travel Agent	Peter	O.K. I have got that. and returning when?
	Caller	Tom	Let's leave the return open right now. He needs to figure that out.
	Travel Agent	Peter	Shall we wait on anything else, until we know, what day he is returning
	Caller	Tom	We should know sometime this afternoon
	Travel Agent	Peter	O.K.
9	Travel Agent	Peter	(a) If you want to give me a call back. (b) If you want to give me a call back? (c) If you want to give me a call back. (d) If you want to give me a call back?
	Caller	Tom	Great.
10	Caller	Tom	(a) Will do. (b) Will do? (c) Will do. (d) Will do?
	Travel Agent	Peter	O.K. Thank you.
	Caller	Tom	Thank you. Bye

B.3 Dialogue 3

Table B.3: Dialogue 3

No.	Speaker	Name	Sentence
	Travel Agent	Peter	Halifax Travel. May I help you?
	Caller	Tom	Yes, this is Tom. I would like to make some travel arrangement.
	Travel Agent	Peter	What do you need to do?
	Caller	Tom	In August. I need to go to Southbend, Indiana. And I want to fly on the United flight.
	Travel Agent	Peter	One second.
11	Travel Agent	Peter	(a) What time do you want to leave? (b) What time do you want to leave? (c) What time do you want to leave? (d) What time do you want to leave?
	Caller	Tom	I want to go from Boston to Southbend on the United 262.
	Travel Agent	Peter	O.K. That leaves at 4:50 p.m.
	Caller	Tom	Arrives 5:24 p.m.?
	Travel Agent	Peter	Right. There is 10:09 a.m. flight out of Florida, that connects fairly well with that.
	Caller	Tom	How close is that?
	Travel Agent	Peter	It is flight 126, leave at 1:00 p.m. And gets into Boston at 3:58 p.m.
	Caller	Tom	Yes.
12	Travel Agent	Peter	(a) It is a little less than an hour. (b) It is a little less than an hour. (c) It is a little less than an hour. (d) It is a little less than an hour.
	Caller	Tom	O.K.
13	Caller	Tom	(a) That sounds good. (b) That sounds good. (c) That sounds good. (d) That sounds good.
	Travel Agent	Peter	O.K.
	Caller	Tom	Is there one that gets there just a little bit earlier. In case of any problems.
	Travel Agent	Peter	The next one earlier leaves at 10:00 a.m. And, gets there at 1:05, which is an awfully long wait.
	Caller	Tom	O.K. Book me on that then.
	Travel Agent	Peter	O.K.
	Caller	Tom	Can you give me that again?
	Travel Agent	Peter	It is the United flight.
	Caller	Tom	Exactly.
	Travel Agent	Peter	And then a return for you.
14	Caller	Tom	(a) I am leaving August 9th. (b) I am leaving August 9th. (c) I am leaving August 9th. (d) I am leaving August 9th.
	Travel Agent	Peter	Back to Florida.
	Caller	Tom	Yes. But from Boston.
	Travel Agent	Peter	Oh, you are going to get your own way, between Southbend and Boston.
	Caller	Tom	Yes.
	Travel Agent	Peter	O.K. When did you want to leave Boston?
15	Caller	Tom	(a) I want to leave Boston around 5 p.m. (b) I want to leave Boston around 5 p.m. (c) I want to leave Boston around 5 p.m. (d) I want to leave Boston around 5 p.m.
	Caller	Tom	United also.
	Travel Agent	Peter	O.K. There is a 4:50 p.m.
	Caller	Tom	That sounds good.
	Travel Agent	Peter	O.K. and 5:59 p.m.
	Caller	Tom	4:50 sounds good.
	Travel Agent	Peter	O.K.

B.4 Dialogue 4

Table B.4: Dialogue 4

No.	Speaker	Name	Sentence
	Travel Agent	Peter	Halifax Travel. May I help you?
	Caller	Tom	Yes. This is Tom. I am calling about a trip for Jame.
16	Caller	Tom	(a) This is his Toronto trip. (b) This is his Toronto trip. (c) This is his Toronto trip. (d) This is his Toronto trip.
	Caller	Tom	and That is on Monday
	Travel Agent	Peter	O.K. I see it here.
	Caller	Tom	O.K.
17	Caller	Tom	(a) He needs to go to Toronto. (b) He needs to go to Toronto. (c) He needs to go to Toronto. (d) He needs to go to Toronto.
	Caller	Tom	I understand another group has made the reservations So, i need to get the information on that
	Travel Agent	Peter	Just the information about what we have.
	Caller	Tom	Yes.
	Travel Agent	Peter	O.K.
18	Travel Agent	Peter	(a) He is on American flight 262. (b) He is on American flight 262. (c) He is on American flight 262. (d) He is on American flight 262.
	Travel Agent	Peter	Out of San Francisco at 11:50 p.m.
	Caller	Tom	O.K.
	Travel Agent	Peter	Arriving Toronto at 5:56 a.m., on 25th.
	Caller	Tom	I am sorry. Time again?
	Travel Agent	Peter	5:56 a.m. on 25th of July
	Caller	Tom	O.K.
	Travel Agent	Peter	Connecting to Boston.
	Caller	Tom	O.K.
	Travel Agent	Peter	On his return, it is Tuesday 25th.
	Caller	Tom	O.K.
	Travel Agent	Peter	American flight 153.
	Caller	Tom	O.K.
	Travel Agent	Peter	Out of Toronto at 8:03 p.m.
	Caller	Tom	Wait a minute.
19	Caller	Tom	(a) That is an American flight. (b) That is an American flight. (c) That is an American flight. (d) That is an American flight.
	Travel Agent	Peter	Yes
20	Caller	Tom	(a) They are all American flights? (b) They are all American flights? (c) They are all American flights? (d) They are all American flights?
	Travel Agent	Peter	Yes. Out of Toronto at 8:03 p.m. Arrive Chicago at 9:13 p.m.
	Caller	Tom	O.K.
	Travel Agent	Peter	And, arrive San Francisci at 12:03 a.m., on 26 th. Just after midnight.
	Caller	Tom	O.K Good. Thank you.
	Travel Agent	Peter	O.K. Bye.

Appendix C

Informed Consent Form

Title Page

(a) Title

Evaluation of Focus Recognition in Computer-generated Speech based on Algorithmic Prosodic Modification

(b) Local Principle Investigator

Lalita Narupiyakul
Faculty of Computer Science, Dalhousie University
6050 University Ave. Halifax NS B3H 1W5
Phone: (902) 489-9566 Email: (902) lalita@cs.dal.ca

(c) Degree Program

Ph.D. (Computer Science)

(d) Supervisor

Vlado Keselj
Faculty of Computer Science, Dalhousie University
6050 University Ave. Halifax, NS B3H 1W5
Phone: (902) 494-2893 Email: vlado@cs.dal.ca
Nick Cercone
FSE Dean's Office, 355 Lumbers Bldg, York University
4700 Keele St. Toronto, ON M3J 1P3
Phone: 416-736-5051 Email: ncercone@yorku.ca

(f) Contact Person

If you have any questions, would like any further information or may need assistance of any kind, please feel free to contact Lalita Narupiyakul (see 1(b) for contact information).

Introduction

We invite you to take part in a research study being conducted by Lalita Narupiyakul who is a graduate student at Dalhousie University as part of her Ph.D program. Your participation in this study is voluntary and you may withdraw from the study at any time. The study is described below. This description tells you about risks, inconvenience, or discomfort which you might experience. Participating in the study will not likely benefit you, but we might learn things that will benefit others. You should discuss any questions you have about this study with Lalita Narupiyakul (see 1(b) for contact information, above).

Purpose of the Study

In the spoken language, the same utterance may convey different meanings to a hearer. Such ambiguities can be resolved by emphasizing accents at different positions in a sentence. The main objective of the study is to determine whether the prosody, generated by speech synthesis with our focus to emphasize tone (FET) system, can convey the focus to the listeners.

Study Design

In this experiment, listeners will listen to computer generated speech that includes accents. Listeners will then rate the clarity of that communication in two different tasks: (i) the listener's preference between utterances with marked focus or without focus; and (ii) the listener's preference among different focus parts, which are emphasized by accents in the different positions of a sentence).

Who Can Participate in the Study

You may participate in this study if you are able to communicate in English and are normal-hearing adult listener.

No special screening for this study is required. The short conversation, such as greeting between the investigator and participant, is enough to qualify your participation.

Who Will Conduct Research

The experiment will be run by Lalita Narupiyakul (see 1(b) for contact information).

What You Will Be Asked to Do

You will listen to several travel reservation dialogues. Within each dialogue, some sentences will be presented with different prosodies (accents). You will select the one that makes the most sense in the context of the dialogue. For example, let us consider two sentences: (i) "Tom will win?", and (ii) "Tom will win." These two sentences have different prosodies and different meanings. The prosody of (i) contains a high tone at the end of sentence, while (ii) has a low tone at the end. The meaning of (i) is that speaker is not sure or is curious whether "Tom will win?", while the speaker of (ii) confirms that Tom will not lose.

Possible Risks and Discomforts

There are no foreseeable risks or hazards specific to this experiment. The participant can choose to use speaker or headphones, which are comfortable and have noise reduction. There is no risk of auditory damage. The loudness is at the normal level of human speaking between 20-60 dB (such as normal talking at 1-meter distance or a quiet room). However, you are allowed to take a break during the experiment or withdraw from the experiment, if you wish or feel uncomfortable, finding the task difficult, etc. (dB = decibel)

Possible Benefits

By participation in this research, you may obtain altruistic benefits and may not benefit you directly. Your participation will contribute to knowledge that should help us improve the computer-generated speech system.

Compensation / Expense Reimbursement

If you are a student of the Dalhousie University, then you will be compensated by the amount of $5.00 for participation and irrespective of whether or not you complete

the study. You will be paid in cash, whenever your participation in the study comes to an end.

Confidentiality & Anonymity

We will not collect or record participant's personal information in the experiment. Your data will be in form of an answer sheet including your answers and your opinion. Your name or identification will be anonymous and is represented by numerical number such as 01, 02,. . ., and so on. All of your confidentiality is protected, concealed and safely stored in a locked file cabinet, Room 330, Faculty of Computer Science, Dalhousie University. The room is locked automatically after office hours. Access to the data will be restricted only to Primary Investigator and her supervisor.

Dalhousie University Policy on Research Integrity requires that data be securely maintained by the institution for five years, post publication. Your identity, such as name, will not appear in any reports or publications.

Questions

You may ask the Principle Investigator (see 1(b) for contact information, above) any questions about this study. Any new information which might affect your decision to participate in the study will be provided to you at the earliest opportunity.

Summary

You will receive a copy of the informed consent form for your records. The published results of study will be posted on the website "www.cs.dal.ca/~lalita/ experiment/result/" after the project has been completed. Please feel free to give us any comments or discuss with the experimenter any details of the study.

Problems or Concerns

In the event that you have any difficulties with, or wish to voice concern about any aspect of your participation in this study, you may contact Patricia Lindley, Director of Dalhousie University's Office of Human Research Ethics Administration for assistance: (902) 494-1462, patricia.lidley@dal.ca

Signature

Study Title: Evaluation of Focus Recognition in Computer-generated Speech based on Algorithmic Prosodic Modification

I have read the explanation about this study. I have been given the opportunity to discuss it and my questions have been answered to my satisfaction. I hereby consent to take part in this study. However I realize that my participation is voluntary and that I am free to withdraw from the study at anytime.

Participant's name printed: _____

Participant signature: _____ Date: _____

Experimenter signature: _____ Date: _____

Appendix D

Experimental Results of Prosodic Annotation

D.1 Prosodic Annotation without Tone Alignment

I'm,sorry,

I'm	Equal=1	L+H*	H*	H*
sorry	NEqual=1	L-L%	0	H*L-L%

Where,are,you,leaving,from,

Where	Equal=2	0	0	0
are	Equal=3	0	0	0
you	NEqual=2	L+H*	H*	0
leaving	Equal=4	L+H*	H*	H*
from	Equal=5	L-L%	0	L-L%

What,city,are,you,leaving,from,

What	Equal=6	0	0	0
city	Equal=7	H*	0	H*
are	Equal=8	0	0	0
you	NEqual=3	L+H*	H*	0
leaving	Equal=9	L+H*	H*	H*
from	Equal=10	L-L%	0	L-L%

Are,you,a,registered,user,

Are	Equal=11	0	0	0
you	Equal=12	L+H*	H*	H*
a	Equal=13	0	0	0
registered	Equal=14	L+H*	H*	H*
user	NEqual=4	H-H%	0	L+H*H-H%

You,may,interrupt,these,instructions,at,any,time,
by,saying,good,enough,

You	Equal=15	H*	0	H*
may	Equal=16	0	0	0
interrupt	Equal=17	L+H*	H*	H*
these	NEqual=5	L+H*	H*	0
instructions	NEqual=6	L-	L-	L-L%
at	NEqual=7	L+H*	H*	0
any	NEqual=8	L+H*	H*	0
time	NEqual=9	L-	L-	0
by	NEqual=10	L+H*	H*	0
saying	NEqual=11	L-	L-	L+H*L-L%
good	NEqual=12	L+H*	H*	0
enough	NEqual=13	L-L%	0	H*L-L%

You,can,interrupt,the,system,at,any,time,by,saying,
anything,you,wish,

You	Equal=18	H*	0	H*
can	Equal=19	0	0	0
interrupt	Equal=20	L+H*	H*	H*
the	NEqual=14	L+H*	H*	0
system	Equal=21	L+H*	H*	H*
at	NEqual=15	L+H*	H*	0
any	NEqual=16	L+H*	H*	0
time	NEqual=17	L-	L-	L-L%
by	NEqual=18	L+H*	H*	0
saying	Equal=22	L+H*	H*	H*
anything	Equal=23	L+H*	H*	H*
you	NEqual=19	L+H*	H*	0
wish	Equal=24	L-L%	0	L-L%

To,end,the,call,say,good,bye,

To	Equal=25	0	0	0
end	NEqual=20	0	0	H*
the	Equal=26	0	0	0
call	NEqual=21	H*	0	H*L-L%
say	Equal=27	0	0	0
good	Equal=28	L+H*	H*	H*
bye	NEqual=22	L-L%	0	H*L-L%

This,is,the,end,of,the,instructions,

This	Equal=29	0	0	0
is	Equal=30	0	0	0
the	NEqual=23	L+H*	H*	0
end	Equal=31	0	0	0
of	Equal=32	0	0	0
the	NEqual=24	L+H*	H*	0
instructions	NEqual=25	L-L%	0	L+H*L-L%

It,knows,about,major,US,cities,and,some,
international,destinations,

It	Equal=33	H*	0	H*
knows	Equal=34	L+H*	H*	H*
about	NEqual=26	L+H*	H*	0
major	Equal=35	L+H*	H*	H*
US	NEqual=27	L+H*	H*	0
cities	NEqual=28	L-	L-	L-H%
and	Equal=36	0	0	0

some	NEqual=29	L+H*	H*	0
international	Equal=37	L+H*	H*	H*
destinations	NEqual=30	L-L%	0	L+H*L+H*L-L%

If,you,need,to,make,a,correction,just,restate,the,
new,information

If	Equal=38	H*	0	0
you	Equal=39	H*	0	H*
need	Equal=40	L+H*	H*	H*
to	NEqual=31	L+H*	H*	0
make	Equal=41	L+H*	H*	H*
a	NEqual=32	L+H*	H*	0
correction	NEqual=33	L-	L-	!H*L-L%
just	NEqual=34	L+H*	H*	0
restate	Equal=42	L+H*	H*	H*
the	NEqual=35	L+H*	H*	0
new	NEqual=36	L+H*	H*	0
information	NEqual=37	L-L%	0	L+H*
				L+H*L-L%

If,you,need,help,at,any,time,please,say,help,

If	Equal=43	H*	0	0
you	Equal=44	H*	0	H*
need	Equal=45	0	0	0
help	NEqual=38	L-	L-	H*L-L%
at	NEqual=39	L+H*	H*	0
any	NEqual=40	L+H*	H*	0
time	NEqual=41	L-	L-	L-L%
please	NEqual=42	L+H*	H*	0
say	Equal=46	L+H*	H*	H*
help	NEqual=43	L-L%	0	H*L-L%

I,didnt,catch,that,

I	Equal=47	H*	0	H*
didnt	Equal=48	0	0	0
catch	Equal=49	L+H*	H*	H*
that	Equal=50	0	0	0

This,call,is,being,recorded,for,development,purposes,
and,may,be,shared,with,other,researchers,

This	Equal=51	0	0	0
call	Equal=52	H*	0	H*
is	Equal=53	0	0	0
being	Equal=54	0	0	0
recorded	Equal=55	0	0	0
for	NEqual=44	L+H*	H*	0
development	Equal=56	L+H*	H*	H*
purposes	NEqual=45	L-	L-	L-H%
and	NEqual=46	L+H*	H*	0
may	NEqual=47	L+H*	H*	0
be	NEqual=48	L+H*	H*	0
shared	NEqual=49	L+H*	H*	0
with	Equal=57	0	0	0
other	Equal=58	L+H*	H*	L+H*
researchers	NEqual=50	L-L%	0	L+H*L-L%

Where,do,you,want,to,go,

Where	Equal=59	0	0	0
do	Equal=60	L+H*	H*	H*
you	Equal=61	L+H*	H*	H*
want	Equal=62	L+H*	H*	H*
to	NEqual=51	L+H*	H*	0
go	Equal=63	L-L%	0	L-L%

Where,would,you,like,to,go,

Where	Equal=64	0	0	0
would	Equal=65	0	0	0
you	Equal=66	L+H*	H*	H*
like	NEqual=52	0	0	H*
to	Equal=67	0	0	0
go	Equal=68	L-L%	0	L-L%

If,you,want,us,to,do,that,please,say,book,this,trip,
or,otherwise,say,continue,

If	Equal=69	H*	0	0
you	Equal=70	H*	0	H*
want	Equal=71	L+H*	H*	H*
us	NEqual=53	L-	L-	H*
to	NEqual=54	L+H*	H*	0
do	NEqual=55	L+H*	H*	0
that	NEqual=56	L-L%	L-L%0	
please	NEqual=57	L+H*	H*	0
say	Equal=72	L+H*	H*	H*
book	Equal=73	L+H*	H*	H*
this	NEqual=58	L+H*	H*	0
trip	NEqual=59	L-	L-	H*L-L%
or	NEqual=60	L+H*	H*	0
otherwise	NEqual=61	L-	L-	H*L-L%
say	Equal=74	L+H*	H*	H*
continue	NEqual=62	L-L%	0	H*L-L%

Sorry,Im,not,sure,I,understood,what,you,said,

Sorry	NEqual=63	0	0	H*L-H%
Im	Equal=75	0	0	0
not	NEqual=64	0	0	H*
sure	Equal=76	H*	0	H*
I	NEqual=65	L+H*	H*	0
understood	NEqual=66	L-	L-	H*L-L%

what	NEqual=67	L+H*	H*	0
you	Equal=77	L+H*	H*	H*
said	NEqual=68	L-L%	0	H*L-L%

To,help,us,improve,our,system,please,answer,the,
following,questions,

To	Equal=78	0	0	0
help	NEqual=69	0	0	H*
us	Equal=79	H*	0	H*
improve	Equal=80	0	0	0
our	NEqual=70	L+H*	H*	0
system	NEqual=71	0	0	!H*L-H%
please	NEqual=72	L+H*	H*	0
answer	Equal=81	L+H*	H*	H*
the	NEqual=73	L+H*	H*	0
following	Equal=82	L+H*	H*	H*
questions	NEqual=74	L-L%	0	L+H*L-L%

This,session,is,now,over,

This	Equal=83	0	0	0
session	Equal=84	H*	0	H*
is	Equal=85	0	0	0
now	Equal=86	0	0	0
over	Equal=87	0	0	0

Thank,you,for,calling,

Thank	NEqual=75	0	0	H*
you	Equal=88	H*	0	H*
for	Equal=89	0	0	0
calling	NEqual=76	L-L%	0	L+H*L-L%

Are,you,satisfied,with,this,itinerary,

Are	Equal=90	H*	0	0
you	Equal=91	H*	0	H*
satisfied	Equal=92	L+H*	H*	H*
with	NEqual=77	L+H*	H*	0
this	NEqual=78	L+H*	H*	0
itinerary	NEqual=79	H-H%	0	L+H*
				L+H*H-H%

To,go,on,I,need,you,to,answer,the,following,question,

To	Equal=93	0	0	0
go	NEqual=80	0	0	H*
on	NEqual=81	H*	0	L+H*L-L%
I	Equal=94	0	0	0
need	Equal=95	0	0	0
you	NEqual=82	L-	L-	H*L-L%
to	NEqual=83	L+H*	H*	0
answer	Equal=96	L+H*	H*	H*
the	NEqual=84	L+H*	H*	0
following	Equal=97	L+H*	H*	H*
question	NEqual=85	L-L%	0	L+H*L-L%

Perhaps,youre,asking,for,something,I,dont,know,about,

Perhaps	NEqual=86	0	0	H*
youre	Equal=98	0	0	0
asking	Equal=99	0	0	0
for	NEqual=87	L+H*	H*	0
something	NEqual=88	L-L%	L-L%	L+H*
I	NEqual=89	L+H*	H*	0
dont	NEqual=90	L+H*	H*	0
know	NEqual=91	L+H*	H*	0
about	Equal=100	L-L%	0	L-L%

Im,still,having,trouble,understanding,you,

Im	Equal=101	H*	0	H*
still	NEqual=92	0	0	H*
having	Equal=102	0	0	0
trouble	NEqual=93	L+H*	H*	!H*
understanding	NEqual=94	L+H*	H*	0
you	Equal=103	L-L%	0	L-L%

Sorry,I,did,not,understand,what,you,said,

Sorry	Equal=104	H*	0	H*
I	Equal=105	H*	0	H*
did	Equal=106	0	0	0
not	Equal=107	0	0	0
understand	Equal=108	0	0	0
what	Equal=109	0	0	0
you	Equal=110	0	0	0
said	Equal=111	L-L%	0	L-L%

What,is,your,full,name,

What	Equal=112	0	0	0
is	Equal=113	0	0	0
your	Equal=114	L+H*	H*	H*
full	NEqual=95	L+H*	H*	0
name	Equal=115	L-L%	0	L-L%

You,can,say,help,at,any,time,

You	Equal=116	H*	0	H*
can	Equal=117	0	0	0
say	Equal=118	0	0	0
help	Equal=119	0	0	0
at	Equal=120	0	0	0
any	Equal=121	0	0	0
time	NEqual=96	0	0	L-L%

Which,one,did,you,want,

Which	Equal=122	H*	0	H*
one	Equal=123	H*	0	H*
did	Equal=124	L+H*	H*	H*
you	NEqual=97	L+H*	H*	0
want	Equal=125	L-L%	0	L-L%

When,youre,having,trouble,please,ask,for,help,

When	Equal=126	0	0	0
youre	Equal=127	0	0	0
having	Equal=128	L+H*	H*	H*
trouble	NEqual=98	L-	L-	!H*L-L%
please	NEqual=99	L+H*	H*	0
ask	Equal=129	L+H*	H*	H*
for	NEqual=100	L+H*	H*	0
help	Equal=130	L-L%	0	L-L%

Please,be,our,guest,

Please	Equal=131	H*	0	H*
be	Equal=132	0	0	0
our	Equal=133	L+H*	H*	H*
guest	Equal=134	L-L%	0	L-L%

You,can,ask,me,for,help,at,any,time,

You	Equal=135	H*	0	H*
can	Equal=136	0	0	0
ask	Equal=137	L+H*	H*	H*
me	NEqual=101	L-	L-	L-L%
for	NEqual=102	L+H*	H*	0
help	NEqual=103	L-	L-	H*L-L%
at	NEqual=104	L+H*	H*	0
any	NEqual=105	L+H*	H*	0
time	Equal=138	L-L%	0	L-L%

Would,you,like,me,to,summarize,your,trip,

Would	Equal=139	H*	0	H*
you	Equal=140	H*	0	H*
like	Equal=141	L+H*	H*	H*
me	NEqual=106	L+H*	H*	0
to	NEqual=107	L+H*	H*	0
summarize	Equal=142	L+H*	H*	H*
your	NEqual=108	L+H*	H*	0
trip	Equal=143	H-H%	0	H-H%

When,youre,finished,just,hang,up,

When	Equal=144	0	0	0
youre	Equal=145	0	0	0
finished	NEqual=109	L+H*	H*	H*L-H%
just	NEqual=110	L+H*	H*	0
hang	Equal=146	L+H*	H*	H*
up	Equal=147	L-L%	0	L-L%

Thank,you,for,using,the,Carnegie,Mellon,Communicator,

Thank	NEqual=111	0	0	H*
you	Equal=148	H*	0	H*
for	Equal=149	0	0	0
using	Equal=150	0	0	0
the	NEqual=112	L+H*	H*	0
Carnegie	NEqual=113	L+H*	H*	!H*
Mellon	NEqual=114	L+H*	H*	0
Communicato	NEqual=115	L-L%	0	L+H* L+H*L-L%

Ive,made,no,hotel,reservations,for,your,trip,

Ive	Equal=151	0	0	0
made	NEqual=116	0	0	H*
no	NEqual=117	L+H*	H*	0
hotel	NEqual=118	L+H*	H*	0
reservations	NEqual=119	L-	L-	L-H%
for	NEqual=120	L+H*	H*	0
your	Equal=152	L+H*	H*	H*
trip	NEqual=121	L-L%	0	H*L-L%

Please,say,something,

Please	Equal=153	H*	0	H*
say	Equal=154	L+H*	H*	H*
something	NEqual=122	L-L%	0	H*L-L%

Otherwise,I,will,need,to,hang,up,

Otherwise	NEqual=123	H*	0	H*H*
I	Equal=155	H*	0	0
will	Equal=156	0	0	0
need	Equal=157	0	0	0
to	Equal=158	0	0	0
hang	Equal=159	0	0	0
up	Equal=160	0	0	0

Were,you,attempting,to,arrange,travel,that,you,
actually,plan,to,take,

Were	Equal=161	0	0	0
you	NEqual=124	0	0	H*H*
attempting	Equal=162	0	0	0
to	Equal=163	0	0	0
arrange	Equal=164	0	0	0
travel	NEqual=125	0	0	H*L-L%
that	Equal=165	0	0	0
you	NEqual=126	0	0	H*H*
actually	NEqual=127	0	0	H*

plan	Equal=166	0	0	0
to	Equal=167	0	0	0
take	Equal=168	H-H%	0	H-H%

Should,I,summarize,your,trip,

Should	Equal=169	H*	0	H*
I	Equal=170	H*	0	H*
summarize	Equal=171	0	0	0
your	Equal=172	0	0	0
trip	Equal=173	H-H%	0	H-H%

Is,that,correct,

Is	Equal=174	0	0	0
that	Equal=175	0	0	0
correct	Equal=176	H-H%	0	H-H%

Which,city,and,state,did,you,want,

Which	Equal=177	H*	0	H*
city	Equal=178	H*	0	H*
and	Equal=179	H*	0	0
state	Equal=180	H*	0	0
did	Equal=181	0	0	0
you	Equal=182	0	0	0
want	Equal=183	L-L%	0	L-L%

Ive,made,no,car,reservations,for,this,trip,

Ive	Equal=184	0	0	0
made	NEqual=128	0	0	H*
no	NEqual=129	L+H*	H*	0
car	NEqual=130	L+H*	H*	0
reservations	NEqual=131	0	0	!H*L-H%
for	Equal=185	0	0	0
this	Equal=186	0	0	0
trip	NEqual=132	L-L%	0	H*L-L%

Im,sorry,I,couldnt,find,your,profile,

Im	Equal=187	H*	0	H*
sorry	NEqual=133	L+H*	H*	H*L-H%
I	NEqual=134	L+H*	H*	0
couldnt	NEqual=135	L+H*	H*	0
find	NEqual=136	L-	L-	H*
your	NEqual=137	L+H*	H*	0
profile	NEqual=138	L-L%	0	L+H*L-L%

Will,you,return,to,pittsburgh,from,seattle,

Will	NEqual=139	0	0	H*
you	Equal=188	H*	0	H*
return	Equal=189	0	0	0
to	Equal=190	0	0	0
pittsburgh	Equal=191	L+H*	H*	H*
from	NEqual=140	L+H*	H*	0
seattle	NEqual=141	H-H%	0	L+H*H-H%

There,is,currently,no,specific,help,for,this,topic,

There	Equal=192	H*	0	H*
is	Equal=193	0	0	0
currently	NEqual=142	L-	L-	H*
no	NEqual=143	L+H*	H*	0
specific	NEqual=144	L+H*	H*	!H*
help	NEqual=145	L-	L-	L-L%
for	NEqual=146	L+H*	H*	0
this	NEqual=147	L+H*	H*	0
topic	NEqual=148	L-L%	0	H*L-L%

Or,you,can,go,on,by,answering,the,question,

Or	Equal=194	H*	0	0
you	Equal=195	H*	0	H*
can	Equal=196	0	0	0
go	Equal=197	0	0	0
on	NEqual=149	L+H*	H*	L+H*L-L%
by	NEqual=150	L+H*	H*	0
answering	Equal=198	L+H*	H*	H*
the	NEqual=151	L+H*	H*	0
question	NEqual=152	L-L%	0	L+H*L-L%

I,might,feel,better,then,

I	Equal=199	H*	0	H*
might	Equal=200	0	0	0
feel	Equal=201	0	0	0
better	Equal=202	L+H*	H*	H*
then	Equal=203	L-L%	0	L-L%

Which,destination,did,you,want,

Which	Equal=204	H*	0	H*
destination	NEqual=153	H*	0	H*H*
did	Equal=205	0	0	0
you	Equal=206	0	0	0
want	Equal=207	L-L%	0	L-L%

Something,is,wrong,with,the,flight,retrieval,

Something	Equal=208	H*	0	H*
is	Equal=209	0	0	0
wrong	Equal=210	0	0	0
with	NEqual=154	L+H*	H*	0
the	NEqual=155	L+H*	H*	0
flight	NEqual=156	L+H*	H*	0
retrieval	NEqual=157	L-L%	0	L+H*L-L%

Please,tell,us,any,comments,

Please	Equal=211	H*	0	H*
tell	Equal=212	L+H*	H*	H*
us	NEqual=158	L-	L-	0
any	NEqual=159	L+H*	H*	0
comments	NEqual=160	L-L%	0	L+H*L-L%

I,have,a,best,Western,

I	Equal=213	H*	0	H*
have	Equal=214	0	0	0
a	Equal=215	0	0	0
best	NEqual=161	L+H*	H*	0
Western	NEqual=162	L-L%	0	L+H*L-L%

At,this,point,you,have,selected,a,leg,of,your,itinerary,

At	Equal=216	0	0	0
this	Equal=217	0	0	0
point	NEqual=163	0	0	H*
you	Equal=218	H*	0	0
have	Equal=219	0	0	0
selected	Equal=220	0	0	0
a	Equal=221	0	0	0
leg	NEqual=164	L+H*	H*	0
of	NEqual=165	L+H*	H*	0
your	Equal=222	L+H*	H*	H*
itinerary	NEqual=166	L-L%	0	L+H* L+H*L-L%

What,time,do,you,need,to,depart,

What	Equal=223	0	0	0
time	Equal=224	L+H*	H*	H*
do	NEqual=167	0	0	H*
you	Equal=225	0	0	0
need	NEqual=168	L+H*	H*	0
to	Equal=226	0	0	0
depart	Equal=227	L-L%	0	L-L%

Im,sorry,we,dont,have,your,profile,yet,

Im	NEqual=169	0	0	H*
sorry	NEqual=170	0	0	H*L-H%
we	Equal=228	0	0	0
dont	Equal=229	0	0	0
have	Equal=230	0	0	0
your	Equal=231	0	0	0
profile	NEqual=171	0	0	L+H*
yet	Equal=232	0	0	0

I,have,a,marriott,

I	Equal=233	H*	0	H*
have	Equal=234	0	0	0
a	NEqual=172	L+H*	H*	0
marriott	NEqual=173	L-L%	0	L+H*L-L%

Hello,Wei,its,nice,to,hear,from,you,again,

Hello	Equal=235	H*	0	H*
Wei	NEqual=174	H*	0	H*L-H%
its	Equal=236	0	0	0
nice	Equal=237	L+H*	H*	H*
to	Equal=238	0	0	0
hear	Equal=239	0	0	0
from	Equal=240	0	0	0
you	Equal=241	0	0	0
again	Equal=242	L-L%	0	L-L%

To,correct,any,part,of,the,itinerary,just,restate,the,
new,information,

To	Equal=243	0	0	0
correct	Equal=244	L+H*	H*	H*
any	NEqual=175	L+H*	H*	0
part	NEqual=176	L+H*	H*	0
of	NEqual=177	L+H*	H*	0
the	NEqual=178	L+H*	H*	0
itinerary	NEqual=179	L-	L-	!H*L-L%
just	NEqual=180	L+H*	H*	0
restate	Equal=245	L+H*	H*	H*
the	NEqual=181	L+H*	H*	0
new	NEqual=182	L+H*	H*	0
information	NEqual=183	L-L%	0	L+H* L+H*L-L%

Please,tell,me,any,comments,

Please	Equal=246	H*	0	H*
tell	Equal=247	L+H*	H*	H*
me	NEqual=184	L+H*	H*	0
any	NEqual=185	L+H*	H*	0
comments	NEqual=186	L-L%	0	L+H*L-L%

Hello,doctor,Rudnicky,Im,glad,to,hear,from,you,again,

Hello	Equal=248	H*	0	H*
doctor	Equal=249	0	0	0
Rudnicky	NEqual=187	H*	0	L-L%
Im	Equal=250	0	0	0
glad	Equal=251	L+H*	H*	H*
to	Equal=252	0	0	0
hear	Equal=253	0	0	0
from	Equal=254	0	0	0
you	Equal=255	0	0	0
again	Equal=256	L-L%	0	L-L%

This,is,already,the,latest,flight,

This	Equal=257	0	0	0
is	NEqual=188	L+H*	H*	0
already	Equal=258	L+H*	H*	H*
the	NEqual=189	L+H*	H*	0
latest	NEqual=190	L+H*	H*	!H*
flight	Equal=259	L-L%	0	L-L%

The,next,day,is,that,OK,

The	Equal=260	0	0	0
next	Equal=261	H*	0	H*
day	NEqual=191	H*	0	H*L-H%
is	Equal=262	0	0	0
that	Equal=263	0	0	0
OK	NEqual=192	H-H%	0	H*H-H%

sum of correct with pp = 263

sum of incorrect with pp = 183

Total words = 455

Correct with pp = 57.8021978021978%

Incorrect with pp = 42.1978021978022%

D.2 Prosodic Annotation with Tone Alignment

I'm,sorry,

I'm	Equal=1	L+H*	H*	H*
sorry	Equal=2	L-L%	0	L-L%

Where,are,you,leaving,from,

Where	Equal=3	0	0	0
are	Equal=4	0	0	0
you	NEqual=1	L+H*	H*	0
leaving	Equal=5	L+H*	H*	H*
from	Equal=6	L-L%	0	L-L%

What,city,are,you,leaving,from,

What	Equal=7	0	0	0
city	Equal=8	H*	0	H*
are	Equal=9	0	0	0
you	NEqual=2	L+H*	H*	0
leaving	Equal=10	L+H*	H*	H*
from	Equal=11	L-L%	0	L-L%

Are,you,a,registered,user,

Are	Equal=12	0	0	0
you	Equal=13	L+H*	H*	H*
a	Equal=14	L+H*	H*	H*
registered	Equal=15	L+H*	H*	L+H*
user	Equal=16	H-H%	0	H-H%

You,may,interrupt,these,instructions,at,any,time,
by,saying,good,enough,

You	Equal=17	H*	0	H*
may	Equal=18	0	0	0
interrupt	Equal=19	L+H*	H*	H*
these	NEqual=3	L+H*	H*	0
instructions	NEqual=4	L-	L-	L-L%
at	Equal=20	0	0	0
any	NEqual=5	L+H*	H*	0
time	NEqual=6	L-	L-	0
by	Equal=21	L+H*	H*	L+H*
saying	Equal=22	L-L%	L-	L-L%
good	Equal=23	L+H*	H*	H*
enough	Equal=24	L-L%	0	L-L%

You,can,interrupt,the,system,at,any,time,by,saying,
anything,you,wish,

You	Equal=25	H*	0	H*
can	Equal=26	0	0	0
interrupt	Equal=27	L+H*	H*	H*
the	NEqual=7	L+H*	H*	0
system	Equal=28	L+H*	H*	H*
at	Equal=29	0	0	0
any	NEqual=8	L+H*	H*	0
time	NEqual=9	L-	L-	L-L%
by	Equal=30	0	0	0
saying	Equal=31	L+H*	H*	H*
anything	Equal=32	L+H*	H*	H*
you	NEqual=10	L+H*	H*	0
wish	Equal=33	L-L%	0	L-L%

To,end,the,call,say,good,bye,

To	Equal=34	0	0	0
end	NEqual=11	0	0	H*
the	Equal=35	0	0	0
call	NEqual=12	H*	0	H*L-L%
say	Equal=36	0	0	0
good	Equal=37	L+H*	H*	H*
bye	Equal=38	L-L%	0	L-L%

This,is,the,end,of,the,instructions,

This	Equal=39	0	0	0
is	Equal=40	0	0	0
the	NEqual=13	L+H*	H*	0
end	Equal=41	0	0	0
of	Equal=42	0	0	0
the	Equal=43	L+H*	H*	L+H*
instructions	Equal=44	L-L%	0	L-L%

It,knows,about,major,US,cities,and,some,
international,destinations,

It	Equal=45	H*	0	H*
knows	Equal=46	L+H*	H*	H*
about	Equal=47	0	0	0
major	Equal=48	L+H*	H*	H*
US	NEqual=14	L+H*	H*	0
cities	NEqual=15	L-	L-	L-H%
and	Equal=49	L+H*	H*	H*
some	Equal=50	L+H*	H*	L+H*
international	Equal=51	L+H*	H*	L+H*
destinations	Equal=52	L-L%	0	L-L%

If,you,need,to,make,a,correction,just,restate,the,
new,information,

If	Equal=53	H*	0	0

you	Equal=54	H*	0	H*
need	Equal=55	L+H*	H*	H*
to	Equal=56	0	0	0
make	Equal=57	L+H*	H*	H*
a	NEqual=16	L+H*	H*	!H*
correction	Equal=58	L-L%	L-	L-L%
just	NEqual=17	L+H*	H*	0
restate	Equal=59	L+H*	H*	H*
the	Equal=60	L+H*	H*	L+H*
new	Equal=61	L+H*	H*	L+H*
information	Equal=62	L-L%	0	L-L%

If,you,need,help,at,any,time,please,say,help,

If	Equal=63	H*	0	0
you	Equal=64	H*	0	H*
need	NEqual=18	0	0	H*
help	Equal=65	L-	L-L%	L-L%
at	NEqual=19	L+H*	H*	0
any	NEqual=20	L+H*	H*	0
time	NEqual=21	L-	L-	L-L%
please	Equal=66	L+H*	H*	H*
say	Equal=67	L+H*	H*	H*
help	Equal=68	L-L%	0	L-L%

I,didnt,catch,that,

I	Equal=69	H*	0	H*
didnt	Equal=70	0	0	0
catch	Equal=71	L+H*	H*	H*
that	Equal=72	0	0	0

This,call,is,being,recorded,for,development,purposes,
and,may,be,shared,with,other,researchers,

This	Equal=73	0	0	0
call	Equal=74	H*	0	H*
is	Equal=75	0	0	0
being	Equal=76	0	0	0
recorded	Equal=77	0	0	0
for	NEqual=22	L+H*	H*	0
development	Equal=78	L+H*	H*	H*
purposes	NEqual=23	L-	L-	L-H%
and	NEqual=24	L+H*	H*	0
may	NEqual=25	L+H*	H*	0
be	NEqual=26	L+H*	H*	0
shared	NEqual=27	L+H*	H*	0
with	Equal=79	L+H*	H*	L+H*
other	Equal=80	L+H*	H*	L+H*
researchers	Equal=81	L-L%	0	L-L%

Where,do,you,want,to,go,

Where	Equal=82	0	0	0
do	Equal=83	L+H*	H*	H*
you	Equal=84	L+H*	H*	H*
want	Equal=85	L+H*	H*	H*
to	Equal=86	0	0	0
go	Equal=87	L-L%	0	L-L%

Where,would,you,like,to,go,

Where	Equal=88	0	0	0
would	Equal=89	0	0	0
you	Equal=90	L+H*	H*	H*
like	NEqual=28	0	0	H*
to	Equal=91	0	0	0
go	Equal=92	L-L%	0	L-L%

If,you,want,us,to,do,that,please,say,book,this,trip,
or,otherwise,say,continue,

If	Equal=93	H*	0	0
you	Equal=94	H*	0	H*
want	Equal=95	L+H*	H*	H*
us	NEqual=29	L-	L-	H*
to	Equal=96	0	0	0
do	NEqual=30	L+H*	H*	0
that	NEqual=31	L-L%	L-L%	0
please	NEqual=32	L+H*	H*	0
say	Equal=97	L+H*	H*	H*
book	Equal=98	L+H*	H*	H*
this	Equal=99	L+H*	H*	H*
trip	Equal=100	L-	L-L%	L-L%
or	Equal=101	L+H*	H*	H*
otherwise	Equal=102	L-	L-L%	L-L%
say	Equal=103	L+H*	H*	H*
continue	Equal=104	L-L%	L-L%	L-L%

Sorry,Im,not,sure,I,understood,what,you,said,

Sorry	NEqual=33	0	0	H*L-H%
Im	Equal=105	0	0	0
not	NEqual=34	0	0	H*
sure	Equal=106	H*	0	H*
I	Equal=107	L+H*	H*	H*
understood	Equal=108	L-	L-L%	L-L%
what	Equal=109	L+H*	H*	L+H*
you	Equal=110	L+H*	H*	H*
said	Equal=111	L-L%	0	L-L%

To,help,us,improve,our,system,please,answer,the,
following,questions,

To	Equal=112	0	0	0
help	NEqual=35	0	0	H*

us	Equal=113	H*	0	H*
improve	Equal=114	0	0	0
our	NEqual=36	L+H*	H*	0
system	NEqual=37	0	0	!H*L-H%
please	NEqual=38	L+H*	H*	0
answer	Equal=115	L+H*	H*	H*
the	Equal=116	L+H*	H*	H*
following	Equal=117	L+H*	H*	L+H*
questions	Equal=118	L-L%	0	L-L%

This,session,is,now,over,

This	Equal=119	0	0	0
session	Equal=120	H*	0	H*
is	Equal=121	0	0	0
now	Equal=122	0	0	0
over	Equal=123	0	0	0

Thank,you,for,calling,

Thank	NEqual=39	0	0	H*
you	Equal=124	H*	0	H*
for	NEqual=40	0	0	L+H*
calling	Equal=125	L-L%	0	L-L%

Are,you,satisfied,with,this,itinerary,

Are	Equal=126	H*	0	0
you	Equal=127	H*	0	H*
satisfied	Equal=128	L+H*	H*	H*
with	Equal=129	L+H*	H*	L+H*
this	Equal=130	L+H*	H*	L+H*
itinerary	Equal=131	H-H%	0	H-H%

To,go,on,I,need,you,to,answer,the,following,question,

To	Equal=132	0	0	0
go	NEqual=41	0	0	H*
on	NEqual=42	H*	0	L+H*L-L%
I	Equal=133	0	0	0
need	NEqual=43	0	0	H*
you	Equal=134	L-	L-L%	L-L%
to	NEqual=44	L+H*	H*	0
answer	Equal=135	L+H*	H*	H*
the	Equal=136	L+H*	H*	H*
following	Equal=137	L+H*	H*	L+H*
question	Equal=138	L-L%	0	L-L%

Perhaps,youre,asking,for,something,I,dont,know,about,

Perhaps	NEqual=45	0	0	H*
youre	Equal=139	0	0	0
asking	Equal=140	0	0	0
for	NEqual=46	L+H*	H*	0
something	NEqual=47	L-L%	L-L%	L+H*
I	NEqual=48	L+H*	H*	0
dont	NEqual=49	L+H*	H*	0
know	NEqual=50	L+H*	H*	0
about	Equal=141	L-L%	0	L-L%

Im,still,having,trouble,understanding,you,

Im	Equal=142	H*	0	H*
still	NEqual=51	0	0	H*
having	Equal=143	0	0	0
trouble	NEqual=52	L+H*	H*	!H*
understanding	NEqual=53	L+H*	H*	0
you	Equal=144	L-L%	0	L-L%

Sorry,I,did,not,understand,what,you,said,

Sorry	Equal=145	H*	0	H*
I	Equal=146	H*	0	H*
did	Equal=147	0	0	0
not	Equal=148	0	0	0
understand	Equal=149	0	0	0
what	Equal=150	0	0	0
you	Equal=151	0	0	0
said	Equal=152	L-L%	0	L-L%

What,is,your,full,name,

What	Equal=153	0	0	0
is	Equal=154	0	0	0
your	Equal=155	L+H*	H*	H*
full	NEqual=54	L+H*	H*	0
name	Equal=156	L-L%	0	L-L%

You,can,say,help,at,any,time,

You	Equal=157	H*	0	H*
can	Equal=158	0	0	0
say	Equal=159	0	0	0
help	Equal=160	0	0	0
at	Equal=161	0	0	0
any	Equal=162	0	0	0
time	NEqual=55	0	0	L-L%

Which,one,did,you,want,

Which	Equal=163	H*	0	H*
one	Equal=164	H*	0	H*
did	Equal=165	L+H*	H*	H*
you	NEqual=56	L+H*	H*	0
want	Equal=166	L-L%	0	L-L%

When,youre,having,trouble,please,ask,for,help,

When	Equal=167	0	0	0

youre	Equal=168	0	0	0
having	Equal=169	L+H*	H*	H*
trouble	NEqual=57	L-	L-	!H*L-L%
please	NEqual=58	L+H*	H*	0
ask	Equal=170	L+H*	H*	H*
for	NEqual=59	L+H*	H*	0
help	Equal=171	L-L%	0	L-L%

Please,be,our,guest,

Please	Equal=172	H*	0	H*
be	Equal=173	0	0	0
our	Equal=174	L+H*	H*	H*
guest	Equal=175	L-L%	0	L-L%

You,can,ask,me,for,help,at,any,time,

You	Equal=176	H*	0	H*
can	Equal=177	0	0	0
ask	Equal=178	L+H*	H*	H*
me	Equal=179	L-L%	L-	L-L%
for	Equal=180	L+H*	H*	H*
help	Equal=181	L-	L-L%	L-L%
at	NEqual=60	L+H*	H*	0
any	NEqual=61	L+H*	H*	0
time	Equal=182	L-L%	0	L-L%

Would,you,like,me,to,summarize,your,trip,

Would	Equal=183	H*	0	H*
you	Equal=184	H*	0	H*
like	Equal=185	L+H*	H*	H*
me	NEqual=62	L+H*	H*	0
to	NEqual=63	L+H*	H*	0
summarize	Equal=186	L+H*	H*	H*
your	NEqual=64	L+H*	H*	0
trip	Equal=187	H-H%	0	H-H%

When,youre,finished,just,hang,up,

When	Equal=188	0	0	0
youre	Equal=189	0	0	0
finished	NEqual=65	L+H*	H*	H*L-H%
just	NEqual=66	L+H*	H*	0
hang	Equal=190	L+H*	H*	H*
up	Equal=191	L-L%	0	L-L%

Thank,you,for,using,the,Carnegie,Mellon,Communicator,

Thank	NEqual=67	0	0	H*
you	Equal=192	H*	0	H*
for	Equal=193	0	0	0
using	Equal=194	0	0	0
the	NEqual=68	L+H*	H*	!H*
Carnegie	Equal=195	L+H*	H*	L+H*
Mellon	Equal=196	L+H*	H*	L+H*
Communicator	Equal=197	L-L%	0	L-L%

Ive,made,no,hotel,reservations,for,your,trip,

Ive	Equal=198	0	0	0
made	NEqual=69	0	0	H*
no	NEqual=70	L+H*	H*	0
hotel	NEqual=71	L+H*	H*	0
reservations	NEqual=72	L-	L-	L-H%
for	Equal=199	L+H*	H*	H*
your	Equal=200	L+H*	H*	H*
trip	Equal=201	L-L%	0	L-L%

Please,say,something,

Please	Equal=202	H*	0	H*
say	Equal=203	L+H*	H*	H*
something	Equal=204	L-L%	0	L-L%

Otherwise,I,will,need,to,hang,up,

Otherwise	Equal=205	H*	0	H*
I	Equal=206	H*	0	H*
will	Equal=207	0	0	0
need	Equal=208	0	0	0
to	Equal=209	0	0	0
hang	Equal=210	0	0	0
up	Equal=211	0	0	0

Were,you,attempting,to,arrange,travel,that,you,
actually,plan,to,take,

Were	Equal=212	0	0	0
you	NEqual=73	0	0	H*H*
attempting	Equal=213	0	0	0
to	Equal=214	0	0	0
arrange	Equal=215	0	0	0
travel	NEqual=74	0	0	H*L-L%
that	Equal=216	0	0	0
you	NEqual=75	0	0	H*H*
actually	NEqual=76	0	0	H*
plan	Equal=217	0	0	0
to	Equal=218	0	0	0
take	Equal=219	H-H%	0	H-H%

Should,I,summarize,your,trip,

Should	Equal=220	H*	0	H*
I	Equal=221	H*	0	H*
summarize	Equal=222	0	0	0
your	Equal=223	0	0	0
trip	Equal=224	H-H%	0	H-H%

Is,that,correct,

Is	Equal=225	0	0	0
that	Equal=226	0	0	0
correct	Equal=227	H-H%	0	H-H%

Which,city,and,state,did,you,want,

Which	Equal=228	H*	0	H*
city	Equal=229	H*	0	H*
and	Equal=230	H*	0	0
state	Equal=231	H*	0	0
did	Equal=232	0	0	0
you	Equal=233	0	0	0
want	Equal=234	L-L%	0	L-L%

Ive,made,no,car,reservations,for,this,trip,

Ive	Equal=235	0	0	0
made	NEqual=77	0	0	H*
no	NEqual=78	L+H*	H*	0
car	NEqual=79	L+H*	H*	0
reservations	NEqual=80	0	0	!H*L-H%
for	Equal=236	0	0	0
this	NEqual=81	0	0	H*
trip	Equal=237	L-L%	0	L-L%

Im,sorry,I,couldnt,find,your,profile,

Im	Equal=238	H*	0	H*
sorry	NEqual=82	L+H*	H*	H*L-H%
I	NEqual=83	L+H*	H*	0
couldnt	NEqual=84	L+H*	H*	0
find	NEqual=85	L-	L-	H*
your	Equal=239	L+H*	H*	L+H*
profile	Equal=240	L-L%	0	L-L%

Will,you,return,to,Pittsburgh,from,Seattle,

Will	NEqual=86	0	0	H*
you	Equal=241	H*	0	H*
return	Equal=242	0	0	0
to	Equal=243	0	0	0
Pittsburgh	Equal=244	L+H*	H*	H*
from	Equal=245	L+H*	H*	L+H*
Seattle	Equal=246	H-H%	0	H-H%

There,is,currently,no,specific,help,for,this,topic,

There	Equal=247	H*	0	H*
is	Equal=248	0	0	0
currently	NEqual=87	L-	L-	H*
no	NEqual=88	L+H*	H*	0
specific	NEqual=89	L+H*	H*	!H*
help	Equal=249	L-	L-L%	L-L%
for	NEqual=90	L+H*	H*	0
this	Equal=250	L+H*	H*	H*
topic	Equal=251	L-L%	0	L-L%

Or,you,can,go,on,by,answering,the,question,

Or	Equal=252	H*	0	0
you	Equal=253	H*	0	H*
can	Equal=254	0	0	0
go	Equal=255	0	0	0
on	Equal=256	L+H*	H*	L+H*
by	NEqual=91	L+H*	H*	0
answering	Equal=257	L+H*	H*	H*
the	Equal=258	L+H*	H*	L+H*
question	Equal=259	L-L%	0	L-L%

I,might,feel,better,then,

I	Equal=260	H*	0	H*
might	Equal=261	0	0	0
feel	Equal=262	0	0	0
better	Equal=263	L+H*	H*	H*
then	Equal=264	L-L%	0	L-L%

Which,destination,did,you,want,

Which	Equal=265	H*	0	H*
destination	Equal=266	H*	0	H*
did	NEqual=92	0	0	H*
you	Equal=267	0	0	0
want	Equal=268	L-L%	0	L-L%

Something,is,wrong,with,the,flight,retrieval,

Something	Equal=269	H*	0	H*
is	Equal=270	0	0	0
wrong	Equal=271	0	0	0
with	NEqual=93	L+H*	H*	0
the	Equal=272	0	0	0
flight	Equal=273	L+H*	H*	L+H*
retrieval	Equal=274	L-L%	0	L-L%

Please,tell,us,any,comments,

Please	Equal=275	H*	0	H*
tell	Equal=276	L+H*	H*	H*
us	NEqual=94	L-	L-	0
any	Equal=277	L+H*	H*	L+H*
comments	Equal=278	L-L%	0	L-L%

I,have,a,best,Western,

I	Equal=279	H*	0	H*
have	Equal=280	0	0	0

a	Equal=281	0	0	0
best	Equal=282	L+H*	H*	L+H*
Western	Equal=283	L-L%	0	L-L%

At,this,point,you,have,selected,a,leg,of,your,itinerary,

At	Equal=284	0	0	0
this	Equal=285	0	0	0
point	NEqual=95	0	0	H*
you	Equal=286	H*	0	0
have	Equal=287	0	0	0
selected	Equal=288	0	0	0
a	Equal=289	0	0	0
leg	Equal=290	L+H*	H*	H*
of	Equal=291	L+H*	H*	L+H*
your	Equal=292	L+H*	H*	L+H*
itinerary	Equal=293	L-L%	0	L-L%

What,time,do,you,need,to,depart,

What	Equal=294	0	0	0
time	Equal=295	L+H*	H*	H*
do	NEqual=96	0	0	H*
you	Equal=296	0	0	0
need	NEqual=97	L+H*	H*	0
to	Equal=297	0	0	0
depart	Equal=298	L-L%	0	L-L%

Im,sorry,we,dont,have,your,profile,yet,

Im	NEqual=98	0	0	H*
sorry	NEqual=99	0	0	H*L-H%
we	Equal=299	0	0	0
dont	Equal=300	0	0	0
have	Equal=301	0	0	0
your	Equal=302	0	0	0
profile	NEqual=100	0	0	L+H*
yet	Equal=303	0	0	0

I,have,a,marriott,

I	Equal=304	H*	0	H*
have	Equal=305	0	0	0
a	Equal=306	L+H*	H*	L+H*
marriott	Equal=307	L-L%	0	L-L%

Hello,Wei,its,nice,to,hear,from,you,again,

Hello	Equal=308	H*	0	H*
Wei	NEqual=101	H*	0	H*L-H%
its	Equal=309	0	0	0
nice	Equal=310	L+H*	H*	H*
to	Equal=311	0	0	0
hear	Equal=312	0	0	0
from	Equal=313	0	0	0
you	Equal=314	0	0	0
again	Equal=315	L-L%	0	L-L%

To,correct,any,part,of,the,itinerary,just,restate,
the,new,information,

To	Equal=316	0	0	0
correct	Equal=317	L+H*	H*	H*
any	NEqual=102	L+H*	H*	0
part	NEqual=103	L+H*	H*	0
of	NEqual=104	L+H*	H*	0
the	NEqual=105	L+H*	H*	!H*
itinerary	Equal=318	L-	L-L%	L-L%
just	NEqual=106	L+H*	H*	0
restate	Equal=319	L+H*	H*	H*
the	Equal=320	L+H*	H*	L+H*
new	Equal=321	L+H*	H*	L+H*
information	Equal=322	L-L%	0	L-L%

Please,tell,me,any,comments,

Please	Equal=323	H*	0	H*
tell	Equal=324	L+H*	H*	H*
me	NEqual=107	L+H*	H*	0
any	Equal=325	L+H*	H*	L+H*
comments	Equal=326	L-L%	0	L-L%

Hello,doctor,Rudnicky,Im,glad,to,hear,from,you,again,

Hello	Equal=327	H*	0	H*
doctor	Equal=328	0	0	0
Rudnicky	NEqual=108	H*	0	L-L%
Im	Equal=329	0	0	0
glad	Equal=330	L+H*	H*	H*
to	Equal=331	0	0	0
hear	Equal=332	0	0	0
from	Equal=333	0	0	0
you	Equal=334	0	0	0
again	Equal=335	L-L%	0	L-L%

This,is,already,the,latest,flight,

This	Equal=336	0	0	0
is	NEqual=109	L+H*	H*	0
already	Equal=337	L+H*	H*	H*
the	NEqual=110	L+H*	H*	0
latest	NEqual=111	L+H*	H*	!H*
flight	Equal=338	L-L%	0	L-L%

The,next,day,is,that,OK,

The	Equal=339	0	0	0
next	Equal=340	H*	0	H*

day	NEqual=112	H*	0	H*L-H%	OK	Equal=342	H-H%	0	H-H%
is	Equal=341	0	0	0					
that	NEqual=113	0	0	H*					

sum of correct alignment = 342

sum of incorrect minus alignment = 113

Total words = 455

Correct with alignment = 75.1648351648352%

Incorrect with alignment = 24.8351648351648%