Lecture Notes in Artificial Intelligence 1307

Subseries of Lecture Notes in Computer Science
Edited by J. G. Carbonell and J. Siekmann

Lecture Notes in Computer Science
Edited by G. Goos, J. Hartmanis and J. van Leeuwen

Ralf Kompe

Prosody in Speech Understanding Systems

Springer

Series Editors
Jaime G. Carbonell, Carnegie Mellon University, Pittsburgh, PA, USA
Jörg Siekmann, University of Saarland, Saarbrücken, Germany

Author

Ralf Kompe
Sony International (Europe) GmbH
European Research and Development Stuttgart
Stuttgarter Straße 106, D-70736 Fellbach, Germany
E-mail: kompe@fb.sony.de

Cataloging-in-Publication Data applied for

Die Deutsche Bibliothek - CIP-Einheitsaufnahme

Kompe, Ralf:
Prosody in speech understanding systems / Ralf Kompe. - Berlin ; Heidelberg ; New York ; Barcelona ; Budapest ; Hong Kong ; London ; Milan ; Paris ; Santa Clara ; Singapore ; Tokyo : Springer, 1997
(Lecture notes in computer science ; 1307 : Lecture notes in artificial intelligence)
ISBN 3-540-63580-7

This book also constitutes the author's doctoral thesis accepted by the Technical Faculty of the University of Erlangen-Nürnberg, Germany, in 1996.

CR Subject Classification (1991): I.2.7

ISBN 3-540-63580-7 Springer-Verlag Berlin Heidelberg New York

Typesetting: Camera ready by author
SPIN 10545882 06/3142 - 5 4 3 2 1 0 Printed on acid-free paper

Foreword

Speech technology, the automatic processing of (spontaneously) spoken words and utterances, now is known to be technically feasible and will become the major tool for handling the confusion of languages. The economic implications of this tool are obvious, in particular in the multilingual European Union. Potential and current applications are dictation systems, information retrieval by spoken dialog, and speech–to–speech translation.

The first step of automatic processing is done by an acoustic front end which so far almost exclusively evaluate the acoustic–phonetic information contained in the speech signal in order to perform word recognition. It is well known that in addition to the acoustic–phonetic information the speech signal contains prosodic information (for example, about intonation, accentuation, prosodic phrases, or hesitation) which is used in human–human communication. Clearly, this should also be used in advanced systems for man–machine communication, and this book is devoted to the problem of exploiting prosodic information.

The book gives a thorough account of prosodic phenomena, the adequate prosodic labeling of speech corpora for training classifiers, neural networks (NN), hidden Markov models (HMM), and their combination to NN/HMM hybrids for classification of sequences of prosodic feature vectors, as well as semantic classification trees and stochastic language models for the classification of prosodic events on the basis of word sequences. New algorithms for the use of prosodic information in speech understanding are described and explained using many examples from real–life data. The tremendous advantage of using prosodic information for parsing of spontaneously spoken utterances is demonstrated for the German Verbmobil speech–to–speech translation system. In addition it is described how accent information can be used for the disambiguation of the focus of an utterance and for the translation of particles, and how an utterance can be segmented into dialog acts. Furthermore, sentence mood classification was integrated into a dialog system for information retrieval.

The results prove that in spontaneous speech there are strong prosodic regularities so that prosodic information can be used in a speaker–independent system to support speech understanding and dialog control. It can be concluded that without prosody the automatic interpretation of spontaneous speech would be infeasible. It is for the first time that prosody was fully integrated into an operational speech understanding and translation system and that its usefulness was shown in experiments on large databases.

Erlangen, July 1997 *H. Niemann*

Preface

The research effort in the field of automatic speech understanding has been increasing a lot in recent years. Current prototype systems for information retrieval dialog or speech–to–speech translation show that technology is not far from being used in products for restricted application domains. Nevertheless, there is still much research to be done before automatic systems reach the capabilities of humans. For example, it is widely known that in human–human communication prosodic information contained in the speech signal plays an important role on many levels of speech recognition and, especially, interpretation. However, so far only little research has been done with respect to the use of prosody in speech understanding systems. Prosody covers acoustic phenomena of speech which are not specific to phonemes. These are mainly intonation, indicators for phrase boundaries, and accentuation. This information can support the intelligibility of speech or even sometimes disambiguate the meaning.

The aim of this book is to describe algorithms developed by the author for the use of prosodic information on many levels of speech understanding such as syntax, semantics, dialog, and translation. An implementation of these algorithms has been integrated into the speech–to–speech translation system VERBMOBIL and in the dialog system EVAR, although the algorithms are not specific for use in these systems. The VERBMOBIL and EVAR prototypes were used to evaluate the approaches on large databases; a summary of the results is presented. The emphasis of the book lies on the improvement of parsing of spontaneous speech with the help of prosodic clause boundary information. Presently, parsing is the field of speech understanding where prosodic information can most successfully be used. This research was conducted by the author while he was with the Institute for Pattern Recognition at the University of Erlangen–Nürnberg. Its technical faculty has also accepted this book as dissertation.

The first part of the book gives a comprehensive review of the mathematical and computational background of the algorithms and statistical models useful for the integration of prosody in speech understanding. It also shows unconventional applications of hidden Markov models, stochastic language models, and neural networks. The latter, for example, apart from several classification tasks, are used for the inverse filtering of speech signals. The book also explains in detail the acoustic–prosodic phenomena of speech and their functional role in communication. In contrast to many other reports, it gives a lot of examples taken from real human–human dialogs; many examples are supported by speech signals accessible over the WORLD WIDE WEB. The use of prosodic information relies on the robust extraction of relevant features from digitized speech signals, on adequate labeling of large speech databases for training classifiers, and on the detection of prosodic events; the methods used in VERBMOBIL and EVAR are summarized as well in this book. Furthermore, an overview of these state–of–the–art speech understanding systems is given.

The book is addressed to engineers and students interested in speech understanding or in prosody. It also addresses linguists and phoneticians who want to get a summary of the state–of–the–art in research on prosody of spontaneous speech and its application. Parts of the book require some background in statistics, computer science and pattern recognition.

Speech is a very complex phenomenon which is still only partly understood. A few years ago P. Ladefoged wrote what might seem to be self–evident but in fact is a serious remark:

> *Nobody . . . could know all that there is to know about speech research. The only way in which one can work in this field is to have good friends. We all have to rely on other people to fill in the gaps — the vast holes — in our knowledge. Any scientist today is part of a team that cannot hope to build a bridge into the future without a lot of help. This is clearly so in my case.* [Lad92]

This also has clearly been the case for the research presented in this book. Without the cooperation of a lot of colleagues within my institute and from other institutes participating in the VERBMOBIL project it would have been impossible to carry out this research. It is a pleasure to acknowledge their effort.

I wish to thank my colleagues in the VERBMOBIL project for the close cooperation, especially, Gabi Bakenecker, Dr. Hans–Ulrich Block, Johan Bos, Thomas Bub, Rudi Caspari, Anke Feldhaus, Pablo Fetter, Stefan Geißler, Prof. Dr. Wolfgang Hess, Dr. Alfred Kaltenmeier, Thomas Kemp, Ute Kilian, Dr. Tibor Kiss, Dr. Thomas Kuhn, Martin Neumann, Dr. Peter Regel–Brietzmann, Matthias Reyelt, Tobias Ruland, and Stefanie Schachtl for the joint effort in elaborating approaches for the use of prosody in the VERBMOBIL system and in performing experiments, as well as for many hints, ideas and discussions concerning my work.

I am particularly deeply indebted to Dr. Andreas Kießling and Dr. Anton Batliner with whom I worked together very closely for many years. This teamwork proved to be very effective.

I also wish to thank my colleagues at the institute of pattern recognition at the University of Erlangen–Nürnberg, who supported my work with a steady openness for discussion, who helped me to approach certain problems from a different point of view, and who made sure that work was fun most of the time. Furthermore, many students supported me in software development and experimental work; especially, I would like to acknowledge the work of Joachim Denzler, Julia Fischer, Richard Huber, Bernhard Kahles, Karin Ott, Volker Warnke, and Andreas Zottmann.

Part of the software which was used for the experiments described in this book was kindly provided by the following colleagues: First of all I have to thank Prof. Dr. Günter Schukat–Talamazzini, who permitted me to use the polygram software in the prosody module and to integrate it in the VERBMOBIL system. From Dr. Roland Kuhn we received our initial version of SCT programs; these were further developed to MSCTs by Stefan Harbeck and Arnd Gebhard. Dr. Thomas Kuhn changed the interface to his word recognizer so that it could be used within the prosody module. I am especially grateful that all of them helped me many times, when problems occurred while using or misusing these programs. I also whish to thank FORWISS Erlangen who permitted me to use the neural network simulator FAST.

I extend many thanks to Dr. Anton Batliner, Joachim Denzler, Dr. Roland Kuhn, Dr. Thomas Kuhn, and Dr. Elmar Nöth for their close reading of the manuscript and the many helpful comments. Since the book touches many aspects of current speech technology, I am grateful to Johan Bos, Dr. Wieland Eckert, Julia Fischer, Florian Gallwitz, Dr. Joachim Hornegger, Dr. Andreas Kießling, and Dr. Marion Mast, who helped to improve the parts of the manuscript that interfere with their field of expertise. I wish to thank also Alfred Hofmann and Prof. Dr. Jörg Siekmann for their support in publishing this book.

Part of the research presented in this book would have been impossible without the funding by the German Federal Ministry of Education, Science, Research, and Technology (*BMBF*) in the framework of the ASL and VERBMOBIL projects.

I am particularly deeply indebted to Prof. Dr. Heinrich Niemann, Prof. Dr. Renato de Mori and Dr. Elmar Nöth who contributed a lot to my education in the fields of pattern recognition and artificial intelligence and who supported my research with many ideas and advice. I also wish to thank Prof. Dr. Heinrich Niemann and Prof. Dr. Renato de Mori for spending their time on the review of my doctoral thesis.

Last but not least, my wife, Petra, who patiently endured my absence from private life for many months, deserves my special thanks.

Fellbach, July 1997 *Ralf Kompe*

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The aim of this chapter is first to motivate the need for automatic speech understanding (ASU) in general by naming available and potential applications. Second, the basic structure of ASU systems is described. Third, the term prosody will be defined and it will be argued that prosodic information should be used by speech processing systems, and the difficulties involved will be described. Finally, an overview of this book will be given and the contribution of the presented research results to the progress of science will be outlined.

1.1 Automatic Speech Understanding (ASU)

1.1.1 Applications

In the past decade computers became more and more important. In many areas such as flight reservation, design of technical devices, book–keeping, or typewriting they are already irreplaceable. Also a high percentage of private homes are "equipped" with personal computers. It can be foreseen that the use of computers will even increase in the near future. In the information society high speed networks ("data highways") will allow for applications like database inquiries, for example, train time table information, interactive TV, all kinds of reservation tasks, for example, in the case of hotels, and customer advice service. A consequence of the progress in robotics and computer vision is that service robots, for example, for disabled people, are not anymore a topic of science fiction.

In all areas like these where humans use computers there is some form of human–machine interaction. Currently, in almost all cases the only means for this interaction are keyboard, mouse, and touch screen. There are good reasons for adding speech as another modality of man–machine communication:

- Speech is the most natural way of human–human communication, thus it would be most convenient if computers could be accessed via speech. Many people are not familiar with typing, so for applications like reservation, which will be used by a wide variety of end users, speech is the best way of communication.
- Speech is the most efficient way of communication: on the average humans speak at a rate of 120 to 250 words per minute; even well trained typists only achieve a rate of 100 to 150 words per minute [Ain88].
- In many applications it is necessary or at least helpful if people can do "hands–free" operation of machines or devices, which means communication by speech. For example, the operation of radio or telephone in a car by speech would contribute a lot to safety.
- It is preferable to access information services like train time tables, or weather reports via telephone because then they become available almost everywhere.
- For many disabled the only possibility to interact with machines is by speech.
- New application fields for computers could be opened to allow for rationalization in industry. For example telephone operators could be replaced easily by automatic call centers if ASR were reliable and robust.
- The increasing international cooperation in politics, economy and so on raises the wish for speech–to–speech machine translation.

There is still a lot of research to be done until the performance of *automatic speech understanding* (ASU) systems reaches (at least) human capabilities, so that it can be used in all kinds of applications. Research activities concern spontaneous speech, statistical modeling of syntax and semantics, microphone arrays, joint optimization of the different modules of an ASU system, and the use of prosody. The research presented in this book concerns with the use of prosody in ASU, which is an important carrier of information in speech communication though it has long been neglected. Prosody is a cover term for speech phenomena like intonation or accentuation. Before we have a closer look at prosodic phenomena, the next two sections will give an overview of the general technological requirements of different ASR or ASU applications.

1.1.2 Application Specific Requirements

In the various application fields different requirements for speech processing systems exist; an overview can be found in [Rab95]. The relatively simple task of *isolated speech recognition* with small vocabulary sizes (a few hundred words), which means that words or standard phrases are spoken in isolation, is sufficient for the user interface of most computer programs, for example, in the field of computer aided design. Another application is the speech recognizer used by North American telephone companies, which is used to recognize if questions like Will you accept the call? are answered with yes or no [Len90]. If the noise level is not too high, this problem can be viewed as solved [Kuh95b]. A similar system is used by the Deutsche Postbank, where customer calls are transferred to a specific service, for example, transactions, based on a hierarchy of yes/no questions. Another isolated speech (speaker–dependent) recognizer on the market is the *Voice-Dialing* System from NYNEX Science & Technology, which recognizes a small and fixed set of spoken commands like repeat dialing, with which a telephone can be operated [Asa95]. The next level of speech recognizers allows for several words spoken in a row with small pauses between the words. This technique is called *connected speech recognition*; it is used in large vocabulary (a few 10,000 words) dictation systems when real–time performance is required. Examples are the systems of Dragon (Dragon Dictate [Bar95]) and IBM (VoiceType Dictation [Sch96b]), which have been available on the market for a few years. However, dictation systems, which will increase the efficiency of the job of typists, should be able to perform *continuous speech recognition*, where speakers can utter many words without pausing between the words; in many application fields for dictation such as the creation of newspaper articles or radiology reports large vocabularies are required [Pau92, Laz94]. In these applications the same speech recognizer will usually be used only by one person. Thus *speaker–dependent speech recognition* is acceptable, which means that in an initialization phase the parameters of the recognizer are adapted to the speaker. Recently, Philips brought a dictation system for continuous speech on the market (Speech Magic, cf. [Dug95]), which does not perform in real–time but records the speech first and conducts the recognition in a batch processing step; as for examples of other large vocabulary, continuous speech recognition systems cf. [Mur93, Ney94b, Nor94a, Boc95, Woo95, Gau95].

With respect to these dictation systems, speech processing becomes more difficult in tasks like database access or flight reservation. In these the spoken utterance has not only to be recognized, but the user's intention has to be "understood", which essentially means that a recognized word sequence has to be structured into syntactic phrases, which then have to be mapped onto structures representing the semantic and pragmatic information contained in the utterance. These build the basis for the machine to take the intended action, for example, the database request. In addition to these *speech understanding* capabilities such a system should have a module for *dialog control*, which among other tasks has to keep track of the dialog history to enable anaphora resolution and to initiate dialogs to clarify under-specified user utterances. Furthermore, such systems have to deal with *spontaneous speech* phenomena, which are for example ungrammatical word order, *em*'s, false–starts, repetitions, cf. [Wei75, O'S92b, O'S93, Tro94]. In many cases database access will be done over the telephone, for example, in the case of train time table inquiries, which causes further difficulties to speech recognition, because of the small bandwidth and the variety of acoustic channel characteristics. The recognizers of such systems of course have to work speaker–independently. Prototypical spoken dialog systems are described in [Aus94, Aus95, Bri94, Dal94a, Pec94, Eck93, Eck96] and Section 3.3; it will probably need a few more years of research to make such systems more robust and flexible, so that they can be used commercially.

In dialog systems for database access a rough interpretation is usually sufficient. In contrast, in *speech–to–speech translation* the utterances should be translated almost literally. As a consequence the semantic and pragmatic interpretation has to capture all fine aspects of speech or language. This also implies a translation of the semantic level structures, which represent the utterance, rather than for example a direct translation of word groups. These translated semantic structures represent the utterance in the target language. They build the basis for the generation of a word sequence annotated with prosodic markers, which eventually will be output as synthesized speech. Research in the field of speech translation started recently [Wah93, Wai95, Suz95, Wai96].

Service robots should not only have flexible spoken dialog capabilities, but they should be able to integrate vision, for example, gestures [Kh95], and speech input in a concurrent and synergetic manner [Bou95a] in the understanding process. Little research has been done so far towards this application, for examples, cf. [DM90, Pat95, Ahl96].

1.1.3 Basic ASU System Structure

Since the major topic of this book is the use of prosodic information in ASU systems, we will give an overview of the basic structure of state–of–the–art systems. Speech understanding is a complex problem [Moo94], thus ASU systems are decomposed into several modules. A typical structure is depicted in Figure 1.1. An extended overview of the technologies used in ASU systems can be found in

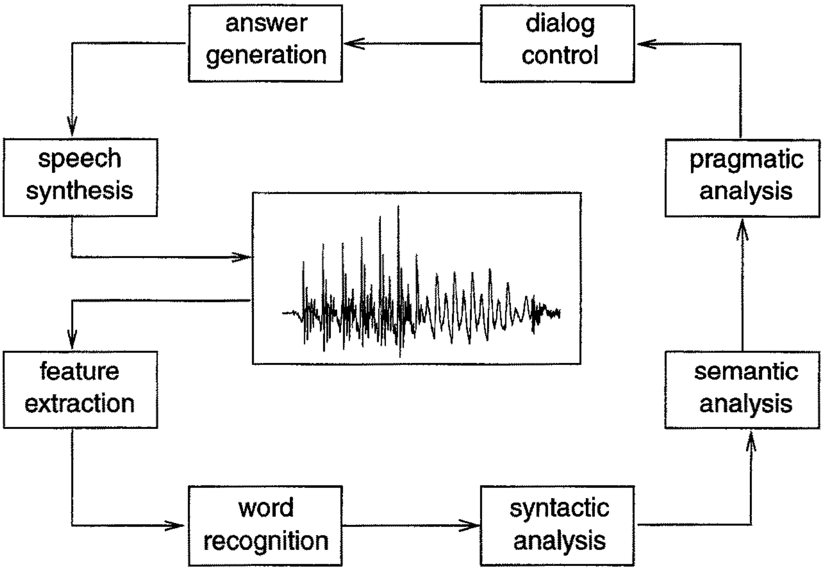


Figure 1.1: Typical structure of ASU systems.

[DM95].

During *feature extraction* the speech signal is decomposed into sequences of short frames of constant length (often 10 msec). Each frame is mapped onto a heuristic feature vector which encodes spectral information [Rab93].

The *word recognizer* consists of an acoustic model, realized by hidden Markov models (HMMs), and a rather simple stochastic language model [Rab93]. The acoustic model specifies the probabilities of sequences of the feature vectors given a sequence of words. The language model consists either of rules specifying correct word sequences or of probabilities for word sequences. To cope with recognition errors, more than one word is hypothesized at a time. In the case of continuous speech, these are output either as n-best word chains or as word graphs; the latter allow for a much more efficient subsequent analysis, and can be viewed as a compact representation of the n-best word chains. An example of an automatically computed word graph is given in Figure 1.2; the nodes are depicted in temporal order.

The task of the *syntactic analysis* is to find the optimal word chain with respect to word recognition scores and to syntactic constraints and to determine the phrase structure underlying the word chain. Note that without higher level knowledge more than one segmentation of the same word chain into phrases is possible, cf. below. Most systems rely on hand written grammar rules and some kind of parser, cf. [Han95, Aus94, Sen95, War91].

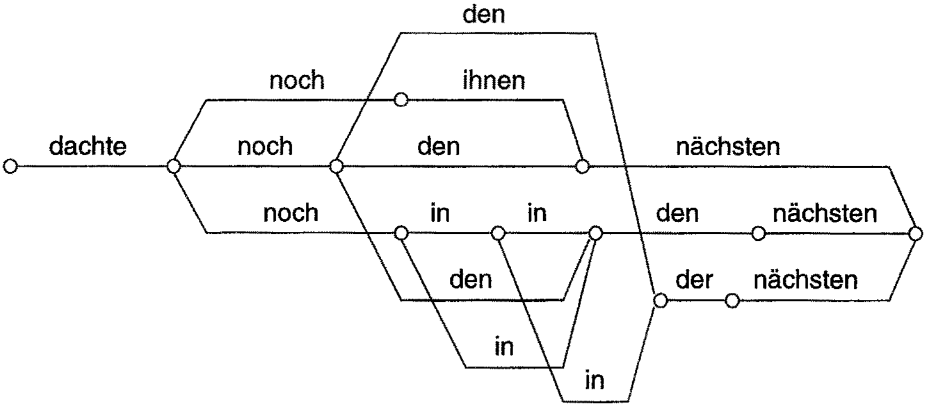


Figure 1.2: Part of an automatically computed word graph.

The *semantic analysis* has to perform an interpretation of the utterance based on the phrase structure(s) determined by the syntax module but independently from the application domain.

Certain expressions like deictic terms, which refer to objects or people, can only be interpreted correctly by taking into account the situative context or the application domain. This is done by the *pragmatic analysis*.

Semantic and pragmatic modules in ASU systems usually are knowledge–based, examples can be found in [Bri94, Aus94, War91, Pec94, Eck96]). Note that recently stochastic approaches for speech understanding have been proposed [Pie95, Kuh95a, Vid94], which are trainable models integrating syntactic, semantic, and pragmatic knowledge.

The *dialog control* keeps track of the dialog history, for example, for the purpose of anaphora resolution, determines the dialog act of the user utterances, for example, greeting, and initiates system responses, for example, answers or clarifying questions. These system responses are generated as *natural language* and output via a *speech synthesis* tool.
In general, problems for a *speech recognizer* are

- acoustic variabilities, even when the same phone is uttered twice by the same speaker,
- phones or words which are likely to be confused,
- continuity, which refers to the fact that word boundaries in general have no acoustic correlate,
- coarticulation and slurring,
- noise and channel characteristics,

- out of vocabulary words,
- word interruptions,
- word repetitions, and
- non–verbals like cough, laughter, and lip smacks.

The *speech understanding* has to cope with

- word recognition errors and/or alternative word hypotheses,
- words not contained in the lexicon,
- ungrammatical word order,
- false–starts and repetitions of phrases,
- anaphora,
- under-specified utterances,
- ambiguities, and
- missing punctuation, when no prosodic information is available.

Ambiguities and under-specification can only be resolved by integrating knowledge from different levels, which is not done in most systems developed so far. Examples for such ambiguities are the attachment of prepositional phrases, like

(He) (shot (the man) with the gun) (1.1)

versus

(He) (shot) (the man with the gun) (1.2)

or the scope of quantifiers such as in

I need a seat for five people (1.3)

versus

I need a compartment for five people (1.4)

Example (1.1) seems to be preferable over (1.2) if one has the knowledge humans have about the words to shoot and gun. However, machines often do not have such a knowledge thus both cases might be reasonable. This might become clearer when considering example (1.5) versus (1.6):

(He) (shot) (the man) with the sword (1.5)

(He) (shot) (the man with the sword) (1.6)

Up to now, ASU systems are very limited with respect to the linguistic coverage and the application domain; for example, in the case of train time table information or appointment scheduling, the vocabulary size usually is only a few thousand words. One reason for this is that in most systems nowadays only a small interaction between the modules is realized. In most systems speech is processed sequentially in a bottom–up manner by the different modules, as for example, in the SUNDIAL [Pec91] or the VERBMOBIL (Section 3.4) system. For simplification this is also indicated in Figure 1.1; however, a high degree of interaction and a mixture of bottom–up and top–down analysis would be preferable in view of the problems indicated above. An example is presented in Section 3.3.

1.2 Prosody in ASU

As for the many different definitions of prosody versus intonation, suprasegmentals, etc. cf. [Kie97, Chaps. 1 and 3] We will use the term *prosody* in a rather broad sense: it covers properties of speech which are related to speech segments larger than phones, like syllables, words, or phrases [Buß90]. Such properties are pitch, relative duration, energy, pauses, voice quality, accent and intonation [Nöt91a, Chap. 2]. This section describes the potential use of prosody in ASU and the principal difficulties when integrating this information in current ASU systems. For a description of the different aspects of prosody and its role in communication cf. Chapter 4.

1.2.1 Potentials

Prosody can potentially contribute to the analysis in all levels of ASU systems. The following summary draws on [Leh70, Lea80a, Vai88, Nöt91a].

The formant frequencies which are relative maxima in the spectrum being characteristic for the different phones are influenced by the pitch level of the speaker. This knowledge could be used in *feature extraction* for word recognition, which usually is based on a spectral analysis of the speech signal.

In *word recognition* knowledge about the syllable carrying the lexical accent could be used, because in German as in many other languages the position of the accent within a word is rather fixed. The acoustic properties of accented syllables differ somewhat from other syllables. The word level disambiguation by the accent position is important for English, for example, permit versus permit, but it only plays a minor role in German. Other prosodic information which might be useful in word recognition are the characteristic patterns of relative duration along the phones of a word. Word boundaries are sometimes marked by laryngealizations, especially if they are ambiguous as in da ich (*/da:IC/, because I*) versus Deich (*/daIC/, dike*), to which the same phoneme sequence corresponds. Throughout this book phonemes are given in SAMPA notation, cf. [Fou89, pp. 141–159] and Appendix A.3.

Syntactic analysis has the problem to structure speech into phrases. This is a highly ambiguous problem. In written language phrase boundaries are partly marked by punctuation as the two following examples demonstrate [Nöt91a]:

John, where James had had "had", had had "had had"; "had had" had been correct. (1.7)

versus

John, where James had had "had had", had had "had"; "had had" had been correct. (1.8)

When the punctuation were omitted in these examples, in the first place they would be difficult to understand, and in the second place a disambiguation would be impossible between the two cases. In speech prosodic boundary markers can take over the role of punctuation. Thus, prosodic information can disambiguate the phrase structure and it can support the analysis by grouping words. Even small phrases not marked by punctuation in written language are in speech sometimes separated by prosodic means.

For the *semantic* and *pragmatic interpretation*, the position of the focus of sentences is a central aspect. The focus is usually defined as the new versus the given or the important information. In speech the focus often is marked by accents. For example, in Paul is sleeping the important information is that it is Paul who sleeps; whereas in Paul is sleeping the speaker wants to express that what Paul does right now is sleeping.

The *dialog control* has to classify dialog acts. Especially in spontaneous speech elliptic clauses like fifteen. or fifteen? are very frequent. In these cases the sentence mood and therefore the dialog act can only be determined by intonation.

The *transfer* and *generation* modules in speech–to–speech translation systems should know about emphatic accents in the source speech signal to transfer them into the generated speech. Since emphatic accents are paralinguistic phenomena, they are not modeled by semantics or pragmatics, but they carry important information about the emotional state of the speaker, which should also be transferred to the target speech. For example, if a person is touched emotionally by some information he could utter the sentence This really is unbelievable by putting extraordinary strong accent on the two underlined words.

The *speech synthesis* in translation systems should adapt to the speaker. With respect to prosody this means that the pitch level, which for example is much higher for women than for men, the speaking rate, the frequency and the average length of pauses, and the voice quality, for example, harshness, should be taken into account.

1.2.2 General Difficulties

So far prosodic information has not been used in fully operational ASU systems and only little research has been done to show the improvement of single ASU modules by prosody within off–line experiments. This is due to two main reasons: the general difficulty in using prosodic information and the inherent properties of current approaches used in ASU systems.

The general difficulties arise from several factors:

- If several prosodic events occur within a few syllables or on the same syllable, their acoustic correlates influence each other strongly. For example, in He drinks? the pitch movement on the word drinks can indicate both accentuation and that the utterance is meant as a question. Therefore, the acoustic–prosodic realization of similar events can be quite different.

- Prosodic attributes are sometimes redundant in the sense that the meaning can as well, maybe with greater "effort", be inferred if the prosodic attributes are missing. Therefore, their strength may vary and they might even be absent, cf. also [Vai88, pp. 95–96].

- Different speakers realize the same event, for example, accent, by different prosodic means.

- There is no exact correspondence between syntactic and prosodic boundaries [Wig92c]. The same is true for focus and accent.

- In the current state–of–the–art, pitch detection and the measurement of duration are error–prone.

Concerning the inherent properties of methods used for ASR or ASU one has to take a closer look at the different levels:

- *Feature extraction* is not syllable–based. However, this would be the appropriate unit for representing prosodic information. Feature vectors are modeled by Gaussian distributions, because its parameters are easy to estimate. Pitch and its movements certainly are not Gaussian distributed.

- *Word recognition* is based on HMMs. They are able to model the duration only to a small extent by the state transition probabilities, since these are more or less based on exponential distributions [Nol87], whereas phone durations are Gamma distributed [Cry82, Wig92b].

- *Syntactic and semantic analysis* are currently knowledge based approaches, which are, for example, based on grammar rules. Researchers were interested in integrating prosodic information for a long time, but the rule based methods require hard decisions about the prosodic attributes, for example, if a word is accented, to be made by a possible prosody module. However, due to the difficulties in determining prosodic information as described above, prosodic information cannot be detected with absolute reliability. Only recently stochastic methods were combined with the knowledge based ASU systems using an A* search, already applied by [Nie85, Ehr88, Sag90] for the semantic network approach used in EVAR, cf. Section 3.3, and later adopted for one of the parsers used in VERBMOBIL [Sch94], or purely stochastic methods were developed [Mag94, Kuh95a]. These approaches allow for an adequate modeling of uncertainty as is the case with prosodic information.

- The first prototypes of *spoken dialog systems* have only existed for a few years, cf. Section 1.1.2. Consequently, interest in using prosodic information in dialog control in a real system has only become significant recently, though already proposed by [Lea80a].

1.2.3 State–of–the–art

A detailed literature survey of research related to the use of prosody in ASU including references will be given in Section 4.3 after the fundamentals of speech recognition and prosody have been given in Chapters 2, 3, and 4. Here, we will give a short summary about the state–of–the–art as far as relevant for the remainder of this chapter; the most important references can be found in Table 1.1.

Concerning feature extraction for word recognition the only work known to the author tried to normalize mel cepstrum features by pitch information [Sin92]. Energy is used by every word recognizer, it is the first cepstral coefficient, because it carries not only prosodic but also phonetic information.

Some work has been done already regarding the integration of prosody directly in a speech recognizer. Explicit models of accent patterns were used in several studies to reject vocabulary items, for example, [Wai88, Nöt91a]. It turned out that this can be done successfully; however, the integration into a continuous

WORD RECOGNITION	
pitch–based mel–cepstrum normalization	[Sin92]
reduction of the lexicon by accent information	[Wai88]
accent based verification of word hypotheses	[Nöt91a]
distinct HMMs for lexically stressed syllables	[Bis92]
explicit pitch and duration models	[Dum95]
n-best rescoring using lexical accent information	[Ana95]
n-best rescoring with prosodic–syntactic boundaries	[Vei93b]

LANGUAGE MODELING	
Markov model prediction of prosodic boundaries from text	[Vei90]
classification trees for the prediction of prosodic boundaries	[Wan92]

SYNTACTIC ANALYSIS	
parse scoring by an analysis/synthesis of prosodic boundaries	[Ost93b]
parse scoring by stochastic correlation of prosodic boundaries	[Hun94a]

LABELS	
tone and break indices: perceptual–prosodic labels	[Sil92a]

Table 1.1: Summary of the state–of–the–art in the research on the use of prosody in ASU.

speech recognizer was either not done at all or resulted in only little improvement. Another study used different phone models within lexically stressed or unstressed syllables of an HMM word recognizer, which reduced the word error rate by 10% [Bis92]. No special prosodic features, for example derived from pitch, were added to the usual cepstrum–based features. Finally, a group reports on experiments with a model of prosody, which was integrated in a continuous speech recognizer [Dum95]. For each sub–word unit the pitch, duration, and intensity were computed and their probabilities given a specific unit were modeled by Gaussian distributions. During recognition these probabilities were combined with the usual acoustic and the language model scores. The use of this model decreased the recognition error by about 8% on read speech.

Two other groups tried to rescore the n-best sentence hypotheses based on prosody. The first one achieved an error reduction of about 10% by phone duration models, which depend on the presence of lexical accent and on the phonemic context [Ana95]. Another group predicts prosodic phrase boundaries and accents based on the word chain and in a different model based on acoustic information. By combining the probabilities computed by the two models the average rank of the correct sentence hypothesis could be improved by 23% [Vei93a].

With a similar approach alternative parses of a given word chain were scored [Ost93b]. In 73% the system decided in favor of the correct out of two manually selected parses. Note that these results were achieved on a small database of minimal pairs read by professional speakers. In this case minimal pairs are sentences with the same wording or at least the same corresponding phoneme sequence but with different phrase structure. On the same data in [Hun94a] similar results are reported; in this case a method was used which statistically correlates prosodic–syntactic and acoustic–prosodic features.

Two studies successfully used stochastic language models for the prediction of prosodic phrase boundaries on the basis of the transliteration of utterances [Vei90, Wan92].

Statistical classifiers for prosodic events need labeled training data. So far an annotation scheme using tone and break indices are widely used [Sil92a]. These labels are created on the basis of a perceptual prosodic analysis of utterances.

To our knowledge no prosodic information has been used so far on the semantic, pragmatic, or dialog levels of ASU systems, neither in transfer nor generation within translation systems.

1.3 About this Book

1.3.1 Prerequisites and General Remarks

In Section 1.2.1 the potentials of the use of prosody in ASU were outlined. We agree with [Lea80a, Vai88, Nöt91a, Pol94] and believe that despite the difficulties mentioned in Section 1.2.2 prosodic information should be integrated more than it is up to now. It is at least one more source of information, which is believed to be useful for improving the performance of current ASU systems. This has been the motivation for the research reported in this book. Furthermore, the field of prosody allows to study the application of well known ASR or ASU methods such as HMMs, n-gram language models, or neural network/HMM–hybrids in new domains, cf. for example Section 7.1, or it inspires the development of new algorithms which eventually might be applicable in other fields of ASR or ASU, cf. for example Section 7.2.

With the research presented in this book we tried to fulfill the following goals:

- The *development* of methods for the use of stochastic prosodic information on several, mainly linguistic levels of ASU systems.

- The *integration* of these methods in existing systems, specifically in the ones described in Chapter 4.
- Eventually, *experiments* on large corpora of real spontaneous speech data produced by non–professional speakers should show if the developed and integrated algorithms are adequate.

Because of the difficulties named in Section 1.2.2 and in agreement with [Pri92, Ost93b, Hun95a] we favor statistical approaches. These allow to cope with uncertainty and with various acoustic realizations of the same prosodic phenomena. In contrast to [Moo94] we believe that uncertainty is inherent to speech in general and it is not just a problem of "poor models". This is especially true for prosody because its information is sometimes redundant and therefore it is left to the discretion of the speaker to utilize it or not, cf. Section 1.2.2. Variability in the acoustic realization of the same attributes arises from context influences or is due to speaker specific or socio–linguistic characteristics [Hel85, p. 15]. Rules describing prosodic phenomena can for practical reasons only be derived from small sets of controlled data, which is what phoneticians usually do, or they are rather intuitive, which is an approach linguists often choose. Such rules rarely are consistent with real–life data. In contrast, statistical models can take into account all these variabilities because they are trained on large speech corpora. Moreover, spontaneous speech data rather than controlled data can be used, cf. also the discussions in [Pri92, Bat94a]. Nevertheless, basic research on controlled data is very important, because it shows at least what the acoustic correlates of prosodic attributes are. Therefore we know for example that pitch movement can indicate accent. For a statistical approach, however, we do not have to know a priori what the shape of the pitch contour marking accent exactly looks like; we just have to provide a set of (heuristic) features encoding the relevant information and to specify a suitable model structure. Everything else can be left to the training procedure of the model. Note that in the long run it would also be desirable to include the feature extraction implicitly in the statistical model and to train not only the model parameters but also the model structure. Preliminary work concerning an implicit feature extraction has been done in course of the research presented in this book; it was conducted in connection with the detection of laryngealizations, cf. Chapter 9.

The methods for the use of prosody in ASU systems will be developed in a way that they can be integrated in existing systems. This is in contrast to Steedman's view, who proposes a new syntactic theory, which is capable of modeling prosodic attributes [Ste89, Ste92]. To our knowledge this model has never been realized, however, in the long run syntactic theories should incorporate rather than merely use prosodic information.

The methods should be developed in a way that they use the information already available at certain processing stages. For example, the interface between word recognition and syntax in VERBMOBIL is a word graph; both modules work sequentially. This means, if the prosody module's task is to enhance a word graph by prosodic information, it can and should utilize the phone level time alignment of the words in the word graph, but it has to cope with the uncertainty due to alternative word hypotheses. When the semantic module will be provided with prosodic information, the output of the syntax module can be used, which is one (or more) parse tree(s) based on one of the word chains contained in the word graph.

To achieve these goals there are important prerequisites to be fulfilled: the prosodic parameters have to be determined from preprocessed speech signals, and they have to be classified into prosodic events. Concerning these topics we base our research on the results achieved by [Nöt91a, Kie97]. In this context it is important to know that when prosodic attributes are to be recognized in spontaneous speech the models have to be trained on spontaneous speech, because spontaneous and read speech differ substantially both prosodically [Dal92, Sil92b, Bat95a] and segmentally [Bla91, Dal94b].

Finally, it should be remarked that the investigations throughout this book are made on German speech corpora. Therefore, all of the examples are given in German and are translated into English, where necessary word–by–word. Nevertheless, the basic algorithms developed are applicable to most languages.

1.3.2 Contribution to the Progress in Science

This book presents new methods for the use of prosodic information in different modules of ASU systems in order to improve their performance. These methods were evaluated on a large spontaneous speech database from many speakers and not, as in other studies, on data read by professional speakers sometimes even consisting of prosodic minimal pairs. The goal was not to model exactly all aspects of prosody, but to find approaches for the most important problems like clause boundary or focus disambiguation. The algorithms are designed in a way that they can in principle be used also for a more extensive prosody model. The basic methods applied were already known in many cases, but they have been adapted to new tasks throughout the research reported here. Furthermore, they have been successfully integrated in the ASU systems EVAR and VERBMOBIL. So, for the first time prosodic information has been used on the linguistic levels of fully operational prototype systems.

The state–of–the–art interface between word recognition and linguistic analysis is the word graph. Since most ASU systems, such as VERBMOBIL, process the information in a bottom–up manner, ways had to be found to score word graphs prosodically, so that this information can be used during subsequent linguistic analysis. Other studies so far tried the rescoring of n-best sentence hypotheses using prosodic information; the computation on word graphs is more complicated, but much more efficient. We believe that prosodic information should not be computed independently from the word recognizers' output.

Based on this prosodically scored word graph, the following methods have been developed for the integration of prosodic information in ASU systems:

- The most important contribution of the research presented in this book is the use of probabilities for prosodic clause boundaries in the parsing of word graphs. We showed that the parsing of spontaneous speech is only feasible if this prosodic information is used. This also means that prosodic information has to be computed before the parsing takes place. In our approach no hard decisions are taken a priori by the prosody module, but all alternatives are potentially explored during a search. Although the linguistic core formalism uses only symbols, the prosodic scores are used during the parsing process by guiding the search for the optimal path in the word graph and for the optimal parse given this word chain. In this context it is important that each word hypothesis in a graph is prosodically scored rather than attaching scores to the nodes in a graph. Our method is by an order of magnitude more efficient than first computing all possible parses of possibly different n-best sentence hypotheses, and afterwards rescoring them using prosodic information. Our approach is not specific with respect to a certain type of parser or grammar.

- In a parser based on an *head driven phrase structural grammar* (HPSG), information about prosodic clause boundaries was used to constrain the search space with respect to the position of empty head traces. This formalism can only deal with binary decisions; in this book a method is described which shows how to treat the probabilities for clause boundaries in a manner which improves the performance of the parser.

- Humans often control the meaning of discourse particles by the position of prosodic accents. We found that a score for the degree of accentuation computed for each word hypothesis can be used to disambiguate the semantic interpretation of these particles.

- On the dialog level the determination of sentence mood from the intonation contour was used successfully for dialog act classification. The continuation of the dialog by the machine was based on this. Furthermore, we adapted the methods for phrase boundary detection for the segmentation of utterances into dialog acts. This is an important prerequisite to the classification of dialog acts.

- Since up to now the integration of prosodic information directly into word recognizers had only limited success, a post–processor operating on word graphs has been developed for the use of prosodic information to improve the recognition of the best word chain. This is done by the combination of an acoustic–prosodic model with a stochastic language model, which predicts prosodic events on the basis of a word chain.

In order to retrieve prosodic information from a speech signal several steps of pre–processing and feature extraction are currently necessary. This book presents a new promising method for the inverse filtering of speech signals and, as an outlook, preliminary results concerning the classification of the phonation type (including laryngealizations) using neural networks which directly take the speech signal as input. The inverse filtering is a trainable non–linear filter. The classifier for the phonation type can be considered as being a direct model of the class a posteriori probabilities given the speech signal. No explicit and heuristic feature extraction has to be performed prior to classification as is usually the case in speech recognition.

For the training of classifiers for prosodic boundaries or accents labeled data are needed. Labeling based on perceptual evaluation of speech signals is very time consuming. We present a new scheme which allows to produce labels solely on the basis of the transliteration of utterances. We show that classifiers trained on these labels have better performance due to the larger amount of available training data. Furthermore, such labels are based on syntactic criteria, and are thus closer to the boundaries a syntax module in an ASU system would prefer to get recognized by a prosody module.

For the modeling of prosodic information the combination of stochastic language models with acoustic–prosodic models were explored. With respect to this the semantic classification tree approach was extended in order to integrate acoustic and language information in a single model and to allow for the use of variable length feature vectors. We adapted the well known n-gram language model approach for the classification of prosodic boundaries on the basis of word graphs, and we compared them with NN/HMM hybrid systems and with semantic classification trees. It turned out that a language model which takes into account information about the wording greatly improves the detection of prosodic boundaries. On our data the n-gram classifier showed best results and was integrated in an efficient way in the VERBMOBIL system.

1.3.3 Overview

The second chapter gives a brief literature review of the basic formalisms and algorithms used in the remainder of the book. How these have been adapted for the purpose of prosodic modeling will be described in Chapters 6 through 8. Since the main topic of this book is the development of algorithms for the improvement of ASU systems by prosodic information, an overview of the systems which are to be improved as well as of prosody will be given in Chapters 3 and 4, respectively. Chapter 5 describes the speech databases as well as their prosodic annotations, which are used in the experiments presented in the subsequent chapters. This includes a description of the labeling schemes developed for the research presented in this book.

Appropriate preprocessing of the speech data is necessary for prosodic analysis. Chapter 6 presents a new neural network based approach for inverse filtering of speech signals. Furthermore, the determination of pitch as well as the extraction of prosodic features and their classification will be reviewed in this chapter as far as relevant for the remainder of the book. In Chapter 7 new approaches for the modeling of prosodic phrases or of sequences of prosodic attributes by means of stochastic models are investigated. In particular hidden Markov models, n-grams, and semantic classification trees are considered. In course of the research presented in this book, for the first time a prosody module has been integrated into (two) ASU systems. In Chapter 8 it is outlined, how the performance of several components of these ASU systems is improved by prosodic information: On the level of word recognition prosodic information is used in a post–processor. Within the VERBMOBIL system prosody is used for boundary disambiguation on the syntactic and dialog levels, and prosodic accent information supports focus disambiguation and particle interpretation within the semantic and transfer modules. Information about sentence mood determined from the intonation is used for dialog act classification within the dialog module of EVAR. The book closes with an outlook and a summary. The appendix contains some equations and detailed examples as well as a reference to audio data which have been made available on the WORLD WIDE WEB for some of the examples given in this book.

The most important result of the research presented in this book is the improvement of the VERBMOBIL word graph parser by the use of prosodic clause boundary scores attached to word hypotheses. Figure 1.3 summarizes the most relevant sections with respect to this. The approach itself is described in Section 8.3.2. It uses a neural network, cf. Section 2.2, as acoustic–prosodic classifier and an n-gram language model, cf. Section 2.4, for computing the probability for a clause boundary occurring after a word hypothesis given a few words in the context. These classifiers are described in Sections 6.4 and 7.2, respectively. Section 7.2 also shows how they are combined to compute a joint probability. In Section 8.1 it is explained how this joint classifier is applied to word graphs. Although in Section 5.2 many different label types are defined, in the current approach we focus on the most important problem, which is the disambiguation of clause boundaries.

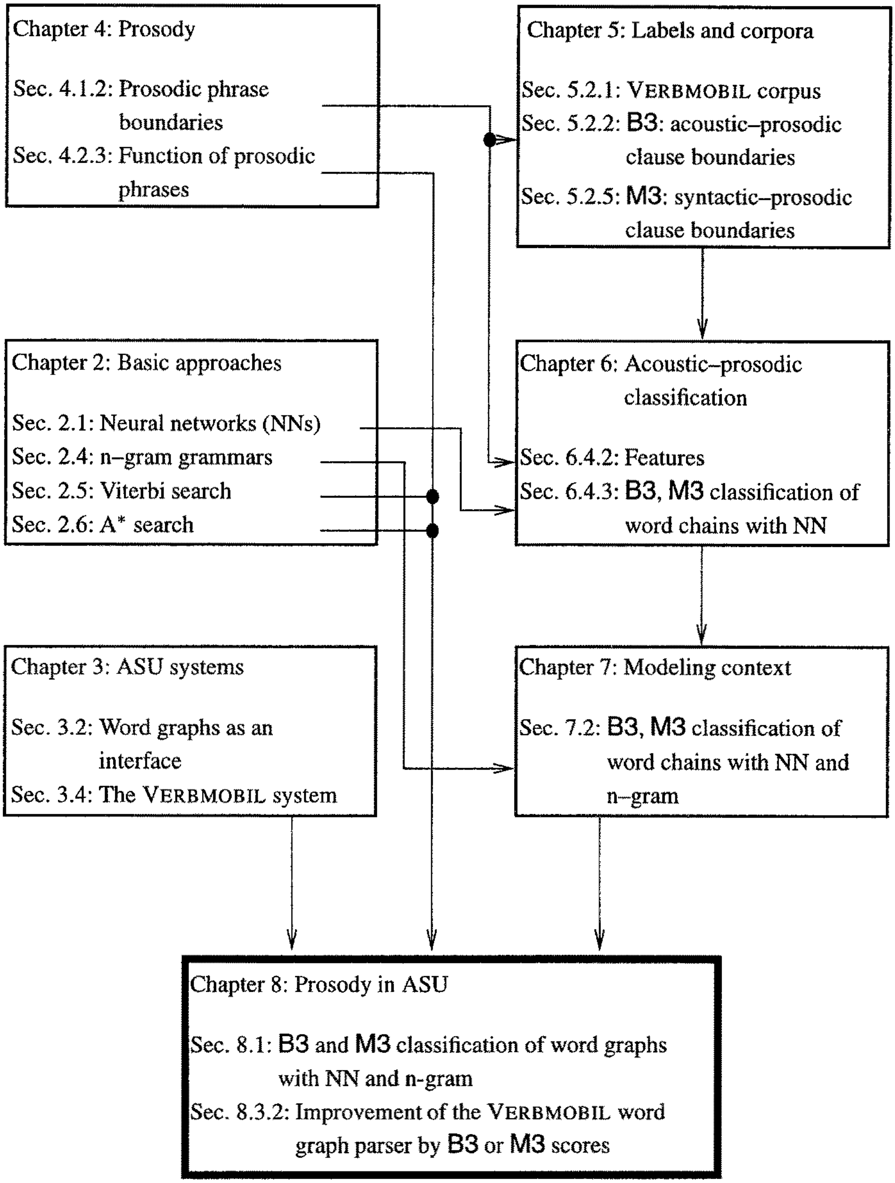

Chapter 4: Prosody
Sec. 4.1.2: Prosodic phrase boundaries
Sec. 4.2.3: Function of prosodic phrases
Chapter 5: Labels and corpora
Sec. 5.2.1: VERBMOBIL corpus
Sec. 5.2.2: B3: acoustic–prosodic clause boundaries
Sec. 5.2.5: M3: syntactic–prosodic clause boundaries
Chapter 2: Basic approaches
Sec. 2.1: Neural networks (NNs)
Sec. 2.4: n–gram grammars
Sec. 2.5: Viterbi search
Sec. 2.6: A* search
Chapter 6: Acoustic–prosodic classification
Sec. 6.4.2: Features
Sec. 6.4.3: B3, M3 classification of word chains with NN
Chapter 3: ASU systems
Sec. 3.2: Word graphs as an interface
Sec. 3.4: The VERBMOBIL system
Chapter 7: Modeling context
Sec. 7.2: B3, M3 classification of word chains with NN and n–gram
Chapter 8: Prosody in ASU
Sec. 8.1: B3 and M3 classification of word graphs with NN and n-gram
Sec. 8.3.2: Improvement of the VERBMOBIL word graph parser by B3 or M3 scores

Figure 1.3: Overview of the most important sections of the book, that is, the sections which are relevant for the improvement of the VERBMOBIL word graph parser by prosodic clause boundary scores are depicted. The arcs indicate their dependencies.

Chapter 2

Basic Approaches

The recognition of prosodic attributes involves methods which can automatically be optimized based on sample data. Furthermore, since different subsequent prosodic attributes influence each other, we need methods for modeling sequences of events. Different approaches are known from the literature. This chapter gives a short overview of the basic methods selected for the research described in this book; their specific adaptation for our purposes will be described later. We use neural networks (NNs) for the classification of single feature vectors and hidden Markov models (HMMs) for the modeling of sequences of features vectors. NN/HMM–hybrids can be used to combine the classification capabilities of NNs with the properties of HMMs regarding sequence modeling. For the modeling of symbol sequences n-gram language models and multi–level semantic classification trees are used. The latter can integrate symbolic and continuous feature processing. Finally, the A* search algorithm is a basic tool for the integration of prosodic attributes into automatic speech understanding. This algorithm also builds the basis for the integration of prosodic information in a word graph parser as will be described in Section 8.3.2. The prosodic information used in this parser are scores for phrase boundaries computed with NNs and n-grams.

2.1 General Considerations

Speech recognition is a special pattern classification problem; the following overview is based on [Nie83, Nie90a], cf. also [Dud73]. A pattern is in general defined as a function $\boldsymbol{f}(\boldsymbol{x}) \in \mathbb{R}^l, \boldsymbol{x} \in \mathbb{R}^m$, a speech signal is a one–dimensional pattern having the form $f(t) \in \mathbb{R}, t \in \mathbb{R}$. The process of pattern classification can be divided into the following steps, cf. Figure 2.1: first the signal is recorded and digitized. The resulting pattern $\boldsymbol{f}(\boldsymbol{x})$ is preprocessed, for example, to reduce noise

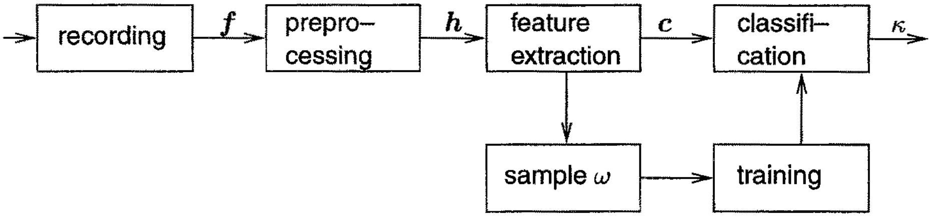


Figure 2.1: General structure of a pattern classifier, after [Nie90a].

or to enhance relevant properties. Afterwards relevant features are extracted from the pattern $\boldsymbol{h}(\boldsymbol{x})$. These features are summarized in a vector $\boldsymbol{c} \in \mathbb{R}^q$. Feature extraction can be seen as a transformation of the pattern so that the amount of data is reduced while the relevant information is kept as far as possible. Furthermore, the features are chosen so as to fulfill conditions imposed by the classifier. In general features of one class should occupy one or more compact regions in the feature space, and regions of different classes should be separated. The classifier finally maps the feature vector to an integer:

$$\boldsymbol{c} \mapsto \kappa \in \{1, \ldots, k\} \tag{2.1}$$

where k classes $\Omega_1, \ldots, \Omega_k$ are distinguished. An additional rejection class Ω_0 can be used for the cases where the feature vector cannot be mapped reliably to any other class.

We will only consider classifiers consisting of parametric functions. The parameters are optimized using a representative sample of patterns $\omega = \{{}^1\boldsymbol{c}, \ldots, {}^R\boldsymbol{c}\}$, so as to minimize a given risk criterion. Classification is based on a decision rule, which can be based on a distance measurement or on class probabilities. The optimal decision rule which minimizes the error probability decides for the class Ω_λ with maximal *a posteriori* probability $p(\Omega_\lambda|\boldsymbol{c})$, which is the probability of class Ω_λ given feature vector $\boldsymbol{c}$. The main problem in training a classifier is to find good estimates for the a posteriori probabilities. Usually the *Bayes rule*

$$p(\Omega_\lambda|\boldsymbol{c}) = \frac{p_\lambda p(\boldsymbol{c}|\Omega_\lambda)}{\sum\limits_{\kappa=1}^{k} p_\kappa p(\boldsymbol{c}|\Omega_\kappa)} \tag{2.2}$$

is applied so that the *a priori* probabilities p_λ and the *likelihood functions* $p(\boldsymbol{c}|\Omega_\lambda)$ can be estimated independently. This kind of classifier is called *statistical classifier*. Often a *normal distribution classifier* is used, where

$$\begin{aligned} p(\boldsymbol{c}|\Omega_\lambda) &= p(\boldsymbol{c}|\boldsymbol{\mu}_\lambda, \Sigma_\lambda) \\ &= (|2\pi\Sigma_\lambda|)^{-1/2} exp[-(\boldsymbol{c} - \boldsymbol{\mu}_\lambda)_t \Sigma_\lambda^{-1} (\boldsymbol{c} - \boldsymbol{\mu}_\lambda)/2] \,. \end{aligned} \tag{2.3}$$

This requires the feature vectors to be distributed according to multivariate normal distributions. The parameters of the distributions, the conditional mean vectors $\boldsymbol{\mu}_\lambda$ and covariance matrices $\boldsymbol{\Sigma}_\lambda$, have to be estimated based on *labeled* samples $\omega_\lambda \subset \omega$. That means, for each sample pattern the class membership has to be known. This process is called *supervised learning*.

If the features are not distributed as required by equation (2.3) the actual distribution can be approximated by a mixture of normal distributions [ST95a, pp. 80–81], which is defined as

$$p(\boldsymbol{c}|\Omega_\lambda) = \sum_{\nu=1}^{L_\lambda} w_{\lambda\nu} p(\boldsymbol{c}|\boldsymbol{\mu}_{\lambda\nu}, \boldsymbol{\Sigma}_{\lambda\nu}) \,, \quad \sum_{\nu=1}^{L_\lambda} w_{\lambda\nu} = 1 \,. \tag{2.4}$$

Usually, for the training sample labels Ω_λ are available but no labels of the form $\Omega_{\lambda\nu}$ are given. Consequently, parameters of each mixture distribution have to be trained *unsupervised*. In the case of maximum likelihood estimation this can be done using the *expectation–maximization* (EM) algorithm [Dem77] as described in [ST95a].

Another possibility is to use *distribution–free* classifiers [Nie90a, pp. 158–160]. In general for k classes these are based on a vector

$$\boldsymbol{d} = (d_1(\boldsymbol{c}, \boldsymbol{a}), \ldots, d_k(\boldsymbol{c}, \boldsymbol{a}))_t \,, \quad \boldsymbol{a} \in \mathbb{R}^a \tag{2.5}$$

of parametric discriminant functions $d_\lambda(\boldsymbol{c}, \boldsymbol{a})$. A feature vector $\boldsymbol{c}$ is classified according to the decision rule

$$\kappa = \underset{1 \leq \lambda \leq k}{\operatorname{argmax}} \{d_\lambda(\boldsymbol{c}, \boldsymbol{a})\} \tag{2.6}$$

The problem in the design of such classifiers is to define an appropriate family of parametric functions d_λ and to optimize the actual parameters $\boldsymbol{a}_1, \ldots, \boldsymbol{a}_k$. Let

$$\boldsymbol{\delta} = (\delta_1(\boldsymbol{c}), \ldots, \delta_k(\boldsymbol{c})) \,, \quad \text{where} \quad \delta_\kappa(\boldsymbol{c}) = \begin{cases} 1 & \text{if } \boldsymbol{c} \in \Omega_\kappa \\ 0 & \text{else} \end{cases} \tag{2.7}$$

be an ideal discriminant function. With equation (2.7) and given labeled sampling data, the expectation of the square error

$$\epsilon = E\{(\boldsymbol{\delta} - \boldsymbol{d})^2\} \tag{2.8}$$

can be used as optimization criterion. It can be shown that if the function $\boldsymbol{d}$ is sufficiently complex, the function d^* minimizing the criterion (2.8) is identical to the vector of a posteriori probabilities [Nie90a, Whi89]:

$$\boldsymbol{d}^* = (p(\Omega_1|\boldsymbol{c}), \ldots, p(\Omega_k|\boldsymbol{c})) \,. \tag{2.9}$$

This also means that the general distribution–free classifier minimizes the error probability. In [Ric91] the proof for this has been extended to other optimization criteria and to other types of ideal discriminant functions.

A traditional type of distribution–free classifiers whose parameters can be optimized automatically is the *polynomial classifier* [Sch77, Sch84]. It is based on discriminant functions, which are a linear combination

$$d_\lambda(\boldsymbol{c}, \boldsymbol{a}) = \sum_{\nu=1}^{L_\lambda} a_{\lambda\nu}\varphi_\nu(\boldsymbol{c}) \tag{2.10}$$

of terms $\varphi_\nu(\boldsymbol{c}) \in \mathbb{R}$ being polynomials in the components of $\boldsymbol{c}$. For example, the quadratic classifier is defined as

$$\begin{aligned} \varphi_1(\boldsymbol{c}) &= 1 \\ \varphi_\nu(\boldsymbol{c}) &= c_\nu\,, \quad 1 \le \nu \le q \\ \varphi_{\nu+q+1}(\boldsymbol{c}) &= c_i c_j\,, \quad 1 \le i \le q, \ \le j \le i, \ \le \nu \le q(q+1)/2\,. \end{aligned} \tag{2.11}$$

In practice, it can often be observed that polynomial classifiers do not compute a posteriori probabilities; usually even $0 \le d_\lambda(\boldsymbol{c}, \boldsymbol{a}) \le 1$ does not hold [Kil93]. Another type of non–linear distribution–free classifier, feed–forward neural networks, will be described in the following section. As for other types of classifiers refer to the text books cited above.

So far we considered only approaches useful for *isolated speech recognition*. However, we will usually deal with utterances containing several events or classes, which can, for example, be words or phrases. This means we have to conduct *continuous speech recognition*, which is a mapping

$$[{}^n\boldsymbol{c}]_{1\le n\le T} \mapsto [\kappa_i]_{1\le i\le m}\,, \quad 1 \le \kappa_i \le k \tag{2.12}$$

of a sequence of feature vectors to a sequence of class symbols where m is the number of class symbols contained in the sequence corresponding to an utterance. Compared to equation (2.1) a symbolic transcription is generated for a pattern rather than mapping the pattern to a single class symbol. In [Nie90a] this is called *pattern analysis* in opposite to *pattern classification*.

After having trained speech classifiers or recognizers we want to measure their *quality*, for an overview cf. [Nie83, Sec. 4.1] and [Pic86]. This is done by recognizing a sample of test patterns which has to be independent of the training sample. The test patterns have to be labeled with the *reference* class $\Omega_\kappa, 1 \le \kappa \le k$, in the case of isolated speech recognition or with the reference class sequence $\boldsymbol{\Omega} = [\Omega_{\kappa_i}]_{1\le i\le m}, 1 \le \kappa_i \le k$ in the case of continuous speech recognition. The recognized class (sequence) is then compared with the reference. In the case of isolated speech recognition we define the *recognition rate* RR as

$$RR = \frac{\#(\text{of correctly classified test patterns})}{\#(\text{of test patterns})} \cdot 100\% \tag{2.13}$$

which is the percentage of correctly classified test set patterns. In addition we sometimes give the class–dependent recognition rates as

$$RR(\Omega_\kappa) = \frac{\#(\text{correctly classified patterns of class } \Omega_\kappa)}{\#(\text{patterns of class } \Omega_\kappa)} \cdot 100\% \tag{2.14}$$

or the (unweighted) average of the class–dependent recognition rates defined as

$$CRR = \frac{1}{k} \sum_{\kappa=1}^{k} RR(\Omega_\kappa)\ . \tag{2.15}$$

In the case of continuous speech recognition we will compare for each utterance the recognized sequence of classes with the reference without considering the position of the recognized classes on the time axis. Since the number of classes within recognized and reference sequence may differ, we have to align both sequences. The following is a possible alignment of two class sequences:

reference	Ω_3	ins	Ω_1	Ω_6	Ω_4	Ω_7
recognized	Ω_2	Ω_2	Ω_1	del	Ω_3	Ω_7

This alignment has to be interpreted as follows: the second recognized class (Ω_2) was *inserted*, because it has no correspondence in the reference. The class Ω_6 was *deleted*, because it has no correspondence in the recognized sequence. Furthermore, the classes Ω_3 and Ω_4 in the reference were *substituted* by Ω_2 and Ω_3, respectively. The classes Ω_1 and Ω_7 in the reference were *correctly* recognized. An optimal alignment between two symbol sequences can be achieved by minimizing the *Levenshtein distance* [Lev66], where the sum of the inserted, deleted, and substituted symbols is minimal. This optimal alignment can be efficiently determined by dynamic programming [Bel72]. The recognition accuracy for continuous speech recognition is defined as

$$\begin{aligned} RA &= (1 - \frac{\#(\text{errors})}{\#(\text{patterns})}) \cdot 100\% \\ &= (1 - \frac{\#(\text{sub}) + \#(\text{ins}) + \#(\text{del})}{\#(\text{patterns})}) \cdot 100\% \end{aligned} \tag{2.16}$$

After these general remarks on pattern classification and analysis we will describe the particular methods used in the research presented in this book.

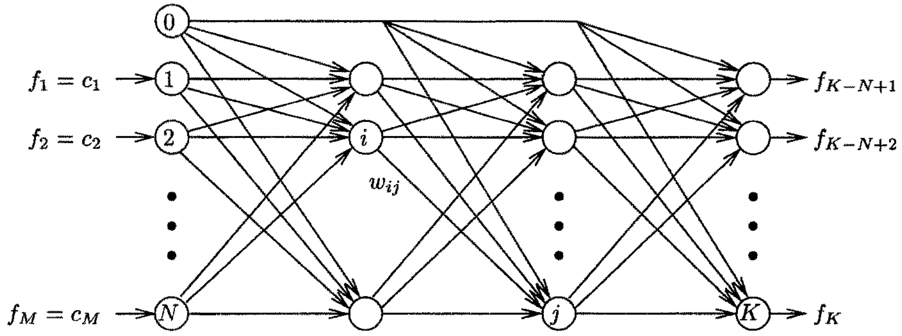


Figure 2.2: Topology of a three–layer perceptron.

2.2 Neural Networks (NNs)

Neural networks (NNs) are *computational* models resembling some of the characteristics of brains of living beings: they consist of a large number of simple computation units, which are highly interconnected. However, state–of–the–art NNs are far from being *artificial* neural networks in the sense of simulating the properties, especially, the learning capabilities, of biological neural networks, cf. the discussion in [Bez94]. Different types of neural networks can be distinguished by the topology of the network, the function computed by the basic units, and the training algorithm. Good detailed treatments of neural networks can be found, for example, in [Rum86, Lip87, Roj93, Rip96a], the descriptions in Sections 2.2.1 and 2.2.2 are based on [Nie90a]. In this book NNs are used as tools for classifying prosodic feature vectors and for the filtering of speech signals. For the classification or transformation of feature vectors obtained from speech signals NNs which are based on the *multi–layer perceptron* (MLP) have been found useful by different researchers, cf. for example [DM88, Leu88, Wai89b, Fal90, Lip90, Bou94b, Ben95]. We know of only one group who experimented with a non–MLP approach; they used *Boltzmann machines* for vowel recognition [Pra86].

2.2.1 Definitions and Computation

For the structure of a three–layer MLP cf. Figure 2.2. An n-layer MLP consists of a layer of input nodes, $n-1$ hidden layers, and a layer of output nodes. The nodes from one layer are fully connected to the ones of the next higher layer by directed edges. The input features are propagated through the network from the input to the output nodes along the edges according to the computations outlined below. In the experiments presented in this book we use MLPs, however, the following definitions and the training algorithms described in Section 2.2.2 hold for general feed–forward networks. These have as only restriction concerning the topology that the network must not contain any cycles. Such a network consists of the $K+1$ nodes $\{n_0, n_1, \ldots, n_K\}$, which are grouped as follows:

- bias node: n_0
- M input nodes: $\{n_1, \ldots, n_M\}$
- N output nodes: $\{n_{K-N+1}, \ldots, n_K\}$
- all other $(K - N - M + 1)$ nodes are called *hidden* nodes: $\{n_{M+1}, n_{M+2}, \ldots, n_{K-N}\}$

Edges must not lead from node n_i to n_j if $i \geq j \vee j \leq M \vee i > K - N$. This assures that the NN has no cycles and that no edges are between input or output nodes, respectively.

With each edge from node n_i to n_j a weight w_{ij} is associated. Input to the network is a feature vector $(c_1, \ldots, c_M)$. The inputs y_j to the non–input nodes n_j are computed successively starting with node n_{M+1} according to

$$y_j = \sum_{i \in N_j} w_{ij} f_i \,, \quad M < j \leq K \,. \tag{2.17}$$

N_j determines the set of nodes from which edges lead to n_j, cf. Figure 2.4. If one defines that "no edge between n_i and n_j" as being equivalent to $w_{ij} = 0$, the set N_j simply becomes the set $\{1 \leq i < j\}$. The output or the *activation* f_i of a node is computed as:

$$f_i = \Theta(y_i) \,, \quad M < i \leq K \tag{2.18}$$

where the *activation function* Θ is usually chosen as

$$\Theta(y) = \frac{1}{1+e^{-y}} \quad \text{with} \quad \frac{d\Theta(y)}{dy} = \Theta(y)(1 - \Theta(y)) \,. \tag{2.19}$$

This function, called *sigmoid function*, is often motivated by the fact that neurons in biological systems "fire" if the incoming signals exceed a certain threshold. The threshold function itself is not useful with respect to the training algorithm described in Section 2.2.2, because it is not continuously derivable. The sigmoid function is a good approximation and fulfills this condition, cf. Figure 2.3. A further useful property is that its range is between zero and one. We use the hyperbolic tangent function for the hidden nodes, because it usually contributes to a somewhat faster convergence during training. It is often also referred to as the *symmetric sigmoid* function. Other continuously derivable functions can be used as well, but are not considered in this book.

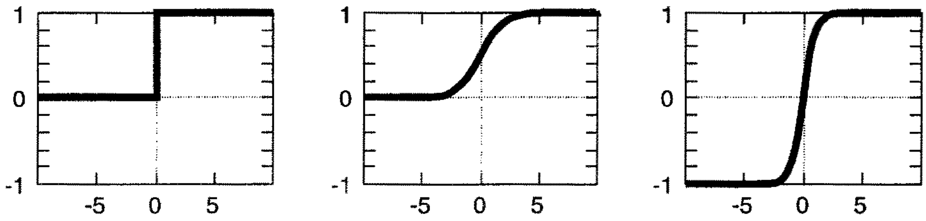


Figure 2.3: Non–linear activation functions. From left to right: threshold, sigmoid and hyperbolic tangent function.

The bias node n_0 has the constant activation of $f_0 = 1$. It is usually used so that even if the outputs of all nodes in N_j (except for the bias) are zero, the output f_j of n_j can be non–zero. The activation of the input nodes is the identity: $f_i = c_i,\ 1 \leq i \leq M$.

2.2.2 Training

The weights of a feed–forward NN can be optimized with the *back–propagation* algorithm. At each iteration a sample of R feature vectors ${}^{\varrho}\mathbf{c}$, $1 \leq \varrho \leq R$, are successively used as input to the NN. For each of these the vector of *desired* or ideal outputs ${}^{\varrho}\boldsymbol{\delta}$ has to be given. Back–propagation is a *gradient descent* method iteratively minimizing the output error ϵ, which means, the weights are updated after each iteration n by

$$ {}^{n+1}w_{ij} = {}^{n}w_{ij} + \Delta({}^{n+1}w_{ij}) = {}^{n}w_{ij} + \alpha\frac{\partial\epsilon}{\partial({}^{n}w_{ij})} + \beta\Delta({}^{n}w_{ij})\,, \qquad (2.20)$$

where α is called the *learning rate*, β is the *momentum weight* and n denotes the iteration number, which is omitted in the following for simplicity. The weights are usually initialized with random numbers taken from a small interval centered at zero. The *mean squared error* criterion is usually used as error function:

$$\epsilon = \sum_{\varrho=1}^{R} {}^{\varrho}\epsilon = \sum_{\varrho=1}^{R}\sum_{j=1}^{N}({}^{\varrho}\delta_j - {}^{\varrho}f_{K-j+1})^2\,, \qquad (2.21)$$

where ${}^{\varrho}f_j$ is the activation of node n_j when ${}^{\varrho}\mathbf{c}$ was the input to the NN.

The essential part of the algorithm is the computation of the partial derivatives of the error. First, these derivatives are computed for the edges leading to output nodes by

$$\frac{\partial^{\varrho}\epsilon}{\partial w_{ij}} = \frac{\partial^{\varrho}\epsilon}{\partial^{\varrho}f_j}\,\frac{\partial^{\varrho}f_j}{\partial^{\varrho}y_j}\,\frac{\partial^{\varrho}y_j}{\partial w_{ij}} = -{}^{\varrho}d_j\,{}^{\varrho}f_i\,, \quad 1 \leq l \leq N,\ j = K - l\,, \qquad (2.22)$$

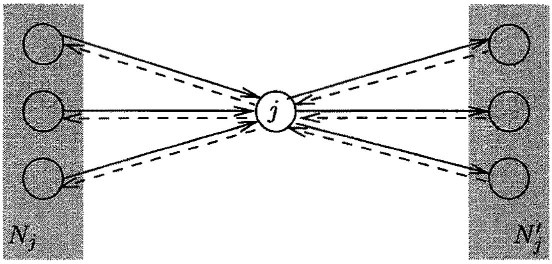


Figure 2.4: Part of a feed–forward NN illustrating the node sets N_j and N'_j. The solid arcs indicate the direction of the forward propagation of the activations f_j through the network, whereas the dashed arcs indicate the direction of the backward propagation of the error term d_j.

with

$$ {}^{\varrho}d_j = ({}^{\varrho}\delta_l - {}^{\varrho}f_j)(1 - {}^{\varrho}f_j){}^{\varrho}f_j \,. \tag{2.23} $$

Then the partial derivatives for the hidden nodes in descending order, starting with node n_{K-N} are computed. As can be shown these derivatives are

$$ \frac{\partial \epsilon}{\partial w_{ij}} = \sum_{\varrho=1}^{R} {}^{\varrho}d_j {}^{\varrho}f_i \,, \quad K - N \geq j > M \,, \quad i \in N'_j \,, \tag{2.24} $$

with

$$ {}^{\varrho}d_j = \sum_{\{i \in N'_j\}} {}^{\varrho}d_i {}^{\varrho}w_{ji}(1 - f_j)f_j \,, \quad K - N \geq j > M \tag{2.25} $$

where N'_j is the set of nodes to which edges lead starting at n_j, cf. Figure 2.4. Since the computation of the terms d_j is based on previously computed d_i, with $i > j$, the algorithm performs a propagation of the mean square error ϵ backwards through the network along the edges but in reverse direction.

The back–propagation algorithm is guaranteed to converge to a local minimum, but it usually converges very slowly, because the partial derivatives of the error function only determine the direction of the optimal weight change, but they do not give information about the best magnitude of weight change. This can be heuristically controlled by the learning rate α, which is often slowly decreased with increasing iteration number. The momentum is often used to prevent the weights from oscillating. The weight $\beta \geq 0$ has to be chosen heuristically.

An extension to the back–propagation algorithm which tries to find a coarse estimate for the optimal magnitude of the weight change is the *quick–propagation* algorithm [Fah89]. It updates the weights according to

$$ \Delta w_{ij,n+1} = \frac{\frac{\partial \epsilon}{\partial w_{i,j,n+1}}}{\frac{\partial \epsilon}{\partial w_{i,j,n}} - \frac{\partial \epsilon}{\partial w_{i,j,n+1}}} \Delta w_{ij,n} \tag{2.26} $$

In the first iteration or if $\Delta w_{ij,n} = 0 \wedge \frac{\partial \epsilon}{\partial w_{i,j,n+1}} > 0$ equation (2.24) is used. Furthermore, if $\Delta w_{ij,n+1} > \gamma$ according to equation (2.26), we use $\Delta w_{ij,n+1} = \gamma \Delta w_{ij,n}$ instead. Though often faster in convergence than back–propagation the quick–propagation algorithm does not necessarily converge to a minimum.

2.2.3 Properties of Feed–forward NNs

For the purpose of pattern classification usually an NN is used where each of the output nodes is associated with one and only one Ω_λ. With respect to statistical classifiers feed–forward NNs have the advantage that arbitrary input can be used, that means, especially, feature vectors do not have to fulfill any statistical properties. NNs can be viewed as distribution–free classifiers such as defined in Section 2.1. In [Lip87] it has been shown that NNs as defined above consisting of at least three hidden layers can compute arbitrary functions and can form arbitrary decision regions in the feature space. This altogether allows that NNs can be viewed as estimators of class a posteriori probabilities $p(\Omega_\lambda|\boldsymbol{c})$ provided they have a sufficient number of weights and during training the error function converges to a global minimum. Note that it is an open problem if these assumptions are fulfilled well enough in practice. However, in [Ric91] this theoretical result has also been supported by simulation studies. This topic is not only of interest if NNs are used as classifiers; it is especially important to know if NN output values behave like probabilities if they are to be further used in combination with probabilities computed by different models. In this context it is not sufficient that the output values sum to one — as it is often enforced by dividing each output value by the sum of all output values — but one has to be sure about the interpretation of the output values as probabilities. Several researchers believe that NNs approximate a posteriori probabilities sufficiently well [Whi89, Gis90, Ric91, Bou94b, Mor95], others disbelieve. We will investigate this problem again empirically with the NNs used in our experiments, cf. Section 6.4.4.

2.2.4 Extensions to Feed–forward NNs

For the recognition of pattern sequences as in continuous speech recognition *time–delay* feed–forward and recurrent NNs were introduced. Assume T feature vectors $({}^1\boldsymbol{c}, {}^2\boldsymbol{c}, \dots, {}^T\boldsymbol{c})$ are subsequently the input of an NN. The output activations of the nodes get a time index n, so that $f_{i,n}$ and $y_{i,n}$ denote the activation and the input, respectively, of node n_i when the feature vector ${}^n\boldsymbol{c}$ is input to the NN. Time–delay NNs (TDNNs) have first been introduced in [Wai89a]; an extension which does not impose any restrictions on the weights has been described in [Ben95]. In a

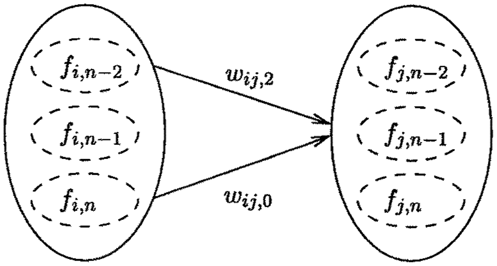


Figure 2.5: Example for time–delayed edges from node n_i to n_j in an NN.

TDNN the input to a node n_j is computed as

$$y_{j,n} = \sum_{\{(i,\tau)\in N_j\}} w_{ij,\tau} f_{i,n-\tau} \ , \quad M < j \leq K \tag{2.27}$$

where τ denotes the time–delay associated with an edge. The definition of N_j has to be extended to capture also the time–delayed edges. Figure 2.5 shows an example of time–delayed connections between two nodes n_i and n_j.

Note that TDNNs are nothing but an efficient representation of a feed–forward network used to model time sequences. If at time instance t_n all the feature vectors $c_{n-\sigma}, \ldots, c_n$ and at time instance t_{n+1} the feature vectors $c_{n-\sigma+1}, \ldots, c_{n+1}$ were input to an "ordinary" feed–forward network, the network would have the same parameters as a TDNN, provided σ was chosen large enough. In contrary, recurrent NNs as depicted in Figure 2.6 allow for a significant parameter reduction, or since the number of trainable parameters is usually limited by the amount of training data, they allow for better recognition rates having the same number of parameters. Note that all recurrent edges have to be time–delayed. The back–propagation algorithm can easily be extended to recurrent NNs [Ben95]. In [DM93, Ben92b] it has been shown that such networks can significantly improve phoneme recognition over ordinary feed–forward networks.

The topology of an NN is usually predefined and kept constant during training. However, it can also be optimized using, for example, the *cascade correlation algorithm* [Fah90] or *genetic algorithms* [Roj93, Chap. 17]. If the topology of an NN is predefined, the number of weights can be significantly reduced based on domain knowledge without a decrease or sometimes even with an increase in recognition rate. For example, in [Ben91, Ben92b, DM93] the spectrum and time–domain features computed from the speech signal are input to NNs for phone recognition. The input and the first hidden layer are not fully connected, but both are segmented into regions, and only the nodes of a few pairs of regions are connected. In [Wai89b] large modularized NNs for phoneme recognition have been successively built out of smaller ones, which were first optimized for dedicated subtasks.

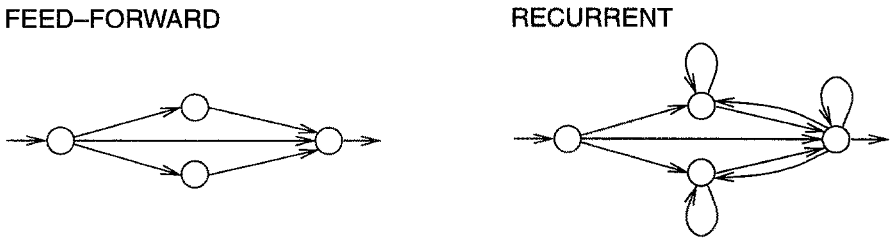


Figure 2.6: A general feed–forward NN and its extension to a recurrent NN.

2.3 Hidden Markov Models (HMMs)

In *word recognition* a speech signal has to be mapped to a word or a sentence hypothesis. It is not convenient to do this directly by a classifier as described in Section 2.1, because the number of features per vector extracted from the entire speech signal would be intractable large and furthermore the number of classes (words, sentences) is far too high. Therefore, one extracts a sequence of feature vectors $[^n\boldsymbol{c}]_{1\leq n\leq T}$ and assumes they were generated by a stochastic process. The problem is to find an appropriate model for this stochastic process. For two decades hidden Markov models (HMMs) have been successfully used for this task. Tutorials can be found in [Rab89, Rab93, Hua90]; in the following we will give an overview.

2.3.1 Structure

An HMM is a stochastic automaton defined by a triple $(\boldsymbol{\pi}, \boldsymbol{A}, \boldsymbol{B})$. It defines a doubly embedded stochastic process: The first one is a discrete process, which generates a sequence of state symbols $\boldsymbol{s} = [s_n]_{1\leq n\leq T}$, with s_n being a random variable which at each time step t_n takes one value from a finite alphabet $S = \{S_1, \ldots, S_I\}$. The vector $\boldsymbol{\pi}$ gives the initial state probabilities:

$$\boldsymbol{\pi} = [\pi_i] = P(s_1 = S_i)\ , \quad 1 \leq i \leq I\ . \qquad (2.28)$$

State transitions are defined by the probability matrix

$$\boldsymbol{A} = [a_{ij}] = P(s_{n+1} = S_j | s_n = S_i)\ , \quad 1 \leq i,j \leq I\ , \quad 1 \leq n \leq T\ . \qquad (2.29)$$

where first order time invariant processes are assumed, that means, the probability of arriving in a state depends only on the immediate predecessor state and is furthermore independent from the time t_n:

$$\begin{aligned} a_{ij} &= P(s_{n+1} = S_j | s_n = S_i) \\ &= P(s_{n+\tau+1} = S_j | s_{n+\tau} = S_i) \\ &= P(s_{n+1} = S_j | s_n = S_i, s_{n-1}, \ldots, s_1) \,, \\ & \quad 1 \leq i, j \leq I \,, \quad 1 \leq n \leq T \,. \end{aligned} \tag{2.30}$$

Note that this assumption is necessary to keep the models computationally tractable but does in general not meet the reality in the case of continuous speech.

When arriving at a state an output can be observed. A state sequence $\boldsymbol{s}$ generates a sequence $\boldsymbol{o}$, called *observation* sequence. It is assumed that in contrast to these observations the underlying state sequence is *hidden*, that is, it cannot be observed. Observations either can be discrete (*discrete HMMs*) taken from a finite alphabet, that is, $\boldsymbol{o} = [o_n]_{1 \leq n \leq T}$ with $o_n \in O = \{O_1, \ldots, O_L\}$ or they can be continuous valued (*continuous HMMs*), that is, the observations are sequences of vectors $\boldsymbol{o} = [{}^n\boldsymbol{c}]_{1 \leq n \leq T}$. A specific output can be observed with a certain probability depending on the actually occupied state. In the case of continuous HMMs these probabilities are defined by the probability density functions

$$\boldsymbol{B} = [b_i({}^n\boldsymbol{c})] = p({}^n\boldsymbol{c} | s_n = S_i) \,, \quad 1 \leq i \leq I \,. \tag{2.31}$$

Mixtures of normal distributions as defined in equation (2.4) are usually used as density functions. For discrete HMMs the output probabilities are given by the IxL–matrix

$$\boldsymbol{B} = [b_{il}] = P(o_n = O_l | s_n = S_i) \,, \quad 1 \leq i \leq I \,, \quad 1 \leq l \leq L \,. \tag{2.32}$$

These symbols are usually abstract classes in the sense of Section 2.1 provided by a vector quantizer, which is a classifier trained *unsupervised* [Gra84, Koh90]. Classes can also be phone symbols as in [Kun90, Ril95]. Often *semi–continuous* HMMs are used [Hua93]. These can either be interpreted as continuous HMMs where a pool of L distributions is used which are independent from the particular states. Only the mixture weights $w_{i\nu}, 1 \leq \nu \leq L$ are specific for each state i. These pooled density functions can also be interpreted as *soft vector quantization*[Cla92]. Note that equations (2.31) and (2.32) assume the independence of subsequent observations o_n and o_{n+1} or ${}^n\boldsymbol{c}$ and ${}^{n+1}\boldsymbol{c}$, respectively. This assumption is usually not fulfilled in the case of speech. In Chapter 7 we will only consider continuous HMMs.

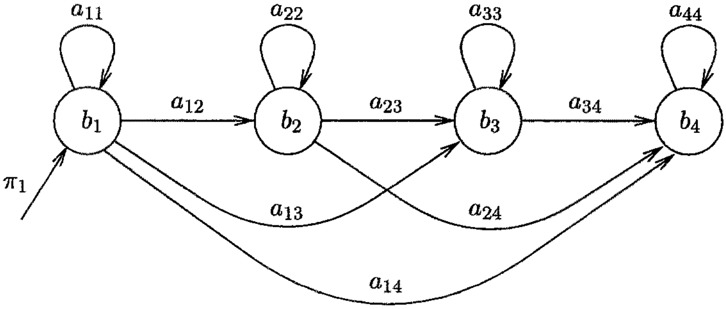


Figure 2.7: A left–to–right HMM with state transition probabilities a_{ij} and observation densities b_i associated with each state S_i.

2.3.2 Speech Recognition with HMMs

In ASR usually so called *left–to–right* HMMs are used, that is, $a_{ij} = 0$ for $i > j$, $\pi_1 = 1$, and $\pi_i = 0$ for $i > 1$. An example is given in Figure 2.7. The transitions "skipping" states as the one from s_1 to s_3 are needed to be able to model observation sequences of arbitrary length.

All computations with HMMs are based on a *trellis* as depicted in Figure 2.8. It indicates that each of the observations could have been emitted by any state. Furthermore, the possible state transitions are shown and thereby all possible state sequences $\boldsymbol{s}$ are defined. An observation sequence $\boldsymbol{o}$ might have been generated by any of these sequences according to $P(\boldsymbol{o}|\boldsymbol{s}, \boldsymbol{\pi}, \boldsymbol{A}, \boldsymbol{B})$.

In *isolated speech recognition* for each class Ω_λ one HMM $(\boldsymbol{\pi}_\lambda, \boldsymbol{A}_\lambda, \boldsymbol{B}_\lambda)$ is used. Classes are often words out of a given vocabulary of size k. One decides for the Ω_κ with maximum a posteriori probability according to the following rule:

$$\begin{aligned}\kappa &= \underset{1\le\lambda\le k}{\operatorname{argmax}} \, \{p_\lambda P(\boldsymbol{o}|\boldsymbol{\pi}_\lambda, \boldsymbol{A}_\lambda, \boldsymbol{B}_\lambda)\} \\ &= \underset{1\le\lambda\le k}{\operatorname{argmax}} \, \{p_\lambda \sum_{\boldsymbol{s}} P(\boldsymbol{o}|\boldsymbol{s}, \boldsymbol{\pi}_\lambda, \boldsymbol{A}_\lambda, \boldsymbol{B}_\lambda) P(\boldsymbol{s}|\boldsymbol{\pi}_\lambda, \boldsymbol{A}_\lambda, \boldsymbol{B}_\lambda)\} \,, \end{aligned} \tag{2.33}$$

where the probabilities for $\boldsymbol{o}$ given a particular state sequence $\boldsymbol{s}$ are computed and summed up over all possible sequences $\boldsymbol{s}$. The direct computation of these probabilities involves in the case of an ergodic, that is a fully connected, discrete HMM $(2T-1)I^T$ multiplications, which is infeasible. However, the probabilities $P(\boldsymbol{o}|\boldsymbol{\pi}, \boldsymbol{A}, \boldsymbol{B})$ can be efficiently computed with the help of the *forward variables*

$$\alpha_{ni} = P({}^1\boldsymbol{c}, \ldots, {}^n\boldsymbol{c}, s_n = S_i|\boldsymbol{\pi}, \boldsymbol{A}, \boldsymbol{B}) \,, \tag{2.34}$$

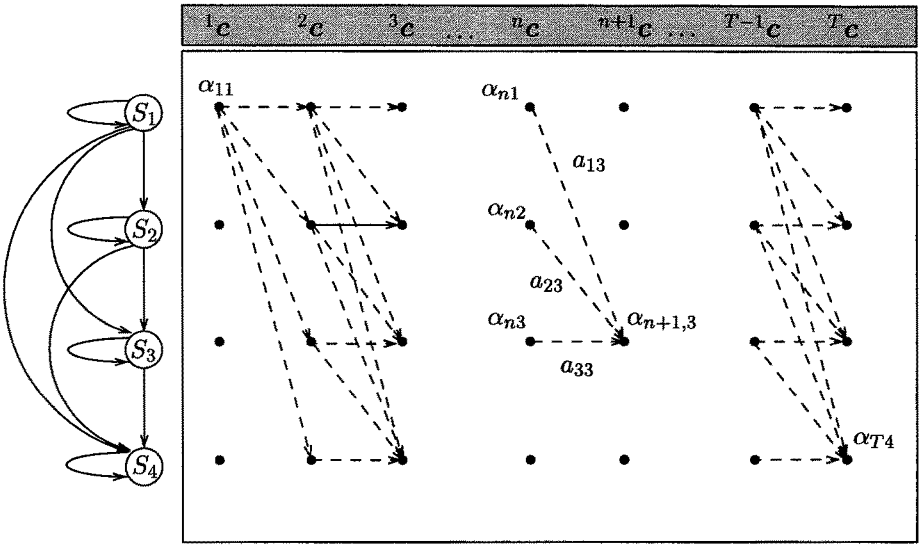


Figure 2.8: Trellis for the computation with HMMs, after [Kuh95b]. In this example the forward variable in state S_3 at time t_{n+1} is computed as $\alpha_{n+1,3} = (a_{13}\alpha_{n1} + a_{23}\alpha_{n2} + a_{33}\alpha_{n3})\, b_3(^{n+1}\boldsymbol{c})$.

which denote the probability for observing the first n symbols in $\boldsymbol{o}$ and being in state S_i at time t_n given a particular HMM. The α_{ni} can be computed as follows:

1. Initialize

$$\alpha_{1i} = \pi_i b_i(^{1}\boldsymbol{c}) , \quad 1 \leq i \leq I . \tag{2.35}$$

2. Iteratively compute all other variables by the following recursion scheme, again assuming the independence of subsequent observations:

$$\alpha_{n+1,j} = \sum_{i=1}^{I} \alpha_{ni} a_{ij} b_j(^{n+1}\boldsymbol{c}) , \quad 1 \leq j \leq I . \tag{2.36}$$

3. Terminate by computing

$$P(\boldsymbol{o}|\boldsymbol{\pi}, \boldsymbol{A}, \boldsymbol{B}) = \sum_{i=1}^{I} \alpha_{Ti} . \tag{2.37}$$

This algorithm only involves $I(I+1)(T-1) + I = O(I^2T)$ multiplications.

In *continuous speech recognition*, one uses a *looped* HMM as described in [Lee89]. Figure 2.9 shows the basic structure of such an HMM. This is built out of

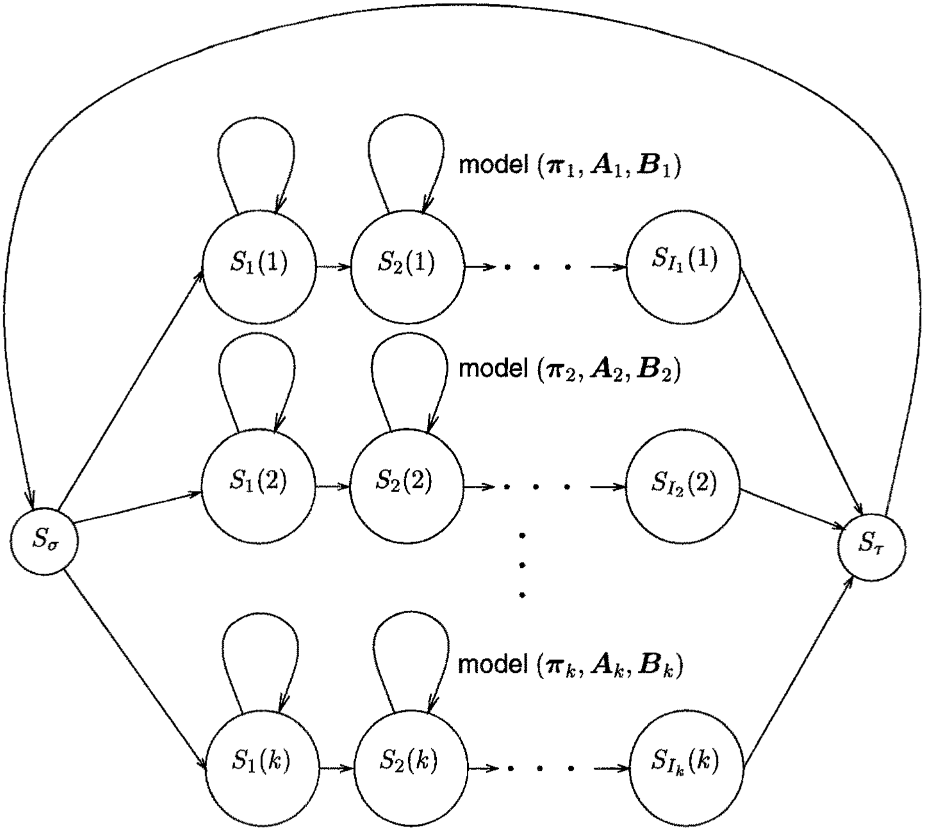


Figure 2.9: Looped HMM for continuous speech recognition, after [Kuh95b]. The models of the classes Ω_κ are put in parallel.

the k individual class HMMs $(\boldsymbol{\pi}_\lambda, \boldsymbol{A}_\lambda, \boldsymbol{B}_\lambda)$ by putting them in parallel, connecting their final states with the state S_τ connecting this in a loop backwards with state S_σ and finally connecting S_σ with the initial states of the HMMs $(\boldsymbol{\pi}_\lambda, \boldsymbol{A}_\lambda, \boldsymbol{B}_\lambda)$. The transition probabilities $a_{\sigma i}$ from the state S_σ to the initial states S_i of the class HMMs are set to $1/k$. The transition probability $a_{\tau\sigma}$ should be one but it is in practice set to a weight $a_{\tau\sigma} = \varpi \in \mathbb{R}$ to balance the number of deletions and insertions in the recognized sequence $\boldsymbol{\Omega}$. The value ϖ is usually determined heuristically. With respect to recognition, the computation of $P(\boldsymbol{o}|\boldsymbol{\pi}, \boldsymbol{A}, \boldsymbol{B})$ is not useful for the looped HMM. Instead one is interested in the optimal state sequence $\boldsymbol{s}^*$ associated with a given observation sequence $\boldsymbol{o}$. This optimal state sequence is defined as follows:

$$\boldsymbol{s}^* = \underset{\boldsymbol{s}}{\operatorname{argmax}}\, P(\boldsymbol{o}, \boldsymbol{s}|\boldsymbol{\pi}, \boldsymbol{A}, \boldsymbol{B}) \,. \tag{2.38}$$

It defines

- the sequence $\boldsymbol{\Omega} = [\Omega_{\kappa_i}]_{1 \leq i \leq m}$, $1 \leq \kappa_i \leq k$ of recognized classes, and
- their position on the time axis, that is, the segmentation of the speech signal.

The states s_σ and s_τ are called empty states, that is, when arriving in one of these states no observation is emitted. To obtain the optimal state sequence an efficient search procedure is needed. Usually the *Viterbi* algorithm is used, cf. Section 2.5, but also the A* algorithm as described in Section 2.6 is sometimes used, cf. for example the word recognizer described in [Pau90]. It should be noted that the search is often forced to end in state S_τ at time t_T.

2.3.3 Parameter Estimation

Now, it remains to show how HMM parameters can be optimized given a training sample. We will first concentrate on the case where one observation sequence $\boldsymbol{o}$ is given and the parameters of an HMM are to be estimated so that the probability $P(\boldsymbol{o}|\boldsymbol{\pi}, \boldsymbol{A}, \boldsymbol{B})$ takes its maximum. Specifically, we are interested in the following maximization task:

$$(\boldsymbol{\pi}, \boldsymbol{A}, \boldsymbol{B})^* = \underset{(\boldsymbol{\pi},\boldsymbol{A},\boldsymbol{B})}{\operatorname{argmax}} \; P(\boldsymbol{o}|\boldsymbol{\pi}, \boldsymbol{A}, \boldsymbol{B}) \,. \tag{2.39}$$

We further assume that an initial $(\boldsymbol{\pi}, \boldsymbol{A}, \boldsymbol{B})$ is given. The following algorithm, called *forward–backward* algorithm, iteratively optimizes these parameters while keeping the HMM topology constant, which is defined by the states and the transitions between the states. Not to change the topology means essentially that all a_{ij} and all π_i, for which the equations $a_{ij} = 0$ and $\pi_i = 0$ hold, do not change.

Before we describe the algorithm we have to define a few auxiliary variables. The *backward* variables

$$\beta_{ni} = P(o_{n+1}, \ldots, o_T | s_n = S_i, \boldsymbol{\pi}, \boldsymbol{A}, \boldsymbol{B}) \tag{2.40}$$

are defined and computed analogously to the forward variables α_{ni} but starting with $\beta_{Ti} = 1$ and moving backwards along the edges in the trellis of Figure 2.8. The probability of being in state S_i at time t_n and in state S_j at time t_{n+1} is denoted by

$$\begin{aligned} p_{nij} &= P(s_n = S_i, s_{n+1} = S_j | \boldsymbol{o}, \boldsymbol{\pi}, \boldsymbol{A}, \boldsymbol{B}) \\ &= \frac{P(s_n = S_i, s_{n+1} = S_j, \boldsymbol{o} | \boldsymbol{\pi}, \boldsymbol{A}, \boldsymbol{B})}{P(\boldsymbol{o}|\boldsymbol{\pi}, \boldsymbol{A}, \boldsymbol{B})} \\ &= \frac{\alpha_{ni} a_{ij} b_j({}^{n+1}\boldsymbol{c}) \beta_{n+1,j}}{P(\boldsymbol{o}|\boldsymbol{\pi}, \boldsymbol{A}, \boldsymbol{B})} \,. \end{aligned} \tag{2.41}$$

With this, the probability γ_{ni} of visiting state S_i at time t_n given $\boldsymbol{o}$ and HMM is computed as

$$\gamma_{ni} = P(s_n = S_i | \boldsymbol{o}, \boldsymbol{\pi}, \boldsymbol{A}, \boldsymbol{B}) = \sum_{j=1}^{I} p_{nij} \,, \tag{2.42}$$

and $\sum_{n=1}^{T} \gamma_{ni}$ is the probability of visiting state S_i at any time t_n. This probability can be equivalently interpreted as the expected number of times S_i is visited while $\boldsymbol{o}$ is generated. In the case of continuous HMMs the variable γ_{ni} as defined in equation (2.42) is extended by considering the ν–th normal distribution contributing to b_j: The variable

$$\begin{aligned} \gamma_{ni\nu} &= \frac{P(s_n = S_i,\ \nu,\ \boldsymbol{o}|\boldsymbol{\pi}, \boldsymbol{A}, \boldsymbol{B})}{P(\boldsymbol{o}|\boldsymbol{\pi}, \boldsymbol{A}, \boldsymbol{B})} \\ &= \frac{\alpha_{ni}\beta_{ni}w_{i\nu}p({}^{n}\boldsymbol{c}|\mu_{i\nu}, \Sigma_{i\nu})}{P({}^{n}\boldsymbol{c}|\boldsymbol{\pi}, \boldsymbol{A}, \boldsymbol{B})} \,. \end{aligned} \tag{2.43}$$

is the probability of being in state S_i at time t_n and emitting ${}^{n}\boldsymbol{c}$ by the ν–th normal distribution.

With these variables and similar interpretations of the probabilities as expected numbers of events taking place while $\boldsymbol{o}$ is generated by this particular HMM an updated $(\boldsymbol{\pi}', \boldsymbol{A}', \boldsymbol{B}')$ is computed by the following reestimation formulas. We only consider the case of continuous HMMs with mixtures of normal distributions, because discrete HMMs will not be considered any further in this book:

$$\pi'_i = \gamma_{1i} \tag{2.44}$$

$$a'_{ij} = \frac{\sum_{n=1}^{T-1} p_{nij}}{\sum_{n=1}^{T-1} \gamma_{ni}} = \frac{\text{expected number of trans. from } S_i \text{ to } S_j}{\text{expected number of visits of } S_i} \tag{2.45}$$

$$w'_{i\nu} = \frac{\sum_{n=1}^{T} \gamma_{ni\nu}}{\sum_{n=1}^{T}\sum_{\nu=1}^{L_i} \gamma_{ni\nu}} \tag{2.46}$$

$$\mu'_{i\nu} = \frac{\sum_{n=1}^{T} \gamma_{ni\nu}}{\sum_{n=1}^{T}\sum_{\nu=1}^{L_i} \gamma_{ni\nu}} \, {}^{n}\boldsymbol{c} \tag{2.47}$$

$$\Sigma'_{i\nu} = \frac{\sum_{n=1}^{T} \gamma_{ni\nu}}{\sum_{n=1}^{T}\sum_{\nu=1}^{L_i} \gamma_{ni\nu}} ({}^{n}\boldsymbol{c} - \mu'_{i\nu})({}^{n}\boldsymbol{c} - \mu'_{i\nu})_t \,. \tag{2.48}$$

The reestimation of the mean vector $\mu_{i\nu}$ can be explained as the average over all T observed feature vectors ${}^n\boldsymbol{c}$ where the individual feature vectors are not multiplied with the uniform weight $1/T$ but they are weighted by the expected number of being in state S_i at time t_n and emitting ${}^n\boldsymbol{c}$. In other words, feature vectors which are likely to be observed when entering state S_i contribute to a large extent to the reestimated mean vector. These reestimation formulas can be interpreted as a realization of the EM algorithm [Dem77]. It can be shown that $P(\boldsymbol{o}|\boldsymbol{\pi}', \boldsymbol{A}', \boldsymbol{B}') > P(\boldsymbol{o}|\boldsymbol{\pi}, \boldsymbol{A}, \boldsymbol{B})$ and that the iterated reestimation of the HMM parameters lets $P(\boldsymbol{o}|\boldsymbol{\pi}, \boldsymbol{A}, \boldsymbol{B})$ converge to a local maximum. In the case of discrete HMMs the forward–backward algorithm is called *Baum–Welch* algorithm.

Usually a set of R observation sequences, each corresponding to one utterance, has to be used for a robust estimation of HMM parameters. The forward–backward algorithm as defined above can easily be extended to maximize

$$(\boldsymbol{\pi}, \boldsymbol{A}, \boldsymbol{B})^* = \underset{(\boldsymbol{\pi}, \boldsymbol{A}, \boldsymbol{B})}{\operatorname{argmax}} \prod_{\varrho=1}^{R} P({}^{\varrho}\boldsymbol{o}|\boldsymbol{\pi}, \boldsymbol{A}, \boldsymbol{B}) \,. \tag{2.49}$$

This equation still assumes the case of isolated speech recognition. In the case of continuous speech for each utterance the spoken sequence of classes $[\Omega_{\kappa_i}]_{1 \le i \le m}, 1 \le \kappa_i \le k$ has to be known. During the training for each utterance ${}^{\varrho}\boldsymbol{o}$ a large $(\boldsymbol{\pi}_{\varrho}, \boldsymbol{A}_{\varrho}, \boldsymbol{B}_{\varrho})$ is built by concatenating the small class HMMs $(\boldsymbol{\pi}_{\kappa_m}, \boldsymbol{A}_{\kappa_m}, \boldsymbol{B}_{\kappa_m})$. Then the forward–backward algorithm is used to optimize the set of HMMs $\{(\boldsymbol{\pi}_1, \boldsymbol{A}_1, \boldsymbol{B}_1), \ldots, (\boldsymbol{\pi}_k, \boldsymbol{A}_k, \boldsymbol{B}_k)\}$ which maximizes

$$\prod_{\varrho=1}^{R} P({}^{\varrho}\boldsymbol{o}|\boldsymbol{\pi}_{\varrho}, \boldsymbol{A}_{\varrho}, \boldsymbol{B}_{\varrho}) \,. \tag{2.50}$$

Recall that k is the number of alternatively recognizable classes, for example, the words out of a vocabulary. It is an important feature of this algorithm that no hand–segmented training data is needed. The time–alignment is, so to speak, optimized during the computation of the forward and backward variables.

HMMs have the advantage over NNs that with the forward–backward algorithm an efficient training method is known. A disadvantage of this algorithm is that it is a maximum likelihood method. Therefore, recently discriminative training algorithms have been developed for HMMs, which *maximize a posteriori probabilities* (MAP) [Lee95] or directly optimize the recognition rate by the *maximum mutual information* (MMI) criterion, cf. for example [Nor94b], as well as by the *generalized probabilistic descent* (GPD) algorithm [Jua92]. A drawback of these methods is that training efficiency of the forward–backward algorithm is lost. In this book we will only use HMMs trained with the forward–backward or the Viterbi algorithm, cf. Section 2.5.

2.4 n-gram Grammars

HMMs trained with the algorithm described in Section 2.3 model the likelihood

$$P(\boldsymbol{o}|\boldsymbol{\Omega}) = P(\boldsymbol{o}|\Omega_{\kappa_1}, \dots, \Omega_{\kappa_m}) = P(\boldsymbol{o}|\boldsymbol{\pi}, \boldsymbol{A}, \boldsymbol{B}) \,, \tag{2.51}$$

continuous ASR assumed. One is, however, interested in the probability $P(\boldsymbol{\Omega}|\boldsymbol{o})$ of a class sequence given an observation sequence. Therefore, according to the Bayes rule (2.2) the a priori probability for $P(\boldsymbol{\Omega})$ has to be estimated additionally. The a priori probabilities are estimated by employing n-gram grammars [Jel90]. Since these can be used to model any kind of symbol sequence, we will use the notation $\boldsymbol{v} = [v_n]_{1 \leq n \leq m}$ where v_n takes a symbol out of a finite symbol set with k elements: $v_n \in V = \{V_1, \dots, V_k\}$.

The basic principle of n-gram models is the following approximation of the probability $P(\boldsymbol{v})$ by the probabilities of sub–sequences of length n:

$$\begin{aligned} P(\boldsymbol{v}) &= P(\boldsymbol{v} = v_1 \dots v_m) = P(v_1) \prod_{i=2}^{m} P(v_i | v_1 \dots v_{i-1}) && (2.52) \\ &\approx P(v_1) \prod_{i=2}^{m} P(v_i | v_{i-n+1} \dots v_{i-1}) \,. && (2.53) \end{aligned}$$

The conditional n-gram probabilities in (2.53) are determined by the following maximum likelihood estimation:

$$\widehat{P}(v_n | v_1 \dots v_{n-1}) = \frac{\widehat{P}(v_1 \dots v_n)}{\widehat{P}(v_1 \dots v_{n-1})} \approx \frac{\#(v_1 \dots v_n)}{\#(v_1 \dots v_{n-1})} \tag{2.54}$$

where $\#(.)$ is the frequency of occurrence of events observed in a training sample.

A basic problem with this approach is to find a non–zero estimate for the probabilities $P(v_n|v_1 \dots v_{n-1})$ where the corresponding n-grams $(v_1 \dots v_n)$ have not been observed in the training data. In most applications even for small n and very large training samples the n-grams observed in the training will not cover all n-grams contained in the test data. In other words, in order to obtain a good approximation of $P(\boldsymbol{v})$ according to equation (2.53) n is chosen "as large as possible". This requires methods for the estimation of the probabilities for ***unseen*** n-grams; various approaches are known from the literature, overviews can be found in [Ney94a, ST95b].

In this book we will use the *polygram* approach as described in [Kuh94, ST95a], which does not only provide a mechanism for the estimation of probabilities of unseen events but, furthermore, increases model robustness by making use of n-grams of variable order. Each n-gram probability is interpolated by a linear combination of all lower order models defined by the particular n-gram according to

$$\tilde{P}(v_n|v_1\dots v_{n-1}) = \xi_0\frac{1}{k} + \xi_1\widehat{P}(v_n) + \sum_{i=2}^{n}\xi_i\widehat{P}(v_n|v_{n-i+1}\dots v_{n-1})\,. \tag{2.55}$$

The term $1/k$ approximates the zerogram probabilities, which are the probabilities for the unseen symbols v_n. Polygrams allow one to use arbitrary large values of n limited only by memory resources during computation. The interpolation coefficients ξ_i are estimated on a cross–validation sample, which is independent from training and test samples. The estimation is conducted iteratively on the basis of the EM algorithm, that is, the new coefficients ξ_i' are for $2 \leq i \leq n$ reestimated by the following formula:

$$\xi_i' = \frac{\sum\limits_{v_1\dots v_n\in W}\widehat{P}(v_n|v_{n-i+1}\dots v_{n-1})}{\sum\limits_{v_1\dots v_n\in W}\{\xi_0\frac{1}{k} + \xi_1\widehat{P}(v_n) + \sum\limits_{j=2}^{n}\xi_j\widehat{P}(v_n|v_{n-j+1}\dots v_{n-1})\}}\,\xi_i\,. \tag{2.56}$$

The reestimation formula for ξ_0 and ξ_1 is obtained by appropriately replacing the numerator in (2.56).

Further robustness of the models can be achieved by using *category–based n-grams* [Der86] where the probabilities $P(\boldsymbol{z} = z_1\dots z_n)$ of sequences of symbol categories are modeled by equation (2.53) [Der86]. In the special case of polygrams, categories must not overlap and build a partition of the symbol set V, that is, the set of categories is defined as $Z = \{Z_1\dots Z_K\}$ with $V = Z$ and $Z_i \cap Z_j = \emptyset$, $1 \leq i,j \leq K, i \neq j$ [Kuh95b, p. 100]. This implies that each symbol sequence $\boldsymbol{v}$ is mapped to a unique category sequence $\boldsymbol{z}$. With category–based polygrams the probabilities for symbol sequences are determined by

$$\begin{aligned} P(\boldsymbol{v}) &= P(\boldsymbol{v}|\boldsymbol{z})P(\boldsymbol{z}) \\ &\approx P(z_1)P(v_1|z_1)\prod_{i=2}^{m}P(z_i|z_{i-n+1}\dots z_{i-1})P(v_i|z_i) \end{aligned} \tag{2.57}$$

where the conditional probability for a symbol given a category is determined by the *Jeffrey* estimation:

$$\widehat{P}(v_n|z_n) = \frac{\#(v_n)+1}{\sum\limits_{v_i\in z_i}\{\#(v_i)+1\}}\,. \tag{2.58}$$

The quality of different n-gram models or the "complexity" of a corpus of symbol sequences, for example, a text corpus, is estimated by the *test set perplexity* [Jel90]. This is defined as

$$Q(v_1 \ldots v_m) = \frac{1}{\sqrt[m]{P(v_1 \ldots v_m)}} \tag{2.59}$$

and approximates the true perplexity which is in the case of regular grammars the average of the number of symbols which can follow a given symbol. If two n-grams are alternatively used in a speech recognizer, the one with the smaller perplexity often, but not necessarily, yields lower error rates.

Since each n-gram model is an approximation of the probability $P(\boldsymbol{v})$, better approximation might be achieved by mixtures of K different n-grams

$$P(\boldsymbol{v}) = \sum_{i=1}^{K} w_i P_i(\boldsymbol{v}) \tag{2.60}$$

as proposed for example in [ST95b]. An example for such a mixture of n-gram models is the *cache–based* natural language model described in [Kuh90]. It combines an n-gram trained on a large corpus with another n-gram, the cache, which only takes into account the recently seen events. This is based on the assumption that in a text or a discourse recently seen n-grams are likely to occur again. The cache can be interpreted as employing a *short–term memory*.

Mixtures of n-gram models as defined in equation (2.60) have also been used for classification purposes. Specifically, in [Mas95c] word sequences are classified into one out of k dialog acts according to the following decision rule:

$$\kappa = \underset{1 \le \lambda \le k}{\operatorname{argmax}} \frac{w_\lambda P_\lambda(\boldsymbol{v})}{\sum_{i=1}^{k} w_i P_i(\boldsymbol{v})} . \tag{2.61}$$

Another possibility of using n-gram models for classification purposes will be presented in Section 7.2.

A drawback of the approximation of $P(\boldsymbol{v})$ with n-gram statistics is that these only take local dependencies into account. However, it might be the case that the probability of a symbol does not depend on the immediate predecessors but it might be influenced by events lying further back in the past. To overcome these problems *permugram* models have been introduced in [ST95c]. These are n-grams operating on permutations of the symbol sequence $\boldsymbol{v}$. Alternatively, classification trees can be used to model long–term dependencies. This will be discussed in Section 2.8.

2.5 Viterbi Search

The Viterbi algorithm is a general *time–synchronous search* algorithm applicable to search spaces as defined by the trellis shown in Figure 2.8 [Vit67]. It is a special dynamic programming (DP) technique [Bel72], and is often used for determining the optimal state sequence which could have generated an observation sequence given an HMM. We will explain it based on this application, however, Viterbi search can be used and originally was used for other purposes as well. The following description is based on [Nie90a].

The Viterbi algorithm is similar to the computation of the forward variables α_{ni}, equation (2.36): The sum over all predecessor states is replaced by a maximization and furthermore a backward pointer Φ is used in each state to store the best predecessor:

1. Initialize

$$\begin{aligned} \delta_{1i} &= \pi_i b_i({}^1\mathbf{c}) \,, \\ \Phi_{1i} &= 0 \,, \quad 1 \le i \le I \,. \end{aligned} \tag{2.62}$$

2. Iteratively compute all other variables δ by the following recursion scheme

$$\delta_{n+1,j} = \max_{1 \le i \le I} \{\delta_{ni} a_{ij}\} b_j({}^{n+1}\mathbf{c}) \,, \quad 1 \le j \le I \,, \tag{2.63}$$

and keep track of the optimal predecessors by

$$\Phi_{n+1,j} = \underset{1 \le i \le I}{\mathrm{argmax}} \{\delta_{ni} a_{ij}\} \,, \quad 1 \le j \le I \tag{2.64}$$

3. Terminate by determining the optimal final state $s_T^* = S_i$ according to

$$i = \underset{1 \le i \le I}{\mathrm{argmax}} \{\delta_{Ti}\} \,. \tag{2.65}$$

4. By this the optimal state sequence

$$s^* = \underset{\boldsymbol{s}}{\mathrm{argmax}} \{P(\boldsymbol{s}, \boldsymbol{o} | \boldsymbol{\pi}, \boldsymbol{A}, \boldsymbol{B})\} \tag{2.66}$$

is given and can be determined by backtracking from $n = T - 1$ to $n = 1$ according to

$$s_n^* = \Phi_{n+1,i} \,, \quad \text{where i is chosen such that} \quad s_{n+1}^* = S_i \,. \tag{2.67}$$

This algorithm employs a *full search* in the state space, that is, effectively, all paths s contained in the search space are evaluated. Although this is done in an efficient manner, for large search spaces, as is the case in state–of–the–art word recognizers, the search space has to be pruned so that no full search is conducted. This is performed by the Viterbi *beam search* [Kuh95b, pp. 119–126]: In each time step t_n of the algorithm after the computation of (2.63) only the paths are further continued for which δ_{ni} is greater than the beam threshold defined as $\theta = \xi \cdot \max_j\{\delta_{nj}\}$, $0 \leq \xi < 1$. By this, usually at time instances t_n where the recognition is reliable, many paths are pruned. A consequence of this pruning is that it is not anymore guaranteed to find the optimal path.

In the Viterbi search n-gram models as defined in Section 2.4 can be directly employed. However, because at each state s_{ni} only links to immediate predecessors are established according to equation (2.63), only *bigram* models, which are n-grams with $n = 2$, can be used in an optimal way unless the algorithm described above is extended so as to capture larger histories, which, however, increases the search space considerably [Ney94b]. If such *bigram* models are used in the search, equation (2.63) is still used within class HMMs but the following equation is used when a transition crosses a class boundary [Kuh95b, p. 117]:

$$\delta_{n+1,j} = \max_{1 \leq i \leq I}\{\delta_{ni} a_{ij} P^{\xi}(v_l | v_{l-1})\} b_j({}^{n+1}\mathbf{c}) \,, \quad 1 \leq j \leq I \tag{2.68}$$

with v_l, v_{l-1} denoting the class corresponding to the model after/before the boundary, respectively. The weight ξ balances the influence of the bigram model and is determined on a cross–validation sample so as to minimize recognition error. "Crossing a class boundary" roughly means following the transition from state S_τ to state S_σ in the looped model shown in Figure 2.9, however, the topology of the looped HMM itself has to be modified for the use of bigrams according to [Kuh95b, Sec. 5.1.2].

2.6 A* Search

The *A** *algorithm* is a general heuristic *state–space* search procedure which can be used for a variety of problems in artificial intelligence or pattern analysis [Nil80, Pea84]; the following description is based on [Nie90a]. As we will see below, it has the advantage over the Viterbi search that under mild conditions it is guaranteed to find an optimal solution without exploring the full search space. This is due to the fact that the cost function used during the search is better informed by taking into account not only information about what has been analyzed so far, as it is done by the δ_{ni} in equation (2.63), but it also utilizes an estimate of the costs which can be expected during the remaining analysis.

The A* algorithm operates on a *search graph.* Each node n_i in the graph represents a hypothesis about a partial analysis of a pattern. The edges of a graph are given by transformations, which define how a node can be transformed into other nodes. The set of all transformations applicable to a node is called *expansion* of a hypothesis. With a transformation of a node n_i into an immediate successor node n_j we associate costs $\vartheta(n_i, n_j)$.

To perform a search in a graph an initial node n_0 and a set of final or goal nodes N_g have to be defined. A sequence of transformations transferring the hypothesis attached to node n_0 to the hypothesis of any node n_i is called a *search path.* A *solution path* is a search path leading from n_0 to an $n_g \in N_g$. In general, several solution paths my lead through a particular node n_i to the same goal n_g or to different goals. The goal of the search is to find the solution path with minimal costs. A *cost function* $\varphi(n_i)$ gives the costs of the optimal solution path passing n_i. This cost function can be split according to

$$\varphi(n_i) = \psi(n_i) + \chi(n_i) \tag{2.69}$$

where $\psi(n_i)$ are the costs of an optimal path from n_0 to n_i and $\chi(n_i)$ are the *remaining* costs for the optimal path from n_i to a node from N_g. In general, the costs do not have to be additive [Sag90], but in the following only this case will be considered. Since usually the above functions are unknown when arriving in a node n_i on any search path, they have to be replaced by estimates so that equation (2.69) changes to

$$\widehat{\varphi}(n_i) = \widehat{\psi}(n_i) + \widehat{\chi}(n_i)\ . \tag{2.70}$$

Based on these estimates of cost functions, the A* algorithm is defined in Figure 2.10. The set OPEN collects the end nodes of expanded paths. Expanded nodes are kept on the CLOSED set so that paths ending in the same node n_i can be recombined. For efficiency it is useful to implement OPEN as a list sorted by the cost estimates $\widehat{\varphi}(n_i)$; this means, each new node is inserted in the position defined by this order. The optimal estimate $\widehat{\psi}(n_i)$ are the actual costs from n_0 to n_i, that is, $\widehat{\psi}(n_i) = \widehat{\psi}(n_j) + \vartheta(n_i, n_j)$. In general, to find a good estimate $\widehat{\chi}(n_i)$ is a problem. We define an estimate of remaining costs $\widehat{\chi}_1(n_i)$ as being

- *optimistic* if it is smaller than the actual remaining costs, that is, $\widehat{\chi}_1(n_i) \leq \chi(n_i)$, and

- *monotonous*, if $\widehat{\chi}_1(n_i) - \widehat{\chi}_1(n_j) \leq \vartheta(n_i, n_j)$ for immediate successors n_j of n_i,

```
Initialize: OPEN $= n_0$, CLOSED $= \emptyset$.
WHILE OPEN $\neq \emptyset$
    Determine $n_i$ with minimal $\hat{\varphi}(n_i)$.
    OPEN = OPEN$\backslash\{n_i\}$.
    CLOSED = CLOSED $\cup$ $\{n_i\}$.
    IF    $n_i \in N_g$
    THEN  Determine the optimal path from $n_0$ to $n_i$ by backtracking.
          STOP
    ELSE  Expand $n_i$.
          FOR each successor $n_j$ of $n_i$
              Compute $\hat{\varphi}(n_j)$.
              IF    $n_j \notin$ OPEN AND $n_j \notin$ CLOSED
              THEN  OPEN = OPEN $\cup$ $\{n_j\}$.
                    Set backwards pointer from $n_j$ to $n_i$.
              ELSE  IF    $\hat{\varphi}({}^{NEW}n_j) < \hat{\varphi}({}^{OLD}n_j)$
                    THEN  Set backwards pointer from $n_j$ to $n_i$.
                          IF    $n_j \in$ CLOSED
                          THEN  Recompute $\hat{\varphi}(n_l)$ for all successors $n_l$
                                of $n_j$.
```

Figure 2.10: A* algorithm.

- better *informed* than another estimate $\hat{\chi}_2(n_i)$ if $\hat{\chi}_1(n_i) \geq \hat{\chi}_2(n_i)$ for all nodes n_i.

It can easily be seen that monotonous remaining costs imply monotonous cost estimates $\hat{\varphi}(n_i)$ in the sense that $\hat{\varphi}(n_i) \leq \hat{\varphi}(n_j)$ if n_j is any successor of n_i. If $\hat{\chi}(n_i)$ is optimistic the A* algorithm has the following properties:

- It *terminates* if a solution path exists.

- It is *admissible*, that is, the solution path found when the algorithm terminates represents an optimal solution.

- When n_i is expanded, an optimal path to n_i has already been found, that is, $\hat{\psi}(n_i) = \psi(n_i)$. As a consequence, the costs of nodes on CLOSED have never to be recomputed.

- The more informed the remaining costs are the more *efficient* is the search, that is, less nodes are expanded until the search terminates.

In general, to find a well informed $\hat{\chi}(n_i)$ is difficult. A $\hat{\chi}_1(n_i)$ being better informed than another estimate $\hat{\chi}_2(n_i)$ might involve a significant increase in computational effort which might even exceed the gain in search effort in terms of the number of expanded nodes. In practice, one has to find a trade–off between these two factors. In the case of uninformed search, that is, with $\hat{\chi}(n_i) = 0$ for all n_i the A* search is equivalent to dynamic programming and explores the full search space in a *breadth–first* manner.

Note that in the literature the nodes of a search graph are often called *states*, which, however, should not be confused with HMM states. If A* is applied to HMM based continuous speech recognition, a state in the search space is not necessarily identical with an HMM state. If n-gram language models for word sequences as defined in Section 2.4 with $n > 2$ are used in the search, a node in the search graph in general corresponds to a path in the trellis, cf. Figure 2.8, defined by a sequence $s_1 \ldots s_n$ of HMM states.

In general A* is much more flexible than Viterbi, because it allows the use of higher order knowledge sources, as higher order n-grams, and the search is better informed in the sense as defined above. This, however, is at cost of an increased amount in memory resources and some computational overhead for the management of the OPEN list. As for other search strategies cf. [Pea84].

2.7 NN/HMM Hybrid Systems

Various groups investigated the combination of NNs and HMMs in an integrated system for ASR. The motivation is to make use of the advantages of both NNs and HMMs:

- NNs can model arbitrarily distributed input features, and
- HMMs are better suited for the analysis of sequences of feature vectors.

Even recurrent NNs have in practice been proven to have very limited performance in modeling sequences. This empirical observation has been theoretically justified in [Ben94]. Two types of NN/HMM hybrid systems will be described in the following. In the first one, the NN performs a feature transformation which is jointly optimized with the HMM parameters; the transformed features are modeled with normal distributions as observation densities on the HMM side. In the second approach, the NN is directly used to compute HMM observation probabilities under the assumption the NN output activations are probabilities. Related hybrid approaches developed for ASR as *multi–state TDNNs* [Haf90], *alphanets* [Bri90a, DV94], or the *hierarchical mixtures of experts* / HMM hybrids [Zha95] will not be considered in this book.

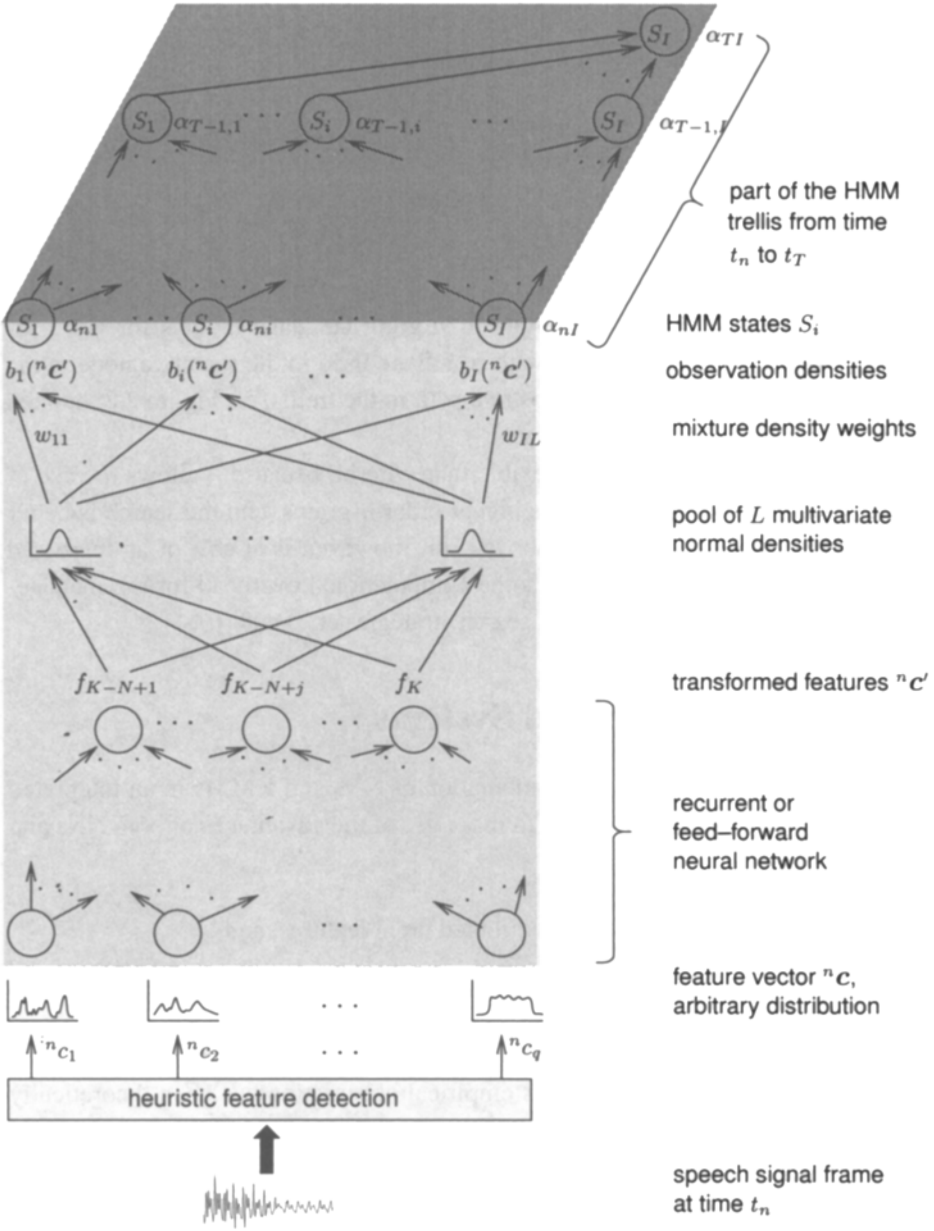


Figure 2.11: Structure of an NN–Feat/HMM hybrid.

2.7.1 NN for Feature Transformation: NN–Feat/HMM

In [Ben92a, Ben95] an NN/HMM hybrid has been proposed, where the NN transforms arbitrarily distributed feature vectors so that they can be modeled by multivariate normal distributions by the HMM. We will limit the following explanations to semi–continuous HMMs. Furthermore, without loss in generality we assume that all computation in the HMM ends at time t_T in state S_I, that is, only state sequences $\boldsymbol{s} = (s_1 \ldots s_{T-1} S_I)$ are considered. This also implies that $P(\boldsymbol{o}|\boldsymbol{\pi}, \boldsymbol{A}, \boldsymbol{B}) = \alpha_{TI}$ instead of equation (2.37). In Figure 2.11 the structure of such a system is depicted. The dark shaded area shows the part of the HMM trellis corresponding to the time interval from t_n to t_T. The light shaded area gives an overview of the computational steps during recognition at time t_n, which are as follows:

1. Compute a heuristic feature vector ${}^{n}\boldsymbol{c}$ based on a frame of the speech signal. The feature vectors are not restricted to any specific distribution, they can be larger than is suitable in practice for HMM modeling, and the individual features $(c_1, \ldots, c_q)$ may be correlated.

2. This feature vector ${}^{n}\boldsymbol{c}$ is used as input to the NN.

3. The NN performs a transformation of the features: ${}^{n}\boldsymbol{c} \mapsto {}^{n}\boldsymbol{c}'$. That means the NN output activations are considered as features: ${}^{n}\boldsymbol{c}' = (f_{K-N+1}, \ldots, f_K)$. Recall that an NN has K nodes including N output nodes. The features $\boldsymbol{c}'$ are assumed to be better suited for modeling by the HMM observation densities b_i than the original features $\boldsymbol{c}$. This is because the NN can perform a data reduction and can be trained in a way that the ${}^{n}\boldsymbol{c}'$ are distributed according to a mixture of L multivariate normal densities, cf. below.

4. Compute the probabilities $p({}^{n}\boldsymbol{c}|\boldsymbol{\mu}_\nu, \Sigma_\nu)$ according to the $1 \leq \nu \leq L$ multivariate normal densities.

5. Determine for each HMM state S_i the observation probability $b_i({}^{n}\boldsymbol{c}')$, which is a mixture of these L densities:

$$b_i({}^{n}\boldsymbol{c}') = \sum_{\nu=1}^{L} w_{i\nu} p({}^{n}\boldsymbol{c}|\boldsymbol{\mu}_\nu, \Sigma_\nu) \, . \tag{2.71}$$

This computation is done for each time step $t_n, 1 \leq n \leq T$. Then the HMM forward variables α_{ni} are computed as shown in equations (2.35) to (2.37).

If in the system as shown in Figure 2.11 the NN level were omitted such that ${}^{n}\boldsymbol{c}' = {}^{n}\boldsymbol{c}$, one obtains a normal semi–continuous HMM. This of course requires that the feature vectors are computed in a way that they are distributed according to a mixture of normal densities. For efficiency, one often restricts the covariance matrices Σ_ν of the multivariate normal densities in HMMs to diagonal matrices,

```
Initialize the NN by back–propagation on a frame–wise labeled training sample.
Train the HMM parameters with EM on the sequence of feature vectors $[{}^n\mathbf{c}']_{1\leq n\leq T}$ computed by the NN.
WHILE the recognition accuracy improves, *jointly optimize* the hybrid:
    Compute the partial derivatives $\frac{\partial\alpha_{TI}}{\partial({}^nf_{K-j+1})}$ of the likelihood computed by the HMM with respect to the NN output activations.
    Optimize the NN with respect to these derivatives using back–propagation.
    Train the HMM on $[{}^n\mathbf{c}']_{1\leq n\leq T}$ with the EM algorithm.
    Determine the recognition accuracy.
```

Figure 2.12: Training of an NN–Feat/HMM hybrid. The NN parameters are optimized with respect to the HMM likelihood α_{TI}.

hence, assuming uncorrelated features. In the hybrid system this can be achieved by initializing the weights of the last layer of the NN to perform the multiplication by a matrix which does a *principal axis transformation* [Mid60] of the feature space. For this the activation function of the output nodes should be the identity $\Theta(y) = y$ instead of the sigmoid function.

The *training* of the hybrid system is conducted according to the algorithm shown in Figure 2.12. The NN has first to be trained supervised by back–propagation on speech samples, where each time frame t_n is labeled. Labels can be classes, for example, phones, or they can be sets of properties, being characteristic for each of the classes. For the purpose of phone recognition in [Ben92b] the *manner* and *place* of articulation of phones was used as properties, cf. also [Lad82]. Each NN output node is associated with one of the classes or properties. Training is performed with a desired output vector $({}^n\delta_1, \ldots, {}^n\delta_N)$ for each time step t_n such that $\delta_j = 1$ if the label includes the j^{th} class or property, and $\delta_j = 0$ else.

The goal of the *joint optimization* of the NN and HMM is to optimize the NN with respect to the likelihood α_{TI} computed by the HMM that means to optimize the likelihood by the NN training. This optimization is achieved by gradient descent, that is, by the partial derivatives $\frac{\partial\alpha_{TI}}{\partial({}^nf_{K-j+1})}$ of the likelihood computed by the HMM with respect to the NN output activations $(1 \leq n \leq T, 1 \leq j \leq N)$. For this we define the NN error function ϵ such that

$$\frac{\partial\epsilon}{\partial({}^nf_{K-j+1})} = -\xi\frac{\partial\alpha_{TI}}{\partial({}^nf_{K-j+1})}\ , \tag{2.72}$$

where ξ is a heuristic weight. Since according to equation (2.21) $\frac{\partial \epsilon}{\partial ({}^n f_{K-j+1})} = 2({}^n\delta_j - {}^n f_{K-j+1})$ for $1 \le j \le N$, we get as new desired output for the NN:

$$ {}^n\delta_j = {}^n f_{K-j+1} - \frac{\xi}{2} \frac{\partial \alpha_{TI}}{\partial ({}^n f_{K-j+1})} \tag{2.73} $$

Optimization of the NN with respect to the likelihood α_{TI} means to update the NN weights by back–propagation using this desired output.

This joint optimization of HMM and NN modifies the NN weights such that the HMM likelihood α_{TI} is maximized. This also forces the NN output activations, that is, the feature vectors ${}^n\boldsymbol{c}'$, to be distributed according to mixtures of multivariate normal densities. The HMM is still optimized with EM and the NN using gradient descent.

Note that normally a set of R utterances with a corresponding set of sequences of feature vectors $\{[{}^{\varrho n}\boldsymbol{c}]_{1 \le n \le T_\varrho}, 1 \le \varrho \le R\}$ is used for training the models. The index ϱ has been omitted in the algorithm described above for simplicity. In the case of continuous speech the likelihood α_{TI} refers to the training HMM $(\boldsymbol{\pi}_\varrho, \boldsymbol{A}_\varrho, \boldsymbol{B}_\varrho)$ built for each of the utterances, cf. page 39.

The partial derivatives $\frac{\partial \alpha_{TI}}{\partial ({}^n f_{K-j+1})}$ are computed for all $1 \le n \le T$ and for all $1 \le j \le N$ on the basis of the HMM forward and backward variables as well as using the derivative of the normal densities as

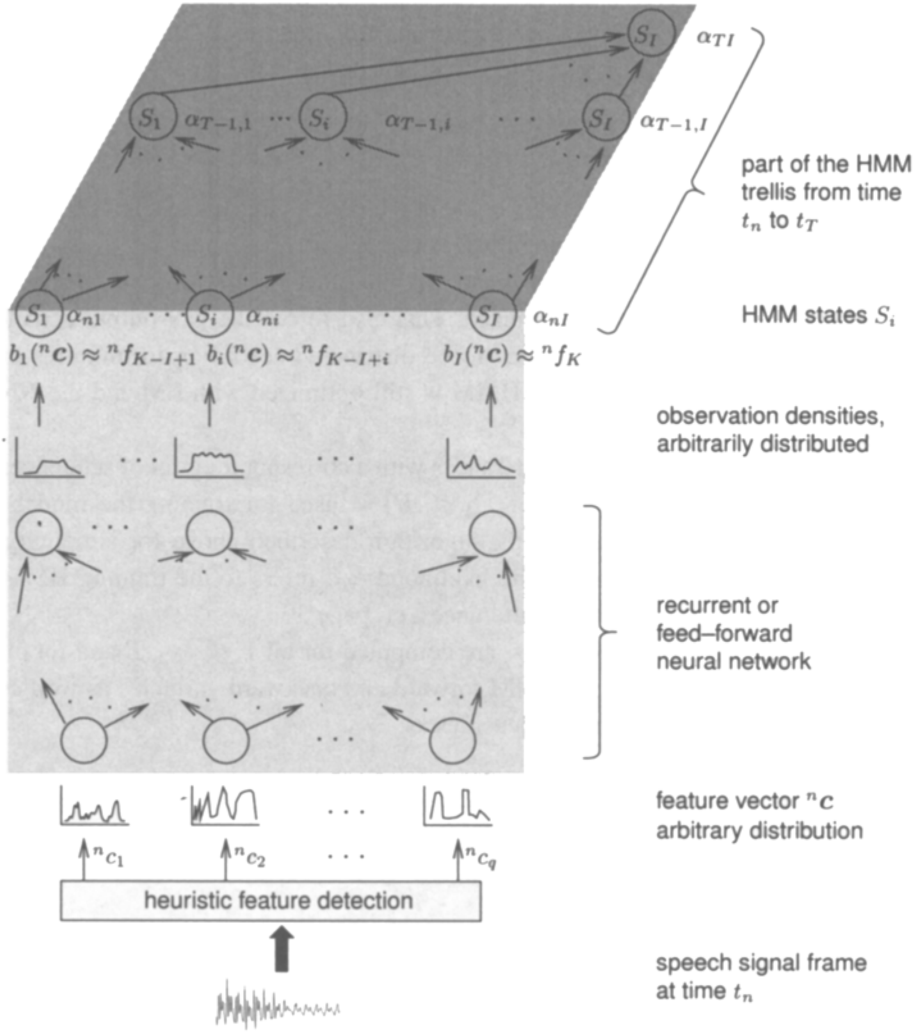
$$ \frac{\partial \alpha_{TI}}{\partial ({}^n f_{K-j+1})} = \frac{\partial \alpha_{TI}}{\partial ({}^n c'_j)} = \sum_{i=1}^{I} \frac{\partial \alpha_{TI}}{\partial b_i({}^n\boldsymbol{c}')} \frac{\partial b_i({}^n\boldsymbol{c}')}{\partial ({}^n c'_j)} , \tag{2.74} $$

$$ \frac{\partial \alpha_{TI}}{\partial b_i({}^n\boldsymbol{c}')} = \beta_{ni} \frac{\alpha_{ni}}{b_i({}^n\boldsymbol{c}')} , \tag{2.75} $$

$$ \frac{\partial b_i({}^n\boldsymbol{c}')}{\partial ({}^n c'_j)} = \sum_{\nu=1}^{L} w_\nu \Big(\sum_{l=1}^{q} d_{\nu,lj}(\mu_{\nu l} - {}^n c_l)\Big)\, p({}^n\boldsymbol{c}|\boldsymbol{\mu}_\nu, \boldsymbol{\Sigma}_\nu) , \tag{2.76} $$

with equation (2.76) being the derivative of a mixture of normal densities and $d_{\nu,lj}$ the component (l, j) of the matrix $\boldsymbol{\Sigma}_\nu^{-1})$, which is the inverse of the covariance matrix of the ν–th normal density. A detailed derivation can be found in [Ben92a]. The weight ξ in the above algorithm normalizes the partial derivatives. It has to be heuristically set so that "most" of the desired outputs as computed by equation (2.73) can be expected to be between zero and one. Otherwise one would try the NN to learn a mapping which it actually cannot perform due to the range of the sigmoid activation function.

As stated above, for the initial NN training frame–wise labeled sample data are needed. However, during the joint optimization it is sufficient if the sequence of classes, for example, words, is given as usually in HMM training. Therefore, the labeled training sample can be a small subset of the entire training data.



S_I α_{TI}
S_1 $\alpha_{T-1,1}$ S_i $\alpha_{T-1,i}$ S_I $\alpha_{T-1,I}$
part of the HMM trellis from time t_n to t_T
S_1 α_{n1} S_i α_{ni} S_I α_{nI}
HMM states S_i
$b_1(^n\mathbf{c}) \approx {}^n f_{K-I+1}$ $b_i(^n\mathbf{c}) \approx {}^n f_{K-I+i}$ $b_I(^n\mathbf{c}) \approx {}^n f_K$
observation densities, arbitrarily distributed
recurrent or feed–forward neural network
feature vector $^n\mathbf{c}$ arbitrary distribution
$^nc_1$ $^nc_2$ ^nc_q
heuristic feature detection
speech signal frame at time t_n

Figure 2.13: Structure of an NN–Prob/HMM hybrid.

Another approach of joint optimization of feature transformation and HMM has been described in [ST95d]. In contrast to the approach described above this one uses a *linear* transformation of feature vectors for the purpose of data reduction.

2.7.2 NN as Probability Estimator: NN–Prob/HMM

In contrast to the approach presented in the previous section the NN/HMM hybrid system described in the following assumes that NNs compute probabilities. We will give a summary of the baseline system. For an overview cf. [Mor95], a detailed survey can be found in [Bou94b]. This is the hybrid system described above used to model sequences of feature vectors $[{}^n\boldsymbol{c}]_{1\leq n\leq T}$. The structure is depicted in Figure 2.13. Input to the NN at time step t_n is a feature vector ${}^n\boldsymbol{c}$. The NN has the same number I of output nodes as the HMM has states. Each output node of the NN corresponds exactly to one HMM state.

The NN output activations are assumed to be a posteriori probabilities of being in an HMM state given a feature vector:

$$P(S_i|{}^n\boldsymbol{c}) = {}^nf_{K-I+i} \tag{2.77}$$

Applying the Bayes rule one achieves HMM observation densities

$$b_i({}^n\boldsymbol{c}) = P({}^n\boldsymbol{c}|S_i) = \frac{p_i P(S_i|{}^n\boldsymbol{c})}{p({}^n\boldsymbol{c})} = \frac{p_i \cdot {}^nf_{K-I+i}}{p({}^n\boldsymbol{c})} \,. \tag{2.78}$$

With this assumption, one obtains arbitrary observation densities $b_i({}^n\boldsymbol{c})$ which do not restrict the feature vectors to a particular distribution. The probability p_i is the a priori probability for state S_i. The observation probabilities are more or less the normalized NN output values, because the probability p_i is constant and the probability $p({}^n\boldsymbol{c})$ is constant for all states S_i. Based on these observation densities the HMM forward variables are computed as usual, cf. equation (2.36).

Training of this system is done by the algorithm shown in Figure 2.14. Again, a small sample with frame–wise labels is required to initialize the NN. Usually it is assumed that one output node of the NN corresponds to one label class. For example if such a hybrid is used for word recognition the labels can be phones. Afterwards, on a larger training sample the NN desired outputs are determined on the basis of the optimal HMM state sequence computed by the Viterbi algorithm. The desired output values along this state sequence are set to one, all others are set to zero. With this procedure one also obtains frame–wise state labels which are used to reestimate the a priori probabilities p_i. In [Bou92] this approach has been improved by using more than one HMM state or NN output node, respectively, per available phone label, hence, achieving a better (context–dependent) modeling.

Recently, extensions to this hybrid system have been developed which allow for a "finer" determination of the desired output values being somewhere between zero and one. In [Bou94a, Bou95b] an EM–like algorithm has been presented which determines NN desired outputs which are estimates of probabilities $P(s_n = S_i|s_{n-1} = S_j, {}^n\boldsymbol{c})$. In [Rei94] the NN desired outputs are determined based on the GPD algorithm, which also means that the whole system is trained so as to minimize the recognition error. In [Haf94] this hybrid approach has been extended in a way that the HMM part, that is, the modeling of the state transitions,

```
Initialize the NN on a (small) labeled subset of the training sample by back–propagation.
Initialize the a priori probabilities $p_i$ for the states $S_i$ on this labeled sample.
    Compute the NN outputs on the entire training sample.
    Reestimate the HMM transition probabilities with the Baum–Welch algorithm (2.46).
    Compute the optimal state sequence $s^*$ by Viterbi.
    Reestimate the $p_i$ based on the number of times the optimal sequence $s^*$ visits state $S_i$.
    Set the desired output vectors $[{}^n\delta]_{1 \leq n \leq T}$ of the NN as follows:
        $\delta_i = \begin{cases} 1 & \text{if } s_n^* = S_i \\ 0 & \text{else} \end{cases}$
    Train the NN with back–propagation.
UNTIL  no improvement in recognition accuracy
```

Figure 2.14: Training of an NN–Prob/HMM hybrid.

is fully integrated in the NN, and the resulting system is jointly optimized by MMI estimation.

2.8 Multi–level Semantic Classification Trees (MSCTs)

In Section 2.4 we mentioned that a drawback of modeling the probabilities $P(\boldsymbol{v})$ of symbol sequences $\boldsymbol{v}$ by n-gram statistics lies in the limited capability of taking into account long–term dependencies. In [Bah89] classification trees (CTs) were used to model such dependencies within word sequences. CTs are a well known approach for the classification of, usually fixed length, feature vectors [Bre84]. They are considered to be especially useful if no assumptions about the distribution density of the feature vectors can be made or if the feature vectors are high dimensional or inhomogeneous. A more flexible application of CTs to natural language modeling than the [Bah89] approach has been developed by Roland Kuhn and Renato de Mori and will be described in this section [Kuh93, Kuh95a]. These so called *semantic classification trees* (SCTs) are CTs for the classification of word or, to be more general, symbol sequences, which can be of variable length. They were developed to solve ASU tasks and are, as we will see below, capable of learning semantic rules from sample data. The SCT approach was extended to *multi–level semantic classification trees,* (MSCTs) which combine the CT and the SCT approaches, allowing questions about symbol sequences as well as about continuous valued features attached to the symbols [Geb95, Nöt96a]. The important extension of SCTs over the classical CT approach is the design of types of questions which allow the semantic modeling of language. These questions are built by the use of *regular expressions* encoding symbol sequences. In the following we will refer to these questions as *SCT–type* questions whereas all other questions will be called *CT–type* questions.

2.8.1 Semantic Classification Trees

SCTs can be used to classify entire utterances, for example, they decide whether an utterance is a question about air travel fares, or they classify parts of an utterance (*substrings*), for example, whether an airport mentioned in an utterance is a start, stop or destination airport. For this classification to take place, the temporal structure of a sequence of symbols $\boldsymbol{v} = v_1 v_2 \dots v_m$ is analyzed by means of regular expressions.

The structure of a binary SCT is as follows: each non–terminal node n_r consists of a YES/NO question, a YES subtree and a NO subtree; each node is labeled with a probability vector for the recognizable classes. The classification of a symbol sequence $\boldsymbol{v}$ begins with the question at the root of the SCT. Depending on the answer to the question, the YES or NO subtree respectively will be entered. This process is repeated until a leaf node has been reached. The probability vector of this leaf is then assigned to $\boldsymbol{v}$.

A possible question q_1 at the beginning of the analysis (at the root node) is:

Has $\boldsymbol{v}$ *the structure* $< +v_i+ >$ *?*

where $+$ is a non–zero gap, that is, an unknown sequence of symbols of length ≥ 1, and v_i is a symbol out of a given vocabulary. If the symbol v_i occurs exactly once in $\boldsymbol{v}$ and neither at the beginning nor at the end, q_1 is answered by YES, otherwise by NO. If q_1 is answered by YES, the known structure of $\boldsymbol{v}$ is $< +v_i+ >$. If q_1 is answered by NO, the known structure of $\boldsymbol{v}$ is the previously known structure, that is, "$+$" in the example. It is important to remark that a question handles exactly one unknown part of a sequence of symbols, that is, one gap of the known structure. Hence, a possible question q_2 to symbol sequences which have the structure indicated by a YES answer to q_1 is

Has $\boldsymbol{v}$ *the structure* $< v_j + v_i+ >$ *?*

information level	${}^{1}c$	${}^{2}c$	${}^{3}c$	${}^{4}c$	${}^{5}c$
1 (syntactic category)	pronoun	verb	adverb	preposition	noun
2 (word)	it	is	okay	on	Friday
3 (accentuation)	0.3	0.1	0.9	0.6	0.8
4 (boundary)	0.0	0.2	0.3	0.1	1.0

Table 2.1: Multi–level information of the utterance It is okay on Friday?

Question q_2 tests whether the first of the two gaps in $< +v_i+ >$ has the structure v_j+.

To analyze a sequence of symbols in a unique way by regular expressions, six different types of questions are needed: the *join-*, *left–*, *right–*, *unique–*, *twin–*, and *non–adjacent*–questions, as shown in [Kuh95a]. In the example, q_1 is a *unique–*question.

2.8.2 Multi–level Information

The only information used for the classification done by standard SCTs is a symbol (word) sequence corresponding to the utterance. However, additional information can be attached to these symbols. This can be word category information dynamically obtained by a *parts–of–speech tagger* or taken from a lexicon as well as additional acoustic information; for parts–of–speech tagging cf. [Bla92]. In general MSCTs are used to classify sequences of feature vectors $[{}^{n}\boldsymbol{c}]_{1\leq n\leq T}$ being mapped to a single class. The individual features ${}^{n}c_i$ can be continuous or discrete. The index i thereby denotes the information level. Each discrete information level contains symbols from a level–specific set V_i. In the discrete case feature values c_i are assumed to refer to symbols, that is, ${}^{n}c_i \in V_i$ with $V_i = \{v_{i1}, \ldots, v_{iL}\}$ being a finite set of symbols, for example, words or word categories.

Examples for such multi–level information, which might be obtained from a spoken utterance, can be found in Tables 2.1 and 2.2. In the first level, symbols denoting the syntactic word category are given. The second level contains the words of the utterance. The third and fourth level indicate if the word is accented or if there is a sentence boundary after the word, respectively. The numbers encode the acoustic evidence for the presence or absence of these attributes. They can be computed by a feature detector or by a numerical classifier. In the following discussion we assume that large values correspond to the presence of the attribute. In this example the information across the levels attached to one word is summed up in the feature vector ${}^{n}\boldsymbol{c} = ({}^{n}c_1, {}^{n}c_2, {}^{n}c_3, {}^{n}c_4)$.

information level	^1c	^2c	^3c	^4c	^5c
1 (syntactic category)	pronoun	verb	adverb	preposition	noun
2 (word)	it	is	okay	on	Friday
3 (accentuation)	0.3	0.6	0.6	0.3	0.7
4 (boundary)	0.1	0.2	0.6	0.2	1.0

Table 2.2: Multi–level information of the utterance It is okay. On Friday?

The first level, containing the features $^nc_1$, is called the *entrance level.* In the current realization, the SCT–type questions, that is, the regular expressions, are exclusively applied at this level; CT–type questions are used on the other levels. The goal of an MSCT is to classify a particular word in an utterance. One word is classified at a time; the different words in an utterance are classified sequentially. For each of the words the same multi–level information determined for the entire utterance is usually used. This method is comparable to the substring classification in SCTs. In the regular expressions the word to be classified is marked by an asterisk. As an example, assume that we want to classify the word Friday in Table 2.1 as being accented or not. In this case, the question

- "*Has v the structure $< +adverb + noun^* >$?*" will be answered by YES, however, the question
- "*Has v the structure $< +adverb^* + noun >$?*" will be answered by NO.

An asterisk can be attached to a gap, that is, regular expressions like $< +^*v_i+ >$ are allowed and denote that the word to be classified is contained in the left most gap.

The CT–type question in a node is restricted to the known structure of the utterance, which is defined by the regular expression of the closest parent node containing such an expression. That is the feature vectors attached to the symbols contained in a regular expression are identified and all CT–type questions apply only to these feature vectors. We will define this a bit more formally: When a particular regular expression is used in a node n_r of a tree, the symbols contained in the expression are numbered from left to right by the sequence $[j] = 1, 2, 3, \ldots, J_r$. The gaps $+$ are not numbered. The following node–specific mapping $\nu_{r(j)}$ is setup dynamically, that is, when a tree is used for the classification of a particular utterance:

$$\nu_r(j) = \quad n\,, \qquad 1 \leq j \leq J_r, \text{ such that } {}^nc_1 \text{ corresponds exactly}$$

$$\begin{aligned} & \quad \text{to the } j^{th} \text{ symbol in the regular expression,} \\ \nu_r(j) = & \text{ undefined}\,, \quad \text{for all } j > J_r\,. \end{aligned} \tag{2.79}$$

Hence, this mapping associates the symbols in the regular expressions with the appropriate feature vectors. The index r is omitted in the following for simplicity. Consider, for example, the regular expression $< +adverb + noun >$ applied to the example given in Table 2.1. In this case the following definition holds:

$$\begin{aligned} \nu(1) &= 3 \\ \nu(2) &= 5 \\ \nu(j) &= \text{undefined}\,, \quad 2 = J < j\,. \end{aligned} \tag{2.80}$$

Assume now that we have given multi–level information for the utterance It is okay on next week Friday. The regular expression $< +noun + noun >$ used in a question results in a definition of $\nu(j)$ as

$$\begin{aligned} \nu(1) &= 6\,, \\ \nu(2) &= 7\,, \\ \nu(j) &= \text{undefined}\,, \quad 2 = J < j\,, \end{aligned} \tag{2.81}$$

because the sixth word in the utterance is the noun week and the seventh word is the noun Friday. A mapping $\nu_r(j)$ defined in a node n_r is inherited by the child node in the YES branch. The NO child inherits the $\nu_s(j)$ of the parent node n_s of the node n_r. If a node n_r has a CT–type question then the children nodes inherit the $\nu_r(j)$ of their parent node.

The following CT–type questions about the features ${}^{\nu(j)}c_i$ with $i > 1$ are realized so far in the approach described in [Nöt96a, Har96] and were used in the experiments described in Section 7.3:

- Questions about *discrete* features are:

  - *Is* ${}^{\nu(j)}c_i = v_{il}$*?*, or
  - *Is* ${}^{\nu(j)}c_i \in V_{ih}$*?*

  where $v_{il} \in V_i$ is a particular symbol and $V_{ih} \subset V_i$.

- Questions about *continuous* features are as follows where θ is a threshold, and $1 \leq j \leq J$:

  - *Is* ${}^{\nu(j)}c_i \leq \theta$*?*
  - *Is* ${}^{\nu(j)}c_i = \min_{1 \leq n \leq m}\{{}^{n}c_i\}$*?*

- *Is* ${}^{\nu(j)}c_i = \max_{1 \leq n \leq m} \{{}^{n}c_i\}$?
- *Is* ${}^{\nu(j)}c_i = \min_{1 \leq n \leq J} \{{}^{\nu(n)}c_i\}$?
- *Is* ${}^{\nu(j)}c_i = \max_{1 \leq n \leq J} \{{}^{\nu(n)}c_i\}$?

This means the questions either compare features with thresholds or with other features of the same level. In the latter, the questions determine if the value of a feature is the minimum or the maximum in a set of features. Currently a set of features can either cover the entire utterance or it contains only features attached to the symbols in the entrance level which are contained in the regular expression of the closest parent node containing a regular expression at all. Questions across levels are not allowed, otherwise the complexity would be too large.

In an MSCT the different types of questions can arbitrarily alternate. In Figure 2.15 an example of a part of an MSCT is shown. Since the MSCTs trained throughout the experiments described in Section 7.3 are very large, Figure 2.15 shows a manually designed MSCT. Rather than showing the probability vector for the classes in the leaf nodes, the label of the class with highest probability is given. This tree can be used for the classification of words as *accented* (Ω_1) or *unaccented* (Ω_2) based on sequences of feature vectors, examples of which are shown in Tables 2.1 and 2.2. MSCTs can be viewed as rule based systems, the rules of which and the order of their application can be trained automatically as we will see below. When descending in the tree depicted in Figure 2.15 from the root to a leaf by always following the YES branches the sequence of rules applied in the nodes of the tree can be described as shown in Figure 2.16. For example, with this tree the word okay in Table 2.1 is classified as accented (leaf node 5), the one in Table 2.2 is classified as unaccented (leaf node 6). The classification of the word Friday yields to leaf node 3 (unaccented) in the case of Table 2.1 and to leaf node 1 (accented) in the case of Table 2.2.

With these examples it should become obvious that MSCTs are classifiers mapping sequences of feature vectors of varying length to a class Ω_κ. If the utterance underlying Table 2.1 were It is okay next week Friday? instead of It is okay on Friday? the classification results for the words okay and Friday would remain the same provided the features in the different levels are the same for the intersection of words. This stands in contrast to other classifiers as normal density classifiers, NNs or CTs, which require single feature vectors or sequences of feature vectors of fixed length so that the total number of input features is constant. Furthermore, MSCTs as well as SCTs can represent language models which capture long distance dependencies where gaps can be left in place of unimportant parts of an utterance. Recall that n-grams only model local dependencies.

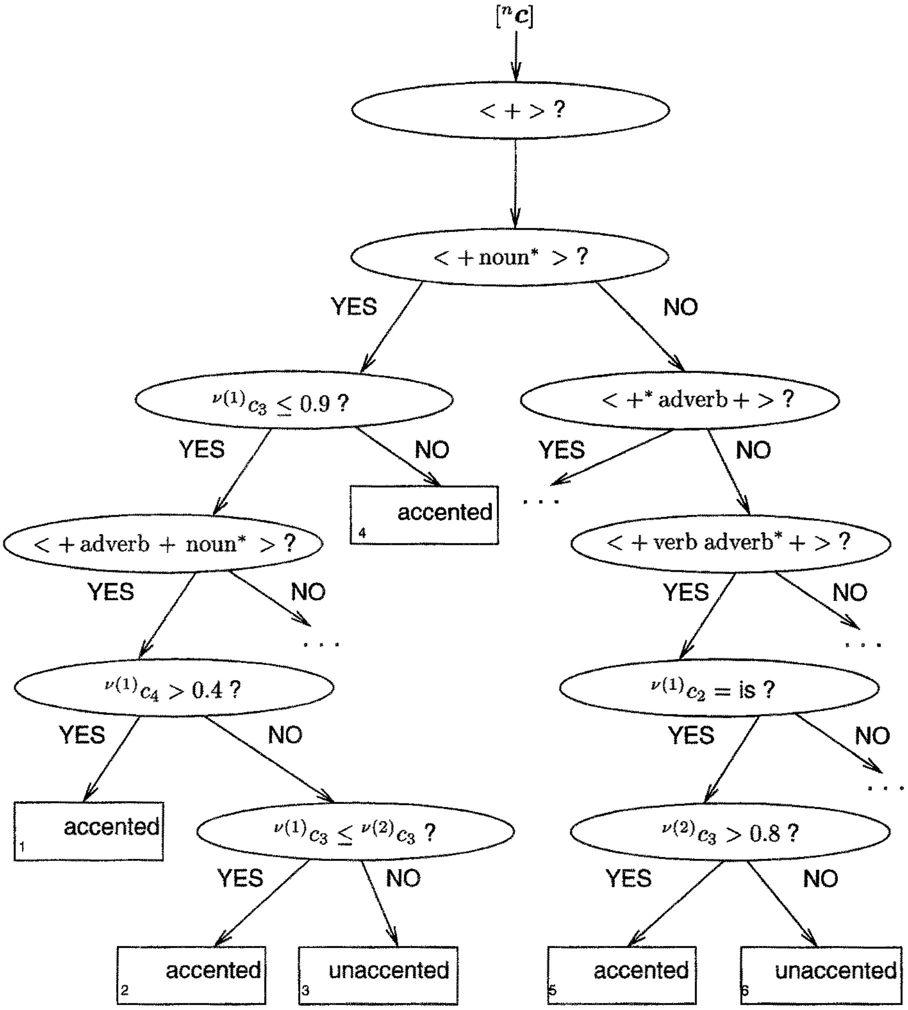


Figure 2.15: A multi–level semantic classification tree (MSCT).

2.8.3 The Training of MSCTs

The training of the MSCTs is done in the same way as described for CTs in [Gel91]. It is carried out by alternating expansion and pruning steps for an initial tree, using two disjunct sets of labeled training data $\omega = \omega' \cup \omega''$, $\omega' \cap \omega'' = \emptyset$. At first, the initial tree is expanded using set ω'. The result is a tree T_1. This tree is pruned by means of set ω'' which gives a tree T_2. By expanding T_2 with ω'', a tree T_3 is created. Pruning T_3 with ω' gives T_4. Continuing this process generates a sequence of trees $T_1, T_2, \ldots, T_K$. It stops if two subsequently pruned trees T_{2i} and $T_{2(i+1)}$ have the same structure, that is, they have the same number of nodes.

IF		the word to be classified is a noun
	AND	this noun is not the first word in the utterance
	AND	the evidence for accentuation of this word is smaller than 0.9
	AND	the word is somewhere preceded by an adverb
	AND	the adverb is not the first word in the utterance
	AND	the evidence for the adverb being succeeded by a clause boundary is greater than 0.4
THEN	the word is accented	

Figure 2.16: A rule encoded with the MSCT of Figure 2.15.

The resulting SCT is $T_{2(i+1)}$. For the expansion, two basic elements are needed [Bre84]:

- a set of possible YES/NO–questions that can be applied to the items of the task domain, and
- a rule for selecting the best question at any node or deciding that it should be a leaf node.

The set of possible questions about the entrance level is built up by employing all the symbols $v_{1l} \in V_1$ with the regular expressions proposed in [Kuh95a]. For all the other levels, the set of possible questions is constructed based on the CT type questions where all possible values for the indices n, i of $^{n}c_i$, for the thresholds θ_r, and for symbols v_{il} or the symbol sets V_{ih} are used. In the case of the thresholds the set of possible values can be defined by

$$\begin{aligned} \theta_r \in \; & \{ \min_{\{^{\varrho n}c\in\omega\}} \{^{\varrho n}c_i\} \\ & + \frac{l}{L}(\max_{\{^{\varrho n}c\in\omega\}} \{^{\varrho n}c_i\} - \min_{\{^{\varrho n}c\in\omega\}} \{^{\varrho n}c_i\}) \mid 1 \leq l \leq L\}, \end{aligned} \tag{2.82}$$

where L has to be predefined heuristically. Large values of L result in a fine modeling, but increase the training effort due to the number of questions to be tried. The set defined in (2.82) is a partition of the range of feature values $^{\varrho n}c_i$ of the entire training set into intervals of uniform length. A better solution would be a non–uniform partitioning as it can be obtained by a vector quantizer of the kind described in [Lin80]. The use of this would result in small intervals in regions where many $^{\varrho n}c_i$ are observed in the training data, and in larger intervals where few ${}^{en}c_i$ are observed. However, this has not been realized yet in the MSCT implementation used for the research presented in this book.

In [Nöt96a, Kuh95a] the *Gini*–impurity criterion has been used for MSCTs and SCTs as the rule to select the best question, which has already been described in [Bre84] for standard CTs. The impurity is a non–negative number. The best question is the one which yields in the greatest impurity decrease.

Expansion of a tree is done for all leaf nodes n_r recursively by the following algorithm:

1. Assign the best question q_r^* to the node n_r,

2. Create child nodes n_{YES} and n_{NO}, splitting the set of training items ω_r arriving at node n_r into ω_{YES} and ω_{NO} according to q_r^*

3. Expand the child nodes with the training set ω_{YES} and ω_{NO}, respectively.

If the decrease in impurity of a node is zero, this node is declared a final leaf node and will not be expanded anymore. Each node n_i in a tree is assigned the vector which contains for each of the classes the a posteriori probability $p(\Omega_\lambda | [{}^n\mathbf{c}], n_i)$ of a class given the multi–level information of the utterance, that is, the sequence of feature vectors $[{}^n\mathbf{c}]$, and given the node n_i or more precisely given the entire sequence of nodes from the root of the tree to n_i. The probability of a particular class Ω_λ is estimated by the the relative frequency of the training data items arriving at this node n_i and belonging to this class. Each node is, furthermore, labeled by the Ω_κ with the highest probability. Note that these probabilities should be determined on the union of both training sets $\omega = \omega' \cup \omega''$ at the end of the training.

To prune the expanded tree with a disjunct training set, the following steps are carried out:

1. Each data item in this set is fed into the root and shuttled to the appropriate leaf; meanwhile, a counter at each node calculates the error rate of items passing through the node, that is, how often the node's class differs from the class of an item in it.

2. In a recursion that moves upward from the leaves to the root, the YES and NO subtrees of a node n_i are deleted and n_i is turned into a new leaf if the sum of the classification error rates achieved in all its descendant leaves is higher than the error in node n_i.

In this way the criterion applied during pruning is independent from the impurity used during tree expansion.

2.9 Summary

Classification of speech signals involves several steps, which are recording, preprocessing, feature extraction, and classification. The classification itself is a mapping of a feature vector or a sequence of feature vectors to a single class. This is called isolated speech recognition. In the case an utterance contains several classes, for example, words, one has to deal with continuous speech recognition which is the mapping of a sequence of feature vectors to a sequence of classes.

Classifiers are based on a distance measure, on discriminant functions or on the computation of class a posteriori probabilities. The latter can easily be estimated if the feature vectors are distributed according to multivariate normal densities. If this assumption does not hold, distribution free classifiers are often used. Neural networks (NNs) are a special case of classifiers which do not impose any restrictions on the features. The input features are propagated through the network along the edges. Each node computes an output activation which is the weighted sum of the activations of the lower level nodes squashed by a non–linear activation function such as the sigmoid function. NNs can theoretically compute any functional mapping. This also means that in classification tasks the feature space can be partioned into arbitrary regions, which correspond to the classes to be classified. In practice NNs have been shown superior to other classifiers for certain classification tasks. The NN weights are trained by back–propagation, a gradient descent method, which minimizes the mean squared error between the activations of the output nodes and a desired output. This optimization converges to a local minimum. It can be shown that if the NN has a sufficient number of weights and if the training actually converged to the global minimum of the error function it computes a posteriori probabilities for the classes.

For continuous speech recognition hidden Markov models (HMMs), a special kind of finite automaton, are widely used. These model a doubly embedded stochastic process: a state sequence generates a sequence of observations. The underlying state sequence is hidden in a sense that each state sequence can produce each observation sequence with a certain probability, but given a particular observation sequence one does not know by which state sequence it has been generated. In the case of continuous HMMs the observations are continuous feature vectors as determined from a speech signal. The state sequences are controlled by state transition probabilities. Observations are generated according to state specific observation densities. HMMs are usually optimized iteratively with the EM algorithm, which maximizes the likelihood for a set of given observation sequences to be generated by an HMM. For each utterance a training HMM is built by concatenating the individual class HMMs. The training algorithm requires the utterances to be annotated manually by the sequence of spoken classes; a (usually time consuming) time alignment of this sequence is not necessary. For the recognition one uses a looped model where the individual class HMMs are put in parallel and their final and initial states are connected to a designated state respectively. A loop back transition allows for the recognition of sequences of arbitrary length.

The recognition task itself is solved by a search in the space defined by the HMM topology. The result of the search is the optimal state sequence which also determines the optimal class sequence. The search can either be performed by the Viterbi or by the A* algorithm. The Viterbi algorithm is a special dynamic programming technique which for each state in the search space chooses in a time synchronous manner the optimal predecessor before the paths are extended. The cost function only takes the costs along the so far expanded paths into account. Viterbi explores the full search space and thus guarantees to find the optimal path.

The A* search explores only part of the search space. This is achieved by not only considering the costs for the expansion of paths but by also taking into account an estimate of the remaining costs for the expansion of a path to the goal. Thus the search is better informed than in the Viterbi search. If the remaining costs are constantly set to zero, A* performs a breadth first search and is therefore equivalent to the Viterbi search. At each expansion step the search path with the minimal costs is expanded. This results in a time asynchronous search. If the remaining costs are monotonous and if they underestimate the real costs then the first path which reaches any goal of the search is the optimal path. Both A* and Viterbi search can be applied to a number of search problems and not only for the search of the optimal state sequence in an HMM.

HMMs only compute the likelihood of an observation sequence given a class sequence. Since one is interested in the a posteriori probability of a class sequence given the observations, one needs, according to the Bayes rule, estimates for the a priori probabilities of class sequences. These can be approximated by the relative frequencies on n-grams which are sub–sequences of classes of length n. A problem in the use of n-grams is the handling of unseen events, that is, n-grams which occur during test but have not been observed in training. Polygrams are a useful approach for the interpolation of higher order n-grams with lower order ones including zerograms. The interpolation weights are automatically estimated on the training data. This allows on the one hand a robust modeling and on the other hand the use of almost arbitrarily large n-grams. Bigrams can be easily incorporated in a Viterbi search. When higher order n-grams are to be used the A* search is the better choice.

NN/HMM hybrid systems combine the ability of NNs to classify arbitrarily distributed feature vectors with the sequence modeling capabilities of HMMs. Two major approaches can be distinguished. The first one uses the NN for feature vector transformation, that is, unrestricted feature vectors are transformed so that they can be modeled by mixtures of normal densities on the HMM side. NN and HMM are jointly trained so as to optimize the likelihood of an observation sequence. This especially means that during the NN weight optimization the derivative of this likelihood rather than the derivative of the squared error is used. Other NN/HMM hybrids rely on the assumption that the NN computes probabilities. These probabilities are used to replace the HMM observation probabilities. Hence, these hybrids can model sequences of unrestricted feature vectors.

For the classification of sequences of unrestricted feature vectors multi–level semantic classification trees (MSCTs) can be used. The sequences can have variable length and are mapped to a single class. The multi–level information can for example be a sequence of words to which other information such as word categories or acoustic features can be attached. The MSCTs are binary decision trees. At each node a question is asked and answered with YES or NO. SCT–type questions contain regular expressions consisting of words and gaps. If a regular expression matches an utterance, further questions can be asked concerning the additional information attached to the words which were matched by the regular expression. An MSCT is trained on sample data, which means the local criterion that determines the subset of rules and the order of their application is related to minimizing the classification error.

The methods described in this chapter build the basis for the approaches developed in Chapters 7 and 8 concerning the use of prosody in ASU. Furthermore, many of them are used in the ASR systems described in the following chapter, and they should allow for a rough understanding of the literature survey about the use of prosody in ASU given in Chapter 4.

Chapter 3

The Baseline ASU Systems

This chapter gives an overview of the ASU systems EVAR and VERBMOBIL, in which prosodic information has been integrated by the author as described in Chapter 8. The first section describes a typical state–of–the–art speech recognizer as it is used in EVAR and VERBMOBIL; we applied a prosodic post–processor to its output. The dialog module of the EVAR system has been extended so that the dialog control is partly based on prosodic information. The VERBMOBIL system uses prosodic information on different linguistic levels. So far, the prosodic information we compute has the highest impact on the syntax module: The parsing of word graphs becomes much more efficient and results in a lower number of readings if prosodic boundary scores are used.

3.1 Speech Recognition

The experiments described in Sections 8.1 and 8.2 are based on the output of a continuous speech recognizer. Those speech recognizers which have state–of–the–art performance only differ in minor parts; the basic methods are the same for all of them. Actually, the experiments have been conducted on the output of three of such recognizers developed by different institutes[1]. Based on the one presented in [Gal96] the principle structure will be sketched in the following (cf. also [Nie94c, Nie95, Kom95a]); as for detailed surveys on ASR cf. [Rab93, ST95a]). The basic underlying methods were outlined in Chapter 2: for hidden Markov models (HMM) cf. Section 2.3, for n-gram (polygram) language

[1] Two of these recognizers are the ones alternatively used in the VERBMOBIL research prototype system, which were developed by Daimler Benz, Ulm (cf. [Cla93b, Cla93a, Kuh96]), and by Universität Karlsruhe (cf. [Suh95, Rog95]); the third one was developed by Universität Erlangen–Nürnberg (cf. [ST94, Kuh95b, Gal96]).

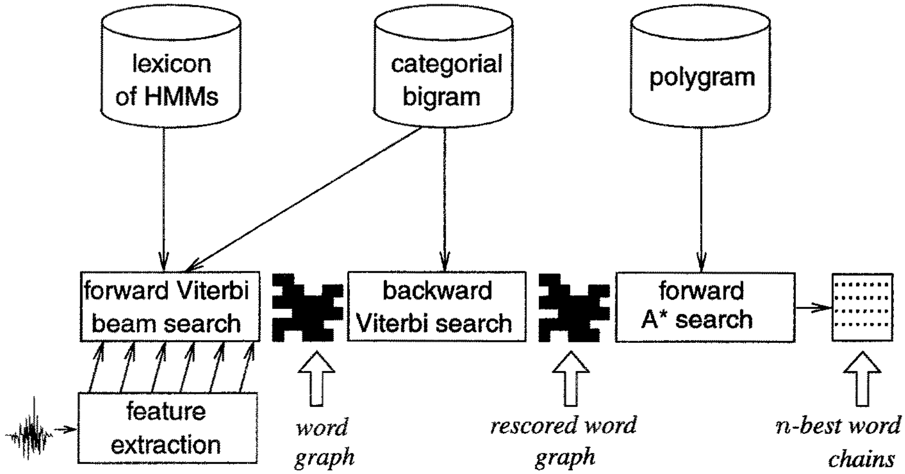


Figure 3.1: Structure of a speech recognizer (after [Gal96]).

models cf. Section 2.4, and for Viterbi and A* search algorithms cf. Section 2.5 and Section 2.6, respectively.

The ultimate goal of statistical speech recognition is to select from all possible word sequences $\boldsymbol{v}$ that sentence hypothesis $\boldsymbol{v}^*$ which maximizes the a posteriori probability $P(\boldsymbol{v}|\boldsymbol{c})$ or, equivalently, the joint probability

$$P(\boldsymbol{v}, \boldsymbol{c}) = P(\boldsymbol{c}|\boldsymbol{v}) \cdot P(\boldsymbol{v}) \tag{3.1}$$

of $\boldsymbol{v}$ and the acoustic input representation $\boldsymbol{c}$. The conditional probability function $P(\boldsymbol{c}|\boldsymbol{v})$ is referred to as the *acoustic* or the *speech model* of the recognizer; estimates of these quantities are usually provided by (semi–) continuous HMMs operating on context–dependent phones as sub–word speech units. The a priori sentence probability $P(\boldsymbol{v})$ is referred to as the *linguistic* or *language model* of the recognizer; $P(\boldsymbol{v})$ is approximated by a polygram model.

The structure of such a system is depicted in Figure 3.1. Due to the limited performance of current technologies, the recognition process is divided into four steps:

1. During *feature detection* [Rie95] the digitized speech signal is partitioned into frames of constant length (usually 10 msec), in which the spectral properties are assumed to be invariant. This in fact does not always hold, especially, for phone transitions and plosives. For each of these frames a vector of heuristic features is computed, consisting of mel cepstrum coefficients, which are a cosine transform of the mel scaled spectrum, and their time

derivatives, cf. [O'S87, pp. 422–423]. The advantages of the cosine transform are a data reduction and that the features become distributed according to a normal density. To keep the memory requirements low the feature extraction is performed quasi–parallel to the synchronous forward search.

2. A *Viterbi beam search* based on word semi–continuous HMMs and on a categorical bigram language model determines those word hypotheses, the HMMs of which best match the sequence of feature vectors given the constraints of the language model. In the search the acoustic and the language model probabilities are combined with a heuristic weight. The Viterbi algorithm is used, because it performs a time synchronous search, and therefore keeps the memory requirements during the computation tractable. The output of this search is a word graph.

3. This word graph is then rescored by a backward Viterbi search. As a result in each node of the word graph the costs of the optimal path from this node to the last node of the word graph are known. These are used as approximations for the remaining costs used in the subsequent A* search.

4. Finally, an A* search is conducted on the word graph combining the probabilities of higher order n-gram language models with the acoustic scores.

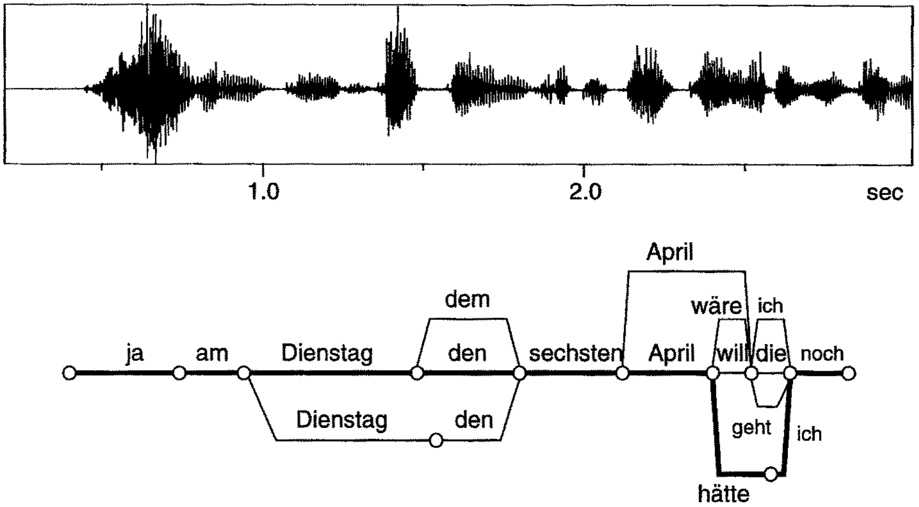
The final step is only performed if the application requires the (n-) best word chain(s) as output. The n-best word chains are computed in order to provide alternative hypotheses, which can compensate for recognition errors in the optimal word chain. The word graph can also be directly passed to the speech understanding module(s), because it allows for a much more efficient linguistic analysis. A definition of word graphs and a comparison with n-best word chains can be found in Section 3.2.

The HMMs constitute the speech model, that is, the observations emitted when passing a transition are the feature vectors described above. To be flexible with respect to the domain and thus to different recognition vocabularies and in order to train robust models with limited training data a set of basic HMMs corresponding to sub–word units is used. These are phones in a certain context, so called *context–dependent phone models* [Sch85, ST92, Kuh92]. This approach is based on the observation that due to coarticulatory effects the acoustic features [Lad82, pp. 52–65] and the durational properties [Man92] of a certain phone depend very much on the phonemic context. Such phone models can depend on a context, which comprises only the adjacent phones or as much as the whole word (or even more than one word if the models are not used for recognition but for acoustic verification and rescoring of n-best word chains). Rare contexts are represented by monophones only.

In a training phase the HMM emission and state transition probabilities are optimized typically on dozens of hours of speech. For the HMM parameter estimation the Baum–Welch algorithm is used. Appropriate context–dependencies for the phone models are determined based on their frequencies in the training corpus; there is a trade–off between the number and the robustness of the models: the larger the contexts the more accurate the models, the less frequent the tokens and the less robust the estimated parameters. To improve the generalization capabilities of such models a training algorithm called A.P.I.S. was developed [ST94, ST95a], which smoothes the parameters of phonetically similar models in a way that the strength of smoothing depends on the frequency of the corresponding context–dependent phones. All the training is conducted in a semi–supervised manner, that is, only the sequences of feature vectors and the orthographic transcription of the utterances are needed; the speech signals do not have to be segmented into words or phones by hand.

Two different n-gram language models are used in the recognizer of Figure 3.1. During the Viterbi search, for complexity reasons only bigram or at most trigram models can be used; the recognizers used in the experiments presented in this book and referenced at the beginning of this section rely on bigrams; as for the use of trigrams cf. [Ney94b, Woo95]. However, the larger the n of the model the better the grammar underlying the language is approximated. Therefore, in the final A* search for the best word chain(s) polygrams are used, in which n-grams are considered where $n \leq 4$.

Basis of the HMM word models is a lexicon, which contains for each word to be recognized one or more alternative pronunciation(s). Additionally, frequent and especially clitic word groups as guten Morgen (*good morning*) or das ist (*that is*) can be contained in the lexicon to allow for a better modeling of their acoustic–phonetics; for the term clitic cf. Section 4.1.2. The lexicon was created manually for the word recognizer described here. Very large lexicons containing some ten thousands of words are usually built with the assistance of automatic text–to–speech devices [Ljo95]. An alternative is the use of context–dependent letter models instead of phone models, which have successfully been used in [ST94]. For each application domain a different lexicon can be used. The specific word HMMs are concatenated from the context–dependent phone HMMs based on the pronunciation defined in the lexicon. Alternative pronunciations which take into account dialectal variations or frequent assimilations or elisions are not considered in any of the word recognizers used in the experiments presented in this book; as for a word recognizer using this cf. [Gau95].



1. ja am Dienstag den sechsten April will die noch
2. ja am Dienstag dem sechsten April will die noch
3. ja am Dienstag den sechsten April die noch
4. ja am Dienstag dem sechsten April die noch
5. ja am Dienstag den sechsten April will ich noch
6. ja am Dienstag dem sechsten April will ich noch
7. ja am Dienstag den sechsten April will geht noch
8. ja am Dienstag dem sechsten April will geht noch

• • •

Figure 3.2: Part of a speech signal from the VERBMOBIL corpus, an automatically computed word graph, and the eight best word chains.

3.2 Interfaces between ASR and Linguistic Analysis

The experiments presented in Section 8.1 and Chapter 8 were performed on the basis of word graphs. These can simply be considered to be a compact representation of n-best word chains. However, when used in the linguistic analysis, word graphs allow for much more efficient processing. Therefore in this section we will first give a definition of word graphs based on [Nöt94b] and then we compare them with n-best word chains with respect to a subsequent analysis. In the following we will refer to Figure 3.2, which shows part of a speech signal from the VERBMOBIL corpus, a corresponding automatically computed word graph, and the eight acoustically best matching word chains included in the word graph. The horizontal straight path through the graph corresponds to the best word chain. The bold edges represent the word hypotheses corresponding to the spoken words, which are

Ja. Am Dienstag, den sechsten April, hätte ich noch (einen Termin frei). (3.2)
Yes. On Tuesday, the sixth of April, I would still (have a date available).

A *word graph* is a cycle–free graph; it consists of labeled edges and nodes. Each of the edges corresponds to exactly one word hypothesis. The graph has a dedicated beginning and a dedicated ending node. All edges are on a path between these two nodes. A word recognizer typically generates many alternative hypothesis for parts of the speech signal where the recognition is unreliable and few or only one hypothesis if the probability of the best hypothesis is by far better than alternatives. We define the r–th word hypothesis in a word graph as a tuple

$$W_r = (v_{h(r)}, l_i, l_j, t_b, t_e, C(v_{h(r)}, t_b, t_e), J) \qquad (3.3)$$

where $v_{h(r)}$ refers to the $h(r)$–th vocabulary entry by its orthographic transcription. Such an entry can be a word or a group of words. The positions on the time axis of the beginning and the end of the word are given by t_b and t_e respectively. The acoustic score of the word is specified by the logarithm of the likelihood computed by the word specific HMM: $C = log(p({}^{t_b}\boldsymbol{c}, \ldots, {}^{t_e}\boldsymbol{c} | \boldsymbol{\pi}_{h(r)}, \boldsymbol{A}_{h(r)}, \boldsymbol{B}_{h(r)}))$. J can optionally contain information as the time alignment of the phones underlying a hypothesis or prosodic scores. The word hypotheses are connected via *logical* graph nodes, where l_i and l_j denote the index of the beginning and of the end node of the edge corresponding to the hypothesis. The order on these indices reflects the order of the logical graph nodes on the time axis. Most of this information has been omitted in the figure, however the n-best word chains depicted at the bottom are determined on the basis of the actual scores $C(v_{h(r)}t_b, t_e)$ computed by the word recognizer.

The graph nodes in the figures are properly time aligned; note, however, that a node in a word graph not necessarily has a unique position on the time axis, but during the forward search several words ending close to each other can be united to one graph node in order to yield small word graphs for efficient subsequent processing [Kuh96]. The position $\tau(l_i)$ of the node l_i can be defined as the mean of all the final frames t_e of the hypotheses ending at l_i. Therefore, in addition to l_i, l_j each word hypothesis is annotated with t_b, t_e. A word graph can especially contain several edges in parallel, that is, between the same pair of nodes, which are labeled with the same word but have different positions on the time axis.

The n-best word chains allow for an easy linguistic analysis, because one can simply start with the best scored word chain and proceed with the subsequent word chains until a word chain can be successfully analyzed. The drawback is that each pair of word chains differs only in a few words, so that the linguistic analysis for the same partial word chains has to be repeated quite often. The effort for a linguistic analysis based on the n-best word chains becomes obvious, when looking at the example in Figure 3.2. The graph contains only 18 word hypotheses but as much as 30 different word chains. The spoken word chain is at the twelfth position. Already the first word chain could be the beginning of a syntactically and semantically correct one:

Ja. Am Dienstag, den sechsten April, will die noch (ein Treffen) (3.4)
Yes. On Tuesday, the sixth of April, she wants another (meeting)

If at all, it might be decided rather late that this word chain is not correct. Then, in the case of n-best word chains, the analysis would continue with the beginning of the next best word chain. When operating on a word graph it can simply consider another alternative in the search space, while reusing the interpretation of a partial word chain. Therefore, it is much more efficient if the linguistic analysis is combined with a search algorithm (usually A*), so that it can operate directly on a word graph. This approach has already been suggested by [Nie86, Nie88] and applied by [Nie92] on the basis of word lattices, which are sets of (alternative) word hypotheses not connected as to form a graph, cf. also Sections 3.3, 3.4 and [Sch94, Han95, Sen95]. How such a search procedure can be combined with a rule based syntactic analysis will be explained in detail in Section 8.3. In this context the prosodic scoring of word graphs is an important aspect of this book; it will be described in Section 8.1.

When working with word graphs one has to be able to judge its quality. In VERBMOBIL the following measurements are used. The *density* of a word graph is the average number of hypotheses per transliteration item. A transliteration item includes the spoken words, non–verbals such as laughter, pauses and noise. The *best matching* word chain is the one contained in a word graph which has the smallest *word accuracy* (RA, cf. Section 2.1). If more than one word chain share the smallest (RA), the best matching word chain is the one with the best acoustic score determined by the word recognizer. The best matching word chain is efficiently determined by a DP search [Pau94]. The *word accuracy* of a word graph is defined as the word accuracy of the best matching word chain. A *correct* word graph is a word graph which has 100% word accuracy. In contrast to the best matching word chain, we define as the *first–best* word chain contained in a word graph the one which has the highest score. The score can be a combination of the scores computed by different knowledge sources, for example, the acoustic and n-gram scores. It is assumed that knowledge about the spoken word chain is not used when searching for the optimal word chain. In the same way the n-best word chains contained in a word graph can be determined.

3.3 Information Retrieval: EVAR

The speech understanding and dialog system EVAR (the acronym stands for the German words for "to recognize", "to understand", "to answer", and "to ask back") is an experimental automatic travel information system in the domain of German *InterCity* train time table inquiries. It has been developed at the Lehrstuhl für Mustererkennung, Universität Erlangen–Nürnberg. This section will give an overview; for a more detailed description cf. [Mas94a]. Figure 3.3 shows the structure of EVAR. The two main components are the linguistic analysis and the acoustic processing.

Acoustic Processing

Continuous German speech is input to the system either via microphone or via telephone line; it is recorded and digitized with a DESKLAB from Gradient directly connected to the work station where the system is running. In the current version, output of the speech recognizer is the best word chain, but word hypotheses graphs can be used as well. The generation and verification of word hypotheses is based on HMMs and a bigram language model, cf. [ST94, Kuh95b]. If the recognizer generates word graphs, larger word groups underlying hypotheses of the linguistic analysis can also be verified in a top–down mode by the speech recognizer, that is, the score v is recomputed using an HMM for the entire word group. This in general results in a better time–alignment. Furthermore, HMMs consisting of context–dependent phone models crossing word boundaries can be used.

The Semantic Network System ERNEST

For the representation of linguistic and domain knowledge the *semantic network* system ERNEST is used, which is a general application–independent framework for pattern analysis [Nie90b, Sag90]. A semantic network consists basically of *concepts, attributes* and *links* between concepts. The concepts constitute the *declarative* knowledge representing objects or abstract terms. They are mainly characterized by attributes, relations (among concepts), temporal or spatial adjacency conditions, and its links to other concepts. Attributes are realized as *procedural* knowledge. Different links between concepts are distinguished: between different *levels of abstraction* the concepts are connected with *concrete (con)* links.

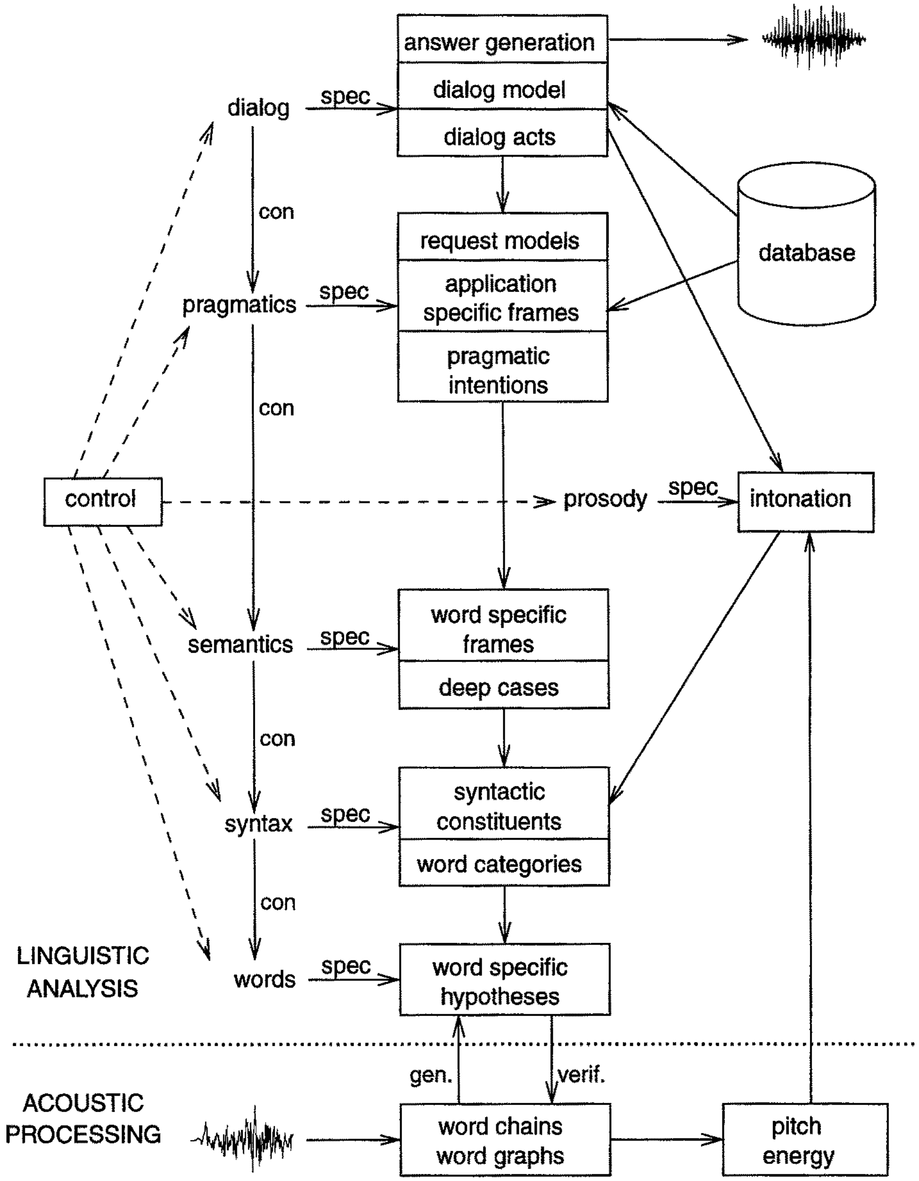


Figure 3.3: The speech understanding and dialog system EVAR.

Within a level of abstraction concepts are linked using *part* and *specialization (spec)* links. A noun group, for example, is more specialized than a concept describing syntactic constituents in a general way, but it *inherits* some properties (attributes and so on) from the more general concept. Parts of a noun group are nouns, proper names, adjectives, determiners, and pronouns. ERNEST includes tools for the creation and compilation of concept definitions. Furthermore, it provides a problem–independent control module operating on the basis of the A* algorithm and problem–dependent judgment vectors. The task of the control is to search for the optimal interpretation of given sensor data. This will be explained below using an example from the ASU domain. A search is necessary to cope with competing hypotheses computed from the sensor data (for example, alternative word hypotheses) and to select among different interpretations, which are the result of non–deterministic knowledge and ambiguous partial interpretations.

The Linguistic Model

All knowledge needed for the speech understanding process and for the dialog is embedded within a single semantic network using the same representation language. Thus it is easy to propagate constraints between all levels to support the understanding process. In fact a high degree of interaction between the modules in terms of an early exchange of restrictions based on partial analyses takes place. This especially allows for the interpretation of an entire dialog not only of single utterances; in other words, in EVAR a complete *discourse model* is realized, which of course is restricted to the application domain. Nevertheless, the knowledge–base is easy to extend and to modify because of its modularization into levels of abstraction, which are described in the following, cf. Figure 3.3 for an overview and Figure 3.4 for a part of the semantic network:

The *word hypotheses* module is the interface between speech recognition and linguistic analysis. Word hypotheses restricted to the linguistic and task–specific expectations (which depend on the actual state of the analysis) are requested from the speech recognizer. This is implemented by a single concept (H_WORDHYP).

In the *syntax* module, word categories, syntactic constituents and special dialog relevant cue phrases are represented [Bri87, Kum92]. The order of the constituents on the sentence level is rather free, because in German this is characteristic for spontaneous speech. In Figure 3.4 a prepositional phrase (represented by the concept SY_PP) is depicted; it has the parts SY_PREP (preposition) and SY_NP (noun phrase). A noun phrase can consist of a proper name (SY_PROPER), a noun (SY_NOUN), a determiner (SY_DET), and one or more adjectives (SY_ADJECTIVE). All of these concepts are linked via part links to the concept SY_NP. Which subsets of these parts can constitute a noun phrase at a time, and in which order they have to occur in the utterance, is specified in the definition of concept SY_NP; for example, a possible noun phrase can consist of a determiner and a noun in this order.

In the *semantic* module verb specific frames with their deep cases accord-

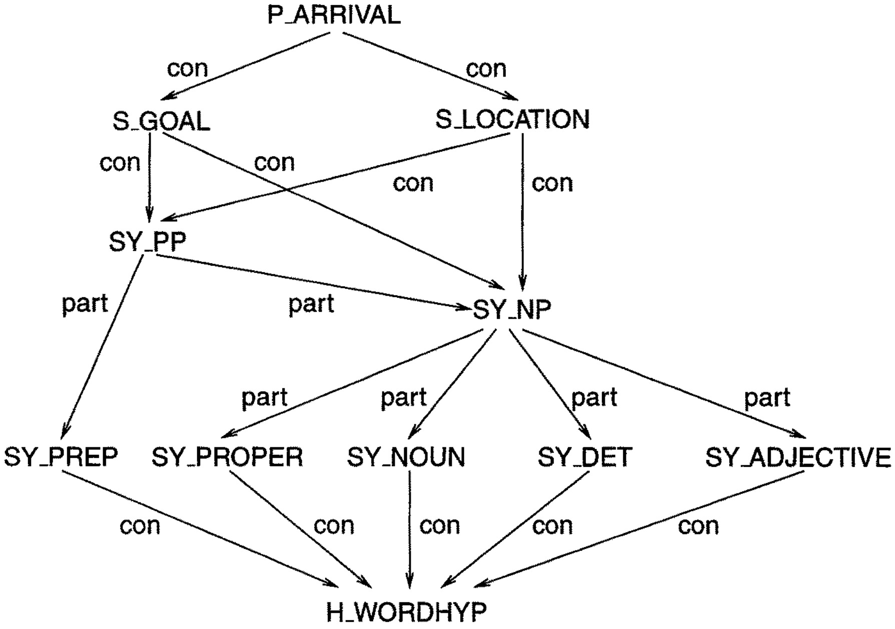


Figure 3.4: Realization of the pragmatic concept P_ARRIVAL.

ing to Fillmore's deep case theory are represented as concepts [Fil68, Ehr90]. This theory has also been extended to nouns; for example, the term *train connection* requires always a goal and optionally a source of the connection. Deep cases are constituents which may fill specific roles of a verb or a noun. For example, one of the deep cases of the verb fahren (*to go*) is the goal of a movement, which can be realized as a prepositional phrase like nach Hamburg (*to Hamburg*) (concept S_GOAL), whereas the prepositional phrase in Hamburg (*in Hamburg*) is interpreted as a location (concept S_LOCATION) and can fill this role for the deep case of the verb ankommen (*to arrive*) realized by the specific concept S_ANKOMMEN, which is not shown in the figure.

The *pragmatic* module represents task–specific knowledge [Ehr90]. For example, the semantic level concept S_LOCATION can be pragmatically interpreted as place of arrival (concept P_ARRIVAL) or as place of departure (P_DEPARTURE). Furthermore, the internal representation of the interpretation is transformed into a form suitable for a database request. This is done by concepts representing possible pragmatic intentions of user utterances and by others specifying the information necessary for a database request.

The *dialog* module models possible sequences of dialog acts [Mas92, Mas93], cf. also Section 8.6.2. With this it is responsible for the overall dialog control. It

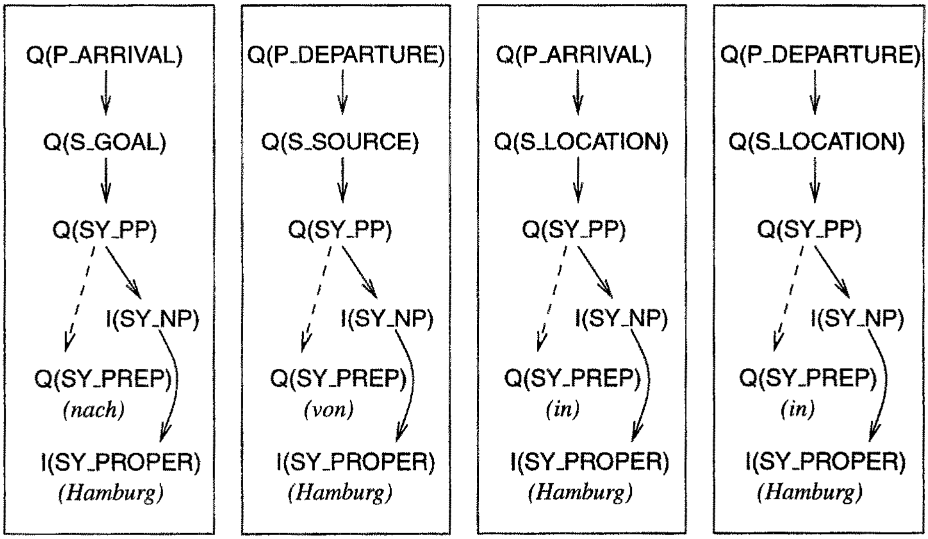


Figure 3.5: Four hypotheses for a partial interpretation of Hamburg (solid lines) and predictions (dashed lines) based on constraint propagation. The lines indicate part or concrete links. The word hypotheses level is omitted.

starts a database request as soon as all necessary information is available. As long as this is not the case a clarifying dialog is conducted. The dialog act specific concepts define how a dialog act can be realized in terms of concepts of the lower linguistic levels. Furthermore, natural language system answers are generated and handed to a text–to–speech facility.

The *prosody* module assists the dialog control by analyzing the intonation of an utterance and thereby providing information about the sentence mood. This is used to determine the dialog act of elliptical time of day expressions, which can be a confirmation, a question or a feedback. Upon this depends the continuation of the dialog. Since the integration of the prosody module into EVAR was one aspect of the research presented in this book, we will explain it in detail in Section 8.6.2.

Analysis and Control

In addition to the provision of the knowledge representation scheme, ERNEST provides problem–independent mechanisms to use this knowledge for the analysis process [Kum92]. It allows for a flexible control of this process, which is alternating between *bottom–up* and *top–down search.* During the analysis alternative partial hypotheses compete. A hypothesis consists of a part of the knowledge base, where each concept A was adapted to the specific input signal (*instances* $I(A)$) or was *modified*, denoted $Q(A)$, so as to take into account restrictions resulting from the actual state of analysis. An instance of a concept A can be built when instances to a predefined set of concepts have been built, to which A is connected via concrete or part links. The A* algorithm in combination with problem–dependent judgment vectors is the basis of this control. In each iteration the best hypothesis is expanded with respect to problem–independent *inference rules*. The goal of the analysis is the creation of an instance for a dedicated concept.

In Figure 3.5 alternative partial (bottom–up) interpretations of an utterance containing for example the word hypothesis Hamburg and subsequent top–down predictions (dashed lines) are depicted. Hamburg can be interpreted, for example, as part of a prepositional phrase (P_PP) on the syntactic level, which in turn can be alternatively semantically interpreted as a goal (S_GOAL) or a source (S_SOURCE) of a movement or it can be a location (S_LOCATION). On the pragmatic level a goal of a movement is interpreted as place of arrival (P_ARRIVAL), and equivalently, a source is a place of departure (P_DEPARTURE). A location can be pragmatically interpreted as both place of departure or place of arrival.

Each of these four different interpretations builds alternative hypotheses in the search space for the optimal interpretation. Based on these partial interpretations and on the knowledge attached to the concepts, constraints can be propagated top–down through the layers of the network to the word hypothesis level. In this example depending on the hypothesis in the search space it can be predicted that the word Hamburg is preceded by a preposition, which can be either nach (*to*), in (*in*), or von (*from*). Which of them is actually predicted depends on the higher level interpretation. In Figure 3.5 the prediction step is indicated by a dashed line. In the subsequent analysis step the search tries to determine a corresponding word hypothesis in the word graph, which was computed by the word recognizer.

Several judgments of a hypothesis are combined in a vector. The most important ones are a score for the reliability of the interpretation, and the acoustic score of the word hypothesis underlying the actual interpretation, which was computed by the speech recognizer. The ultimate goal of the analysis is the creation of an instance of a sequence of dialog–level concepts until all the parameters for a database request are known. Eventually, this results in the creation of an instance of the concept modeling an entire dialog.

For experiments with naive users, it is important to give correct train connections. Otherwise the users often do not take the experiment seriously as has been pointed out in [Hit89]. Therefore EVAR is connected to a *database* containing the official and always up–to–date German train time table (HAFAS, developed by HaCon).

3.4 Speech–to–speech Translation: VERBMOBIL

VERBMOBIL is a joint research project for speech–to–speech translation [Wah93][2]. The application domain for the current (first) phase of the project is business appointment scheduling in face–to–face dialogs between a German and a Japanese person. A prerequisite concerning the potential application of the VERBMOBIL system is that the dialog partners have a basic active and a larger passive knowledge of English. Therefore, it is expected that they will talk to each other in English for most of the time. They only occasionally will switch to their mother tongue and will explicitly demand a translation into English by pressing a special button. Furthermore, as a consequence no German–Japanese or Japanese–German translation is conducted, but the system translates all German and Japanese utterances into English. The task of the VERBMOBIL system is

1. to build up context knowledge by monitoring the dialog, also while the partners speak in English,
2. to translate the turns uttered in German or Japanese into English, and
3. to conduct a clarifying dialog with a speaker if information for correct translation is missing.

The main research effort lies in

- the recognition and understanding of spontaneously spoken German
- the German to English translation, and
- German speech generation and synthesis.

A first demonstrator of the VERBMOBIL system consisting of the most important modules including our prosody module has been set up in February 1994. A fully operational prototype system as it is described in this section has been built in October 1995 and was continuously improved since then. It includes the prosody module, which has been developed in course of the research presented in this book. Only little has been and will be done regarding the Japanese–English translation; thus Japanese ASU or translation has not been incorporated yet in the VERBMOBIL system. Currently, English speech understanding in the system is reduced to the determination of dialog acts so that information about the dialog history is available if a speaker switches to his mother tongue. The English speech synthesis was not developed within VERBMOBIL.

[2] The project started January, 1993; cf. also http://www.dfki.uni-sb.de:80/verbmobil/

The structure of the VERBMOBIL system is shown in Figure 3.6. In contrast to EVAR the analysis of a user turn is performed strictly bottom–up. Therefore, no overall control module is needed; however, the dialog module conducts a few control tasks like the appropriate reaction to not recognizable or not understandable speech input. Two analysis modes are distinguished: A *shallow analysis* is performed by the dialog module if the *deep analysis* by the modules for syntactic analysis, semantic construction, semantic evaluation and dialog fails. Flat analysis refers to a coarse extraction of the most important information.

Figure 3.6 gives an overview of the VERBMOBIL system architecture. In the remainder of this section we will describe the different modules. These are in contrast to EVAR implemented independently from each other and run as distinct processes, which communicate through pre–defined interfaces relying on the *INTARC Communication Environment* [Amt95], which is based on the concept of *parallel virtual machines* [Gei94b]. DFKI, Kaiserslautern, integrated the modules into one system [Amt96]. Note that it is beyond the scope of this book to explain the underlying linguistic theories and formalisms, instead the reader will be asked to consult the referenced literature. We will restrict ourselves to mention the most important problems to be solved in the different modules.

Acoustic Processing

Input to the system is spontaneous speech, which is recorded and digitized with a Desklab device as in the case of EVAR. Currently, German and English speech can be processed. English speech input is the default, the processing of German is only started if a speaker presses a special button. In the case of German speech input, a *word recognizer* as described in Section 3.1 computes a word graph containing the optimal word hypotheses. Alternatively a word recognizer of Universität Karlsruhe (cf. [Suh95, Rog95]) or of Daimler Benz, Ulm (cf. [Cla93b, Cla93a, Kuh96]), can be used. In the English speech processing the dialog module determines the dialog act(s) based on key words. These are either obtained from a word graph computed by the recognizer of Universität Karlsruhe or from a set of key words hypothesized by an HMM *key word spotter* developed by Siemens München.

Computation of Prosodic Information

The research presented in this book resulted especially in the integration of a *prosody* module in the VERBMOBIL system. Input to the module is the word graph and the speech signal; to each of the word hypotheses probabilities for prosodic accent, for prosodic clause boundaries, and for three types of sentence mood are attached, cf. Section 8.1. This information is used by the linguistic modules for

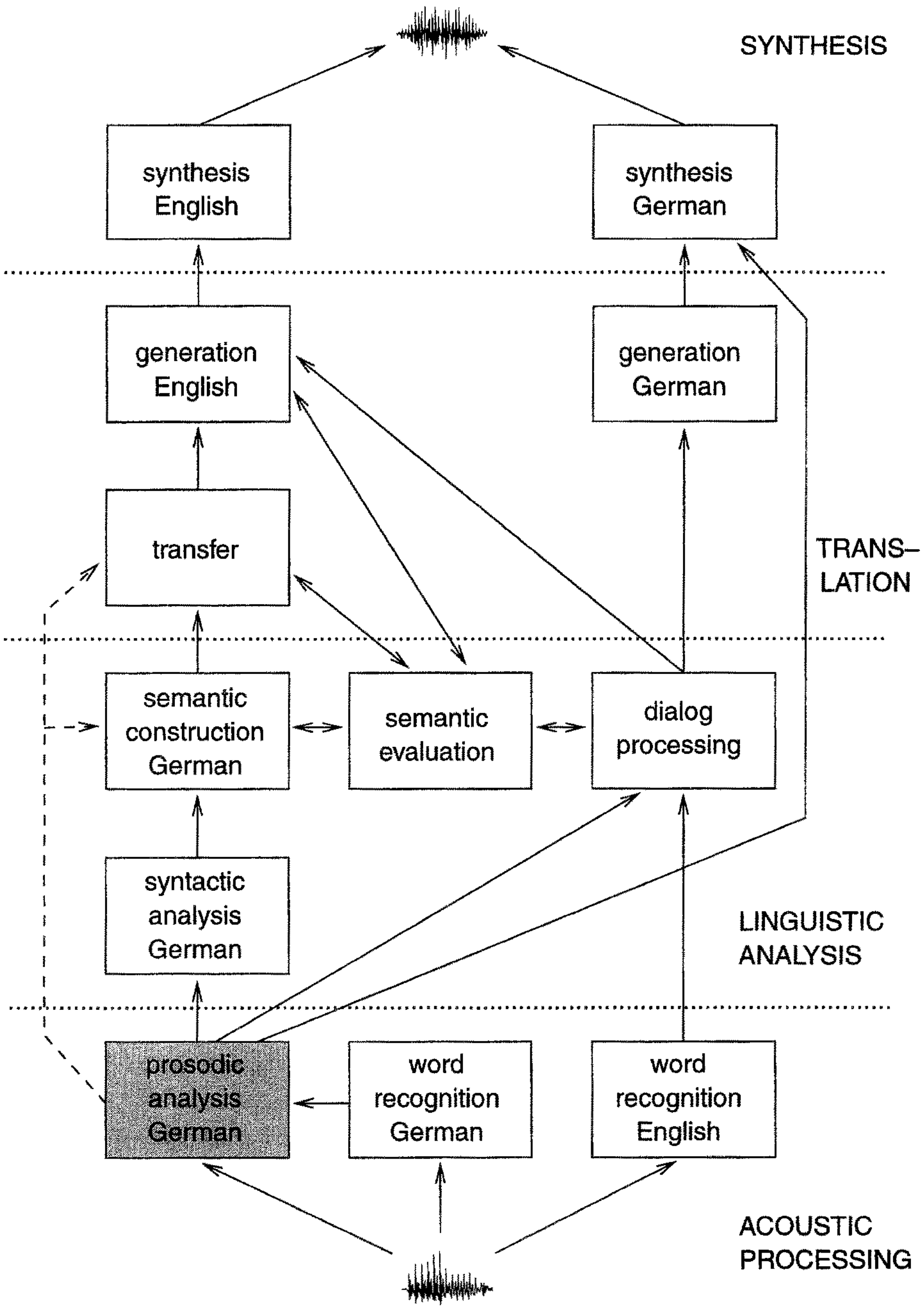


Figure 3.6: Modules and flow of data (solid arcs) and of prosodic information (dashed arcs) in the VERBMOBIL prototype system as of February, 1996.

the purpose of syntactic, semantic and pragmatic disambiguation and for the detection of dialog act boundaries within turns. This use of prosodic information by the VERBMOBIL linguistic modules will be explained in detail in Chapter 8. Furthermore, the F0 contour of each utterance is passed to the synthesis module for adaptation purposes.

A second prosody module has been built in the VERBMOBIL project by the Universität Bonn. Its main purpose is the integration in a separate system, where new architectures for ASU shall be explored. The emphasis is on a high interaction between the modules at early processing stages. Therefore, this prosody module computes its attributes based on the speech signal only; it does not get information from the word recognizer. A comparison of both modules is given in [Hes96].

Syntactic Analysis

The prosodically scored word graph is then passed to the *syntax* module. Two alternative syntax formalisms were developed in VERBMOBIL. In the module developed by Siemens, München, the syntactic analysis is done by means of a trace unification grammar (TUG) [Blo92] and a Tomita parser [Tom86]. The other module was developed by IBM, Heidelberg [Gei94a, Kis95]. It relies on the widely used HPSG formalism [Pol87]. Both formalisms make use of *traces* to model the rather free word order in German; they rely on *unification* as the basic inference mechanism [Nil80, pp. 140–144]. A difference between the two formalisms is that the HPSG approach integrates syntactic and semantic analysis, whereas the TUG only performs syntactic analysis; in this case, the subsequent semantic analysis is conducted by a separate module, cf. below. A comparison of the different formalisms is beyond the scope of this book. The general problems in syntactic and semantic analysis are the same, the differences are in the solutions and in formal representations, which eventually is relevant for the efficiency of the analysis.

The IBM module operates on the n-best word chains. These are extracted from the word graph by a preprocessor which incorporates a stochastic language model and a parts–of–speech tagger in a search procedure [Gei95]. In this preprocessor prosodic boundary information has been integrated to segment the word chains and to predict the position of so called *empty elements* as is described in more detail in Section 8.3.3.

The Siemens module searches for the best scored path in the word graph whose corresponding word chain is grammatically correct [Sch94]. The score in the search integrates acoustic scores of the word hypotheses, bigram scores, and prosodic scores.

The search component is based on the A* algorithm and traverses the word graph from left to right. At each iteration the best partial word chain on the agenda is parsed according to the grammar. If the parse fails, the partial word chain is rejected, otherwise it is extended according to the word graph and the resulting (partial) word chains are scored and inserted in the agenda, which is ordered with respect to the scores. The search stops if a word chain corresponding to a path from the first to the last node in the word graph has been found. This search is described in more detail in Section 8.3, where the use of prosodic boundary information for structural disambiguation is investigated.

On an abstract level both grammar formalisms can be viewed as being context–free grammars whose rules are augmented with *typed feature structures*. These are *attribute–value* pairs; the values can be complex structures or functions. These are for example used for restrictions on case and gender as well as for the description of discontinuous dependencies. Note that if one assumes limited depth in the embedding of sentences, a context–free grammar is sufficient for the description of natural language; however, the introduction of attributes can reduce the complexity of the grammar considerably [ST95a, p. 203]. Since the word order is rather free in German, discontinuities are a severe problem. Consider the following two examples, which have the same English translation (the word–by–word translation is given in parenthesis):

Machen wir noch einen Termin aus. (3.5)
Vereinbaren wir noch einen Termin.
(Fix we another a date.)
Let's fix another date.

There are two discontinuities: The prefix of the verb ausmachen (*to fix*) is moved to the end of the sentence. Furthermore, in both cases the verb phrase (einen Termin ausmachen and einen Termin vereinbaren) is split.

Output of the syntax module is one or more alternative parses of the optimal word chain which has been determined during the search. Examples for syntactic ambiguities, which are the reason for alternative parses, will be given in Section 4.2.3. A parse is a structural representation of a turn on the basis of the syntax rules contained in the grammar.

First of all a turn is segmented into clauses or clause–like units such as free phrases (for a definition of the terms *turn* and *free phrases* cf. Section 5.2.1, page 156). For example the following turn consists of three clauses, where the

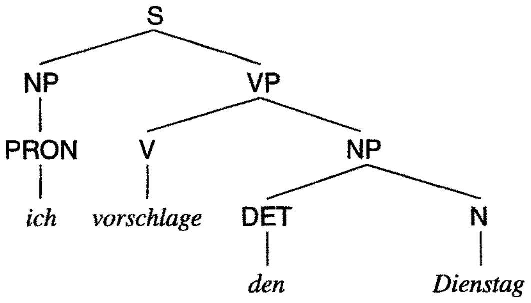


Figure 3.7: A Parse tree of the sentence Ich schlage den Dienstag vor (*I propose the Tuesday*).

first one is elliptic:

Montag? Das paßt echt schlecht bei mir. Ich schlage den Dienstag vor. (3.6)
(Monday? That suits really badly for me. I propose the Tuesday –.)
Monday? That does not suit me at all. I propose Tuesday.

In general clause boundaries within turns are highly ambiguous, especially for spontaneous speech. Prosodic information can contribute a great deal to their determination. This will be discussed in detail in Sections 4.2.3, 5.2.5 and 8.3. The individual clauses are further hierarchically structured into constituents, which can be represented by a parse tree, whose nodes correspond to constituents and whose leaves correspond to words. An example can be found in Figure 3.7. This parse tree does not correspond to the surface structure of the sentence; the separation of the prefix vor of the verb vorschlage is not represented in the tree, which is more appropriate for the subsequent analysis. Attribute–value pairs are not given in the figure; a full representation of the syntactic structure as it serves as input for the semantic analysis in VERBMOBIL can be found in [Bos94c, Mas94b]. As for a survey on grammar formalisms for natural language cf. [Cho81, Shi86].

Semantic Construction

On the basis of the parse tree, the clauses in a turn are then interpreted by the *semantic construction* module. The module which processes the output of the Siemens syntax module is developed by the Universität des Saarlandes, Saarbrücken [Bos94b, Bos94a, Bos94c, Bos96d]. We will describe the underlying formalism in the following, because we come back to it in Section 8.4. Note that the Siemens syntax and the semantic construction have been integrated into one software system, however, the syntactic and semantic processing is still sequential: On the syntax level no semantic restrictions are used. The IBM HPSG semantic formalism, which is integrated with the syntactic analysis, will not be described in this book, but we refer to [Kis95].

In the context of the semantic construction *interpretation* means the mapping of the parse tree onto so called *discourse representation structures* (DRS), which were introduced by Kamp; for an overview cf. [Kam93]. This is an approach for the description of the structure of meaning based upon formal logics. A DRS is a pair consisting of a set of discourse markers and a set of conditions. Conditions are specified in terms of predicates. Let us consider the following sentence:

der Montag geht bei mir (3.7)
(*the Monday suits for me)*
that Monday suits me

This translation corresponds to the default reading, in a few contexts der Montag could also mean "next Monday" and bei mir could refer to "at my place". The result of the semantic construction of this sentence is the following simplified DRS (after [Bos95]):

$[\; e\; t\; i \mid montag(t)\; gehen(e)\; theme(e,t)\; bei(e,i)\;] < i, speaker >$

The discourse markers are the variables e, t and i. The DRS conditions are imposed on these variables; they take the form of logical predicates and have the following meaning:

- The condition $theme(e, t)$ states that variable t is the theme of variable e.
- $montag(t)$ restricts t to be a point in time located on a Monday.
- $gehen(e)$ constrains e to be an event of the semantic type "gehen", which means, roughly spoken, e is classified as being an event that suits.
- $bei(e, i)$ links the event e to the variable i; that means together with $gehen(e)$ and montag(t) it is the speaker who is pleased with this event at that specific date.

The variable i is deictic referring to an individual, in this case the speaker, who is specified in addition to the DRS by the so called *anchor* denoted with $< . >$. The variables e and t are unspecified in this anaphoric utterance referring to *antecedents*, which are expressions used in preceding turns; specifically, e refers to a *type* of an event, for example, a meeting to be scheduled, and t refers to a *specific* date or a set of alternative dates, which were previously mentioned.

The construction of the DRS starts with the leaves of the parse tree, which are replaced by DRSs representing single words. These are taken from an lexicon created offline. The analysis proceeds by moving up in the tree where at each node the DRS of the daughter nodes are combined. The DRSs of different clauses of a turn and of the turns of a dialog are also combined yielding a single DRS representing the entire dialog. This is especially important for the resolution of anaphora or deictic terms. In the original theory DRS construction rules have been based on first order logics [Kam93]; the most important mechanism for context–independent analysis is *unification* [Nil80, pp. 140–144]; additional rules have been provided for context–dependent analysis. In the context of VERBMOBIL this formalism has been extended by applying the well known λ–calculus to DRSs (*λ–DRS*). This results in a *functional and compositional semantics* where all analysis is done within a uniform theory, and which allows the handling of quantifiers and identifications over variables of arbitrary type [Bos94b].

According to [Hei95] special problems of the semantic construction within VERBMOBIL are:

- the interpretation of ellipsis, for example, ich auch (*me too*), and of isolated phrases, for example, richtig (*right*),
- the representation of propositional attitudes, for example, ich denke (*I believe*),
- the determination of focus, and
- the disambiguation of the scope of quantifiers and modal particles

For the determination of the focus of sentences and of the scope of particles prosodic accent information can be of great help; this will be further discussed in Section 8.4.

Not all ambiguities can be resolved by semantic means. As a consequence a turn might correspond to more than one DRS. The DRSs are represented with underspecified structures on the semantic level, so called *labeled underspecified DRS* (LUD) [Bos96b]. This is a formalism for representing a number of DRSs within a single structure, which is to be disambiguated by the module for semantic evaluation if necessary, cf. below. LUDs have been embedded into VERBMOBIL *interface terms* (VITs), which additionally allow for the representation of attributes not used by semantic construction as prosodic information, which has just to be passed to the higher level modules [Bos96c]. These VITs serve as interface to the modules for transfer and semantic evaluation.

Transfer

The *transfer* module is the core of the translation within VERBMOBIL. It is developed by IAI Saarbrücken, Universität Stuttgart, Universität Tübingen and CSLI Stanford [Car94, Cop95, Dor94, But95, Ebe96].

The transfer in VERBMOBIL operates on the semantic level, that is, it maps a DRS representing the source language (German) into a DRS corresponding to the goal language (English). Such a mapping is necessary, because the predicates contained in the DRSs differ in number, names and structure. Without going into details, this should become clear when looking at the following examples, cf. [Cop95]. In

da habe ich immer schon Feierabend (3.8)
(then have I always already time–after–work)
I have usually finished work by then

the noun Feierabend determines a time interval after work has been finished. It can only be indirectly translated into English. In the sentence

das paßt schlecht bei mir (3.9)
(that fits badly with me)
that does not suit me well

the verb paßt together with the preposition bei has to be translated as suit. Another problem are German compound words. For example,

drei Terminvorschläge (3.10)

could be translated as

three suggestions for a date (3.11)

but in most contexts

suggestions for three dates (3.12)

would be better, which results in a different scope of the quantifier.

The transfer is realized as a rule–based approach relying on unification as the basic inference mechanism. Many of the underspecified structures contained in the DRS built by the semantic construction can be left unspecified in the goal DRS. For example, in

Donnerstag den achten oder Donnerstag den fünfzehnten Juli (3.13)
Thursday the eighth or Thursday the fifteenth of July

the scope of Juli is unclear; in most situations it would probably include both Thursdays (*wide scope*), but den achten could be anaphoric referring to another month. This will be represented by an underspecified DRS in the source language and can be left underspecified in the DRS of the target language.

However, some ambiguities have to be resolved; in this case transfer invokes the semantic evaluation. For example, the pronoun er should be translated as him if it refers to a person, whereas it is appropriate if it refers to the word Termin (*appointment*).

Transfer also involves a proper translation of modal and discourse particles where prosody plays an important role. With respect to this, VIT as a data structures has been developed, which allows to pass the prosodic information contained in a word graph through the syntactic and semantic modules to the transfer module. Based on a VERBMOBIL corpus analysis it has been shown in [Rip96c] how prosodic boundary and accent information can be used for the interpretation of such particles. The implementation has been finished shortly before the final version of the VERBMOBIL prototype system has been completed, cf. Section 8.5.

Semantic Evaluation

The task of the module *semantic evaluation* is the pragmatic interpretation of semantic structures using a domain model and the discourse context; it is developed by Technische Universität Berlin [Qua94].

The main goal is the resolution of ambiguities. This is done on demand of the transfer or the generation module in the cases where a DRS contains underspecifications, which have to be resolved in order to achieve a proper translation. For example, the utterance

Geht es bei Ihnen? (3.14)
(*Goes it at you?*)

should be translated as Does it suit you? or as How about your place? depending on the current topic (meeting time versus place).

The approach is a knowledge–based inference engine using so called *description logics*, which are considered as a formal elaboration of representation methods such as semantic networks.

In addition, the task of this module is the creation of context knowledge to be stored in the dialog history. This includes the interpretation of time expressions in a way that they are represented relatively to the current date; for example, the term eine Woche später (*one week later*) will be represented as "current date + 1 week". Furthermore, the turns are segmented and classified into dialog acts.

The pragmatic interpretation of modal particles to retrieve information about the speakers attitude (positive, negative, indifferent) and about the structure of turns (for example repair markers) plays an important role [Sch95b]. Such particles are for example ja (*yes*) and leider (*unfortunately*). Repairs are especially important, because the reparandum should not be translated. Let us consider the following turn from the VERBMOBIL corpus:

sagen–wir–mal auf den ja nee muß ausfallen (3.15)
(let's–say on the yes no should–be canceled)
no it should be canceled

A proper translation would only consider nee muß ausfallen and translate it to *no it should be canceled.* One could even think of contexts where only muß ausfallen has to be translated.

For interpretations with respect to the discourse context the semantic evaluation module relies on the dialog memory maintained by the dialog module. So, it inserts and extracts information into/from the dialog memory.

Natural Language Generation

The module *generation English* produces an English word string from the transferred DRS representation of semantics on the basis of a grammar of English syntax, which is encoded as a so called *tree adjoining grammar* [Fin95]. Before this generation takes place the rough structure of the output turn is planned according to a rule based system, and the words to be put in the utterances are chosen according to some constraints as the appropriate level of politeness.

German generation is necessary for a clarification dialog with the German speaker, and for the output of advice to the user like "Please, speak louder".

Synthesis

Finally the *speech synthesis* transforms the generated word sequence into a speech signal which is output over the speaker [Por94]. The synthetic speech should sound as much as possible like the voice of the speaker whose utterance has been translated. In general this means an adaptation of the output speech to the voice quality and to the pitch properties of the speaker. This is especially important with respect to multi–speaker situations envisioned for the second phase of the VERBMOBIL project. Since no research effort has been spent on English speech synthesis in the first phase of VERBMOBIL, this topic was instead investigated for the German speech synthesis. One of the results is that the output speech of the VERBMOBIL prototype is adapted to the speaker's F0 base value which is approximated by the median of the F0 computed by the prosody module over the entire input speech signal.

The Dialog Module

The *dialog* module [Ale95, Mas95c, Mas95a, Rei95] mainly has to monitor the dialog between the two persons by keeping track of the dialog history. In the case of German input the information to be put into the dialog memory is generated by the semantic evaluation module. If the dialog partners talk in English, input to the dialog module is key words detected by the English speech recognizer or, alternatively, obtained from a word graph generated by the recognizer; these are used for the determination of the dialog act(s) contained in the English spoken turns. If a turn contains more than one dialog act their boundaries have to be detected. The segmentation as well as the dialog act classification itself can be supported by prosodic information; for preliminary experiments cf. Section 8.6.

Based on the dialog history and the current dialog act, speech recognition and linguistic analysis can be provided with constraints concerning vocabulary, stochastic language model, or grammar rules, and with contextual information; note however that the speech recognizers in the March 1996 prototype still do not use this information. The contextual information is essential for semantic evaluation.

Furthermore, the dialog module performs a basic control of the system functions: In cases where the processing of a turn results in inconsistencies or fails completely, the dialog module has to take the initiative and to conduct a clarification dialog with the user. For example, inconsistent date expressions like "April, 31st" are detected by the system, or a user is advised to speak louder if necessary.

An important task of the dialog module is the so called *shallow analysis*. It extracts the main information, that is, date, time, and dialog act, out of the best word chain extracted from the word graph. In the case of English speech input the shallow analysis is currently the only interpretation mode available; furthermore, it is restricted to the determination of dialog acts to update the dialog memory. For German input usually the *deep analysis* is conducted. If it fails the most important information extracted by the shallow analysis is translated by the system. In turns consisting of several dialog acts, the boundaries between them have to be determined prior to the dialog act classification. The use of prosodic and textual information for this task has been investigated in the research presented in this book, cf. Section 8.6. We cannot give any further information on the shallow analysis because of a patent pending.

Lexicon

The *lexicon module* is a database containing phonological, morphological, syntactic and semantic information for each of the words out of the application specific vocabulary. This includes most of the knowledge necessary for the unification based parsing or semantic analysis, for example DRSs for single words. Since the lexicon knowledge is pre–compiled prior to the analysis it is not directly involved in the prototype system. Full forms are especially derived from the morphological representation of the words. This module is developed by Universität Berlin and Universität Bielefeld [Dud94]. The final version of the VERBMOBIL lexicon consists of 2,461 full forms.

3.5 Summary

In the research presented in this book a prosody module has been added to the two ASU systems EVAR and VERBMOBIL. In both of them there is a clear distinction between acoustic processing and linguistic analysis.

The two systems contain state–of–the–art HMM word recognizers. The principal techniques are the same. The main components are the acoustic model, realized by HMMs, and the n-gram language model. The n-gram approximates the probabilities of word sequences. The HMMs compute the probabilities of sequences of cepstral feature vectors extracted from the speech signal. Within a Viterbi search the optimal word hypotheses are determined using a bigram language model. To cope with recognition errors alternative hypotheses are computed at each time frame and are arranged in a word graph. The word graph can be processed by an A* search incorporating further linguistic knowledge. This can be a higher order n-gram language model, if the task is "simply" to compute the optimal word chain.

If the task is to retrieve the user's intention, the search can use structural knowledge in form of a semantic network or a formal grammar. This is done within the linguistic modules of EVAR and VERBMOBIL, respectively. In both systems the interface between acoustic and linguistic processing are word graphs. At the first view, word graphs seem to be just a compact encoding of the widely used n-best word chains. However, when used in linguistic analysis they allow for a much more efficient search for the best interpretation.

EVAR is a dialog system for train time table inquiries. The linguistic processing is based on the semantic network system ERNEST. All the linguistic modules for syntactic, semantic, pragmatic and dialog processing are implemented within the same framework thereby allowing for a high interaction between the modules at early processing stages. This implies a search alternating between top–down and bottom–up. The control uses the A* algorithm and is based on problem–independent inference rules, which govern the expansion of partial interpretation hypotheses, and a problem–dependent judgment vector, which ranks competing hypotheses. The judgments include acoustic and linguistic scores.

VERBMOBIL is a system for speech–to–speech translation in the domain of appointment scheduling. Currently the dialog partners are a German and a Japanese talking for most of the time in English. A German–English translation is done on demand. Japanese–English translation has not been integrated yet. The linguistic processing is strictly bottom–up. The syntax searches for the optimal word chain in the word graph which meets grammatical constraints. The grammatical structure of a turn is basis for the semantic interpretation. The meaning is encoded in discourse representation structures, which basically are a hierarchy of predicates. The translation transforms these structures in a form corresponding to the English goal language. Based on this and a formal grammar English turns are generated.

Fully operational prototype systems have been developed in the case of EVAR and VERBMOBIL. One aspect investigated in this book is the addition of a prosody module to each of these systems. In EVAR the intonation of a user utterance is used to classify the dialog act of elliptic time of day expressions. In VERBMOBIL prosodic information is used to determine boundaries between clause–like segments and between dialog acts within turns. The boundary scores, furthermore, speed up the search of the parser for the optimal word chain contained in the word graph. The detection of accented words helps to determine the focus of utterances and to disambiguate the interpretation of particles. Before we go into this in more detail, the next chapter will define the necessary prosodic terms and will explain their function in human–human communication. Furthermore, a literature survey shows what has been done in the past regarding the use of prosody in ASU.

Chapter 4

Prosody

This chapter first gives basic definitions of acoustic–prosodic speech phenomena in Section 4.1. Then in Section 4.2 the importance of these phenomena in human–human communication is summarized. These two sections are based on [Leh70, Koh77, Vai88, Nöt91a]. Finally, Section 4.3 gives a literature survey on the usage of prosody in ASU systems.

4.1 Definition of Basic Terms

The term *prosody* comprises speech attributes which are not bound to phone segments [Buß90]. We distinguish between *basic* and *compound* prosodic attributes. Basic prosodic attributes are *loudness, pitch, voice quality, duration, speaking rate* and *pause*. Variations of these over time constitute the compound prosodic attributes, which are *intonation, accentuation, prosodic phrases, rhythm,* and *hesitation*. These attributes are bound to units which contain more than one phone; such units are syllables, words, phrases, sentences, dialog acts or even entire turns. So called *micro–prosodic* attributes are relevant for the recognition of phones, cf. [Dum94b], but they do not play a role in this book, because they do not contribute to the prosodic attributes carrying linguistic information [tH90, p. 48]. Most of the prosodic attributes refer to perceived phenomena that means there is no unique correspondence between these attributes and acoustic features measurable in the speech signal.

4.1.1 Basic Prosodic Attributes

Despite the non–unique correspondence, each of the basic attributes named above has an *acoustic correlate*:

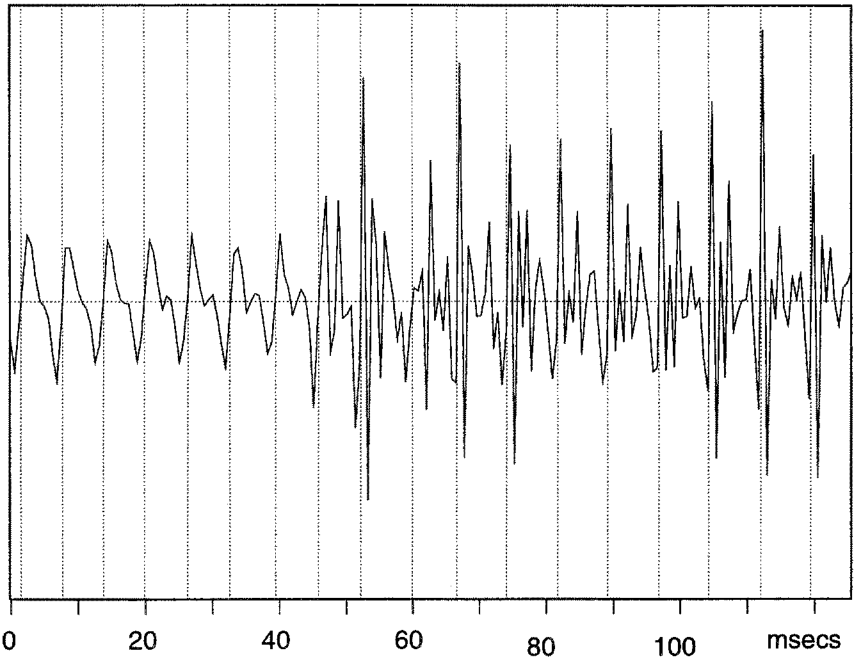


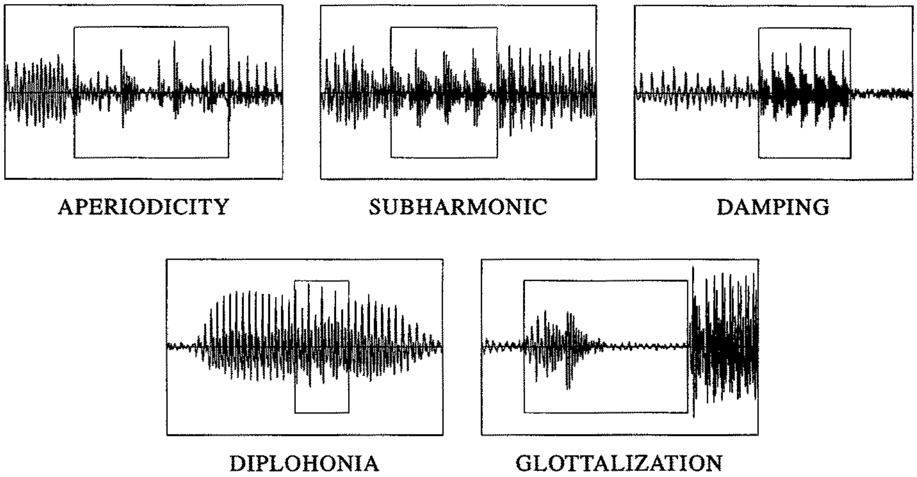
Figure 4.1: Speech signal with fundamental period markers.

Loudness

The correlate of loudness is signal *energy*; the mapping between them depends on the sensitivity of the human auditory system to different frequencies [Zwi67]. Furthermore, the auditory system obviously does some kind of energy normalization with respect to phone intrinsic values [Leh59, p. 429], cf. also [Bec86, pp. 141–144]. It has not been investigated yet, what kind of normalization exactly is performed during speech perception. The phone intrinsic differences result from the specific positions of the articulators, which are tongue, lips, pharynx, uvula, and velum; for more details cf. [Lad82, Chap. 1], [O'S87, Chap. 3]).

Pitch

Fundamental frequency (henceforth F0) is the correlate of pitch. F0 is the reciprocal of the fundamental period, which is defined as the time between two successive glottis closures [Hes83, p. 10 and Sec. 4.4]. The periodic closing and opening of the glottis produces the voiced excitation signal, which is rich in harmonic frequencies. During unvoiced excitation the glottis remains open constantly. The excitation signal is filtered in the vocal tract (mouth, nose, and pharynx) such that



APERIODICITY SUBHARMONIC DAMPING

DIPLOHONIA GLOTTALIZATION

Figure 4.2: Prototypical laryngealizations (after [Bat93a]).

the phone specific frequencies (the formants) are enhanced; the phone characteristics are caused by the position of the articulators. Figure 4.1 shows 125 msec of a speech signal from a male speaker; the signal corresponds to the phones /ma/[1]. Boundaries between the periods are indicated by vertical bars, the length of which varies slightly around the average of 6.6 msec (152 Hz).

In the case of laryngealizations, cf. the next paragraph, an often unusually low F0 does especially not correspond to the perceived pitch, which itself is about equivalent to an extrapolation of the pitch contour around the laryngealized segment [Hub88, Hed90, Bat93a], cf. also [Dal90]. Phone intrinsic parameters and coarticulation also influence F0; for an overview cf. [Nöt91a, pp. 35–36]. These only cause small, micro–prosodic variations ([Leh70, p. 68] reports a range of about 20 Hz), which do not alter the pitch perception and therefore do not contribute to the functional roles of the prosodic attributes as they are defined in Section 4.2. Furthermore, the excitation signal itself is not truly periodic, but it shows small variations in period duration (*jitter*) and in period amplitude (*shimmer*). These variations are not directly perceptible [O'S87, p. 50], but they contribute to the naturalness of speech for example in synthesis.

Voice Quality

The term voice quality subsumes attributes, which concern the overall phone independent spectral structure, for example, jitter, shimmer, or the relative en-

[1] In this book phones are given in SAMPA notation, cf. [Fou89, pp. 141–159] and Appendix A.3.

ergy of the higher harmonics with respect to F0, cf. [O'S87, Sec. 3.2] and [Kla90]. These cause phenomena like timbre, or hoarseness [Leh70, pp. 120–122]; most of these are not well understood so far, but cf. [Lav80] for an overview. The phenomenon which has been mostly investigated so far is *laryngealization* [Leh70, Hub88, Hub92, Bat93a]. These are portions of voiced speech with irregular excitation, which causes F0 to be aperiodic and often to be much lower than the average in the utterance. Other characteristics can be low energy and damping. Prototypical examples for laryngealizations are depicted in Figure 4.2; in each case, the inner frame marks the laryngealized portion of the speech signal. Another voice quality phenomenon is that the first and second formant of unaccented vowels are centralized, that is, they are more close to the ones of the *central vowel* (Schwa, /@/) than is the case for their accented counterparts, cf. [Leh70, p. 139] and [Nöt91a, p. 45]. The term *central vowel* refers to a vowel, whose first and second formant frequencies are about at the center of the frequency range covered by all the vowels.

Duration and Speaking Rate

The duration of single phones or syllable segments as well as the speaking rate are prosodic attributes. An acoustic correlate of the speaking rate can be defined as the reciprocal of the average of the phone durations within a certain time interval. The phone durations are measured in milliseconds. However, each phone has its own intrinsic mean and standard deviation of the duration [Bec86, pp. 141–144]. It is not clear up to now how duration and speaking rate are perceived [O'S95], but in any case an objective measurement of speaking rate can be obtained by normalizing the durations with the phone intrinsic values [Wig92b, Sec. 3.2]. [Cam90, Cry90] moreover found that a context dependent duration model is even more perceptually adequate, where the neighboring phones and the position within the word are taken into account. With respect to this topic cf. also the description of Figure 4.3 on page 101.

Pause

There are two types of pauses: *unfilled* and *filled pauses*. An unfilled pause is simply silence or it may contain breathing or background noises, whereas a filled pause is a relatively long speech segment of rather uniform spectral characteristics consisting of a sort of a Schwa (eh, /@/) sometimes followed by an /m/ (ehm) [O'S93]. The F0 is flat or slightly falling and it is at a comparably low level. Filled pauses are often enclosed by silence. For the perception of silence in the first place it seems to be important if a segment of silence of a minimal length is present; but also the duration of the silence contributes to the perception of certain prosodic attributes, cf. page 118.

4.1.2 Compound Prosodic Attributes

Before we describe the compound prosodic attributes we want to make a few remarks: in the phonetic literature, these attributes are always defined in qualitative terms rather than in quantitative terms; for example, a typical definition of prosodic accent is "accented syllables are perceived as being more prominent than others". If quantitative measurements are given in addition, they usually are determined on a small set of data spoken by one or few speakers, usually consisting of artificial words or minimal pairs. Though such definitions are rather unsatisfactory, in the context of this book it is sufficient to know what qualitative changes in the prosodic attributes can express a linguistic phenomenon. The exact quantitative representation will be automatically determined during the training of the statistical models. So, in the following we just describe which phenomena exist, and by which basic attributes they can be represented.

It should be stressed that the compound attributes as described below can only be determined in a larger context, since they are the result of variations of the basic attributes relative to the context; for example, by definition an isolated syllable cannot be accented. Also in the case of an emphatic what spoken in isolation the accentuation is determined by an implicit context, that is the way the person normally speaks. Note that usually these attributes are defined by perceptive criteria, which is in general the case for accentuation. However, perception is influenced by linguistic knowledge, that is, listeners sometimes believe to perceive prosodic attributes, which have no correspondence in acoustic correlates and vice versa, cf. [Nöt91a, p. 33] and [Hei69, p. 15]. We will describe these attributes with the help of figures showing examples taken from the VERBMOBIL spontaneous speech corpora.

Description of the Figures

Each of the compound prosodic attributes results from the variation of one or more basic attributes in time. Below, we will describe these attributes with the help of the examples in Figures 4.3 through 4.5, which present examples for all of the prosodic attributes being relevant for this book. These figures show from top to bottom: the speech signal, the prosodic labels, the spoken words, the phonemes, the F0 contour, the normalized duration of the syllable nuclei, and the energy contour. The speech signals depicted in the figures are parts of speech signals from turns

prosodic attribute	label	explanation
intonation	F	falling pitch
	R	rising pitch
	CR	continuation–rise
prosodic phrase boundary	B2	prosodic constituent boundary
	B3	prosodic clause boundary
	B9	hesitation lengthening
prosodic accent	PA	primary (phrase) accent
	EA	emphatic (strong) accent
	NA	secondary (weak) accent

Table 4.1: Labels of compound prosodic attributes used in the figures of this section.

out of the VERBMOBIL spontaneous speech database. The time axis reflects the absolute position in the original speech signal. Since such examples are difficult to understand without listening to the speech signals, we made them available under http://www5.informatik.uni–erlangen.de/Private/kp/diss–figs/diss–figs.html .

We do not want to explain all the prosodic attributes found in the figures, but we at least labeled them for completeness[2]. This is especially important when taking the contextual influence of the attributes into account. They mark the compound prosodic attributes perceptible and visible in the plots of the basic prosodic attributes. At least one example of each of the labels will be discussed in this section. Table 4.1 gives an overview of all the labels; they will be explained later in this section. The labels are in accordance with the labels defined in [Rey94] and in Section 5.2.2. However, in this section we do not want to consider accents of different strength.

The position of word boundaries on the time axis are indicated by the long vertical bars; the small bars introduced in the phoneme sequence mark syllable boundaries. The phonemes correspond to the canonical pronunciation of the spoken words, which is also used during the experiments presented in Chapter 7 and 8. It has the drawback that pronunciation variabilities due to dialects or spontaneous speech are not taken into account. For example, perceptive evaluation showed that the phone /@/ has not been uttered at all in the last syllable (/b@n/) in Figure 4.4; this phenomenon is called *elision* and is quite frequent in the case of a Schwa (/@/) followed by a nasal [Koh95]. The time–alignment of the spoken words and phonemes was computed automatically with an HMM word recognizer as it is

[2] The labels have kindly been created by M. Reyelt and A. Batliner.

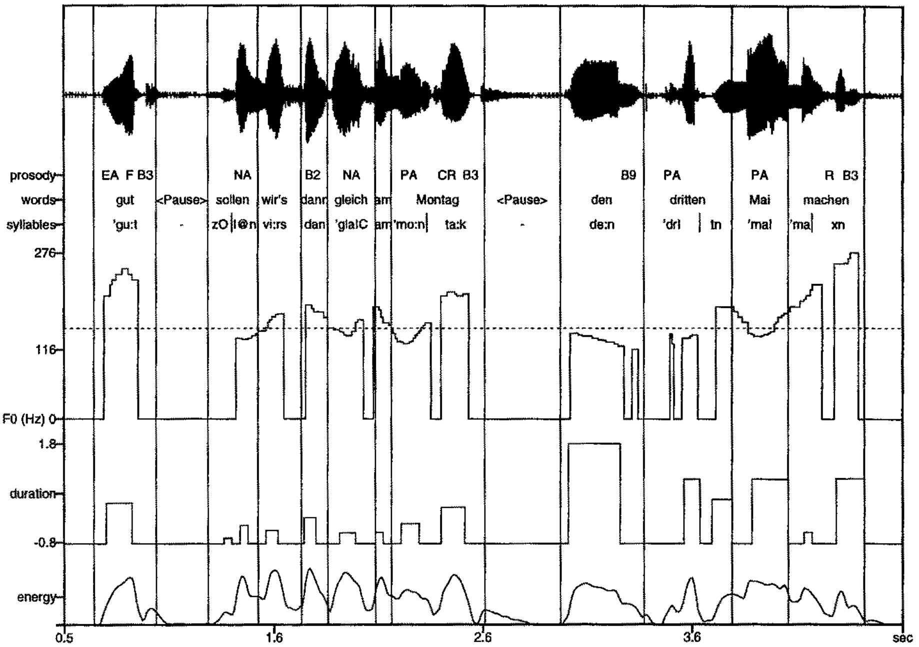


Figure 4.3: Speech signal and acoustic correlates of basic prosodic attributes for the utterance Gut. Sollen wir's dann gleich am Montag den dritten Mai machen? (*Fine. Shall we-it then right on Monday the third May do.*) which is best translated as *Fine. Shall we do it right on Monday the third of May.* For an explanation of the figure cf. page 99.

done in the experiments presented in this book. In the figures there are only minor errors in the segmentation of the speech signal, except for the syllable boundaries within the words sollen and machen in Figure 4.3.

For each 10 msec frame an F0 value was computed automatically from a 30 msec window of the speech signal using the DPF0–SEQ algorithm described in [Kie97, Sec. 6.3]; it is given in Hz in the Figures. In regions of unvoiced excitation the F0 value is arbitrarily set to zero. In each of the figures at the ordinate the minimum and the maximum F0 values in the entire turn are given. The dashed horizontal straight line in the figures crossing the F0 contour corresponds to the F0 base value, defined as the median of all the non–zero F0 values in the utterance.

The duration of the syllable nuclei (measured in msec) were normalized with respect to the speaking rate in the turn and with respect to the phoneme intrinsic mean and standard deviation using the equations given in Appendix A.2. The result of this normalization is a value without a unit of measure which indicates how fast (negative values) or how slow (positive values) a phoneme has been uttered relatively. This value is represented by the ordinate in the figures. It is for

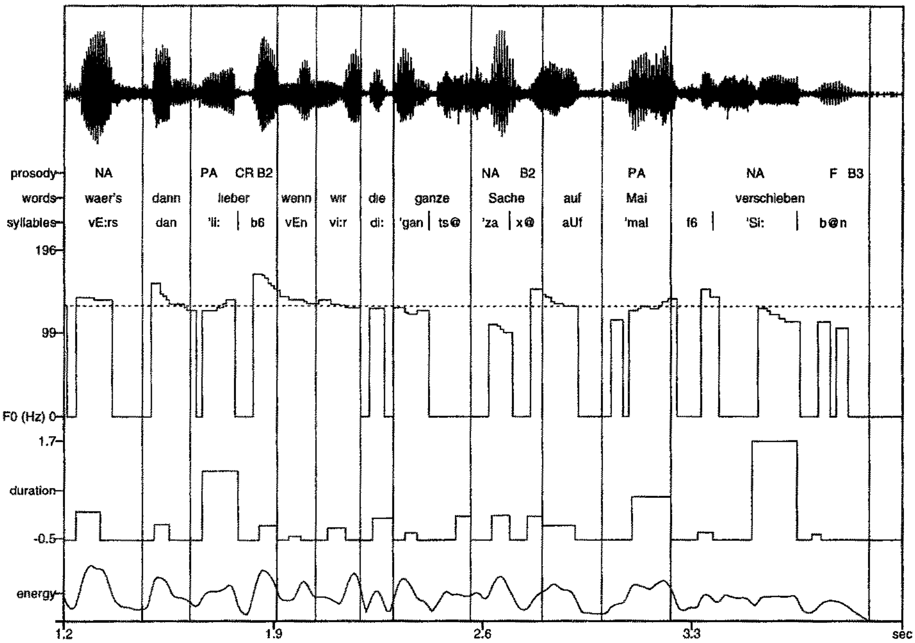


Figure 4.4: Speech signal and acoustic correlates of basic prosodic attributes for the utterance ...wär's dann lieber, wenn wir die ganze Sache auf Mai verschieben. (... *would-it then prefer, if we the whole thing to May move.)* which is best translated as *Then ... would prefer that we move it to May.* For an explanation of the figure cf. page 99.

each syllable nucleus plotted over an abscissa interval which corresponds exactly to the speech segment of the nucleus, the length of which therefore is the absolute (unnormalized) duration. The values outside of syllable nucleus segments have arbitrarily been set to the smallest normalized duration in the entire turn.

The contribution of the intrinsic normalization becomes obvious when comparing, for example, the phoneme /I/ of dritten and the /aI/ of Mai in Figure 4.3, which have the same normalized duration though the segment of /aI/ is about twice as long as the one of /I/. Another example is the /a:/ of gar and /I/ of nicht in Figure 4.5, which both have the same segment length, but the normalized duration of the /I/ is much greater. In the previous section we discussed the basic prosodic attribute speaking–rate; note that it is much higher in the phrase sollen wir's dann am Montag than in the phrase den dritten Mai machen (Figure 4.3).

In [Kie94b] we showed that in German the energy is not very important for the purpose of accent or boundary detection; this is also supported by the results in [Nöt91b]. In the figures the energy is depicted for the reason of completeness. It was computed based on the equation in Appendix A.1.

Intonation

The compound attribute *intonation* is defined as the distinctive usage of pitch [Koh77, pp. 126–127]. Since the F0 base value is speaker–dependent, especially influenced by sex and age, this definition implies that the term intonation always refers to changes in the pitch contour with respect to a certain context, which usually covers more than one syllable. Distinctive pitch contours are general patterns like low and high tones or contour movements described as *fall* (label F), *rise* (R), *fall–rise*, *rise–fall*, and *continuation–rise* (CR). The latter denotes slightly rising or level F0 but being above the utterance specific *pitch base value*. The base value can be defined as the minimum in the utterance [Bat89]. This definition has the advantage that it is easy to measure in the F0 contour. However, we prefer to define it as the average F0 in unmarked syllables, because especially a relatively low F0 can deliberately be used to mark accents, but this should not be taken as base value. The measurement for a base value defined in this way is difficult; we use the median of the pitch in an utterance as an approximation. Since in the context of this book, we will only deal with the function of intonation at the end of clauses, and furthermore, the relevant contours are F, R, and CR, nothing but this has been labeled in the figures of this section, cf. the description of intonational sentence mood, Section 4.2.2. Note also that the intonation labels differ from the commonly used tone sequence approach, however, they represent what will be used in the ASU systems described in this book; for a discussion cf. Section 5.2.3.

"Small" pitch movements do not have distinctive function [Leh70, Roy83]. Since in this book statistical methods will be investigated, we do not have to care about the exact meaning of "small"; the statistical algorithm will learn it based on sample data. Figure 4.3 shows examples of different types of F0 contours:

- gut carries a rise immediately followed by a fall. On a phonological level these are considered as distinct patterns opposed to a rise–fall pattern, because they have different function: As will be outlined in the next section, the rise marks an accent, whereas the fall marks this one–word utterance gut as a statement.
- A fall–rise on the first syllable of Montag marks the syllable as accented.
- The F0 contour on the last syllable of Montag is level and it is above the base value, so it is identified as a continuation–rise.

Accentuation

The term *accentuation* refers to syllables, which are perceived as being more prominent than the syllables in the context [Koh77, p. 122]. Mainly intonation and/or a change in normalized duration, and/or an increase in loudness can be used to accent a syllable. In which way accentuation is realized depends on the specific word, the context and the speaker [O'S83, Bat89]. Again examples can be found in Figure 4.3:

- The strong F0 fall–rise contour on /mo:n/ of Montag together with a slight lengthening of /o:/ signals an accent (label PA). The syllable /drI/ of dritten and the word Mai are marked in a similar way.
- The word gut is strongly accented (label EA) marked by a lengthening of the syllable nucleus and by very high and rising F0; note that the falling F0 at the end of gut signals the end of a clause, as it will be outlined in the next section.
- The rather low F0 on the words sollen and gleich indicates a rather weak accent compared to the ones mentioned above (label NA).

The signal shown in Figure 4.4 is produced by a different speaker:

- The first syllable of lieber and the word Mai carry major accents. In contrast to the examples in Figure 4.3 this speaker seems to prefer to mark the accents mainly by lengthening of the syllables, which is accompanied by a slight F0 movement. This difference between the two speakers holds also when looking at other speech samples not shown in the figures.
- In contrast, the minor accent on the syllable /za/ of Sache is only marked by low and falling F0.

As for pitch the examples show that different kinds of pitch movements on the syllable itself or a level pitch, which is significantly distinct from the surrounding syllables, can mark a syllable as accented. Furthermore, accentuation is not a binary feature, but accents of different strength can be observed. In the phonological literature these different degrees of accentuation are often categorized, for example [Bie66] distinguishes seven categories. However, there is no acoustic correlate which justifies any categorization. The accents of different strength rather build a continuum. Furthermore, an absolute measurement of the degree of accentuation does not seem to be useful, because accentuation is a relational phenomenon: a syllable is accented, if it is more prominent than others in a certain context, and a syllable is strongly accented, if it is more prominent than other accented syllables in a turn. This will be further investigated in Section 8.4.

When a word is uttered out of context, a distinct syllable is produced more prominent; this is called the *lexical accent*. The position of the lexical accent sometimes depend on dialectal influences. In long compound words more than one syllable can carry a lexical accent. In these cases one distinguishes between main and

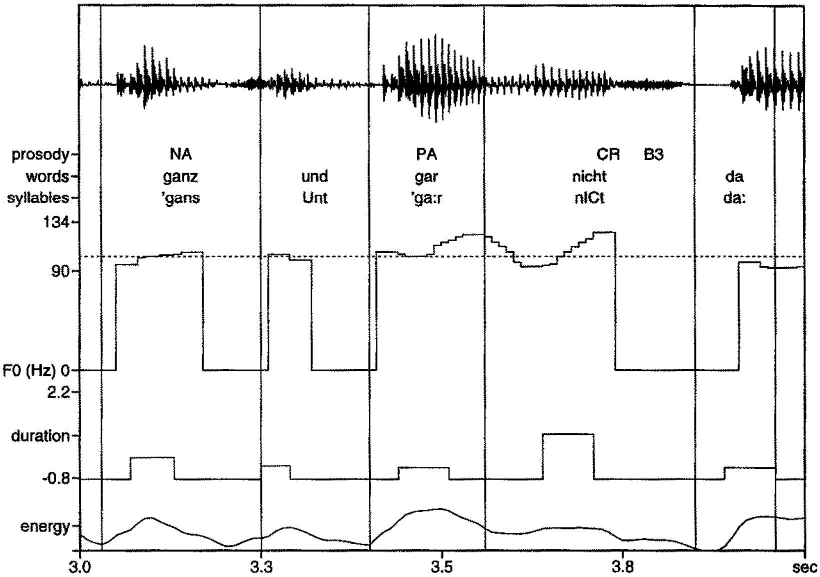


Figure 4.5: Speech signal and acoustic correlates of basic prosodic attributes. (Word–by–word) translation of the utterance: ... *not at all, because* ... For an explanation of the figure cf. page 99.

secondary accent. In continuous speech only some words carry an accent, which is realized on the lexical accent syllables. Exceptions are in the case of contrastive or emphatic accentuation, cf. Section 4.2.7. If a sentence is spoken in isolation disregarding any context (that is called *out–of–the–blue*), usually certain words are accented [Lea80a, Vai88]. In German this is in theory the last content word in a phrase, and furthermore the final phrase accent in a sentence should be the most prominent [Koh77, Uhm91], cf. also the evaluation of a speech corpus presented in Section 5.1.4. These accents are called *default* [Bec80], *neutral* [Fér93], or *normal* [Kip66]; the definitions are only slightly different but in all cases intuitive and informal. In this book we will use the term *default accent*.

Prosodic Phrases

Prosodic phrases are mainly identified through their boundaries. These are word boundaries which have characteristic fall, rise or continuation–rise pitch contours, and/or the last or the last two syllables in the phrase are lengthened (*phrase final lengthening*). Furthermore, a pause can mark major boundaries. Phrasing is also supported by the phrase accents since by default the last content word in a phrase is accented.

Prosodic phrases of different strength can be ordered in a hierarchy: a stronger phrase can include several weaker phrases and so on. In contrast to the accents of different strength, prosodic phrases can be defined in absolute rather than in relative terms. Therefore and because of their relationship to syntactic entities it makes sense to define a hierarchy of such phrases. Unfortunately, in the literature there is no unique definition of different prosodic phrases. We will name the most frequently used terms in the following. Surveys can be found in [Wig92c, Wig92b].

At the lowest level are *clitic* word groups, which are lexical words spoken in sequence in a way that they acoustically build an entity. This is often the case within prepositional phrases like in nach Frankfurt (*to Frankfurt*). These word groups are also called *prosodic words*. At the lexical word boundaries within clitic word groups assimilations and elisions are frequent; for example, in the word group haben wir (*have we*) with the corresponding canonical pronunciation /hab@nwi6/ an elision of the vowels /@/ and /i/ is frequent so that pronunciation becomes /habnw6/; furthermore in this example, almost always an assimilation of the phones /bnw/ occurs such that they are reduced to /m/ resulting in /ham6/ as a pronunciation of haben wir. Reasons for these phenomena are a "reduction in effort" [Koh95], that is, minimizing the articulator movements during speech production, and rhythmic constraints [Koh77, p. 214], which, however, are not well investigated yet in the context of spontaneous speech (as for an investigation of an English speech corpus cf. [Man92]). In Figure 4.4 for example, the lexical words wäre es (*would it*) are reduced to the prosodic word wär's.

The boundaries between subsequent prosodic words can be perceived, but they are not marked prosodically. In contrary, *prosodic phrases* are speech units, which are separated by prosodic means. Often, at least two levels of prosodic phrases are distinguished, which are called *intermediate* and *intonational* phrases. An intermediate phrase contains several prosodic words, whereas an intonational phrase consists of intermediate phrases. The highest level considered in the literature is usually the sentence. Most authors do not investigate turns, which can contain more than one sentence. Some authors identify intonational phrases with sentences, others define a sentence to be made up of several intonational phrases.

Since the use of these terms in the literature is not unique, for this book, we want to distinguish between *prosodic clause boundaries* (label B3) and *prosodic constituent boundaries* (label B2), because these are directly related to syntactic structures, and they can be distinguished by prosodic attributes: Prosodic clause boundaries are much stronger marked than prosodic constituent boundaries; the first ones are often marked by intonation and duration and sometimes by pauses, whereas prosodic constituent boundaries are often only marked by small intonation movements and are never marked by pauses. We use *prosodic phrase* as a cover term for both prosodic clauses and prosodic constituents, respectively.

Examples for differently marked prosodic boundaries can be found in the figures:

- Prosodic clause boundaries are after the words gut, Montag, and machen in Figure 4.3. All of them are strongly marked by *intonation*, *phrase final lengthening*, and by a *pause*.
- A prosodic clause boundary has not necessarily to be marked by a pause. In Figure 4.5 the prosodic clause boundary after nicht is only marked by *F0 rise* and by *lengthening* of the /i/; the short silence after nicht is due to a closure before the plosive.
- The prosodic clause boundary after verschieben in Figure 4.4 is marked by an *F0 fall*, by a *pause*, and by *phrase final lengthening*. In contrast to the examples above the phrase final lengthening includes the two final syllables: The nucleus of the pre–final syllable /Si:/ is extremely lengthened which is due to a superposition of accentuation and of phrase final lengthening. The last syllable is lengthened as well even if in Figure 4.4 the normalized nucleus duration is rather small. The latter is due to the fact that the syllable nucleus /@/ is reduced to zero. Therefore in the classification experiments described in [Kie97], the results of which are used in this book, both the duration of the syllable nucleus and of the entire syllable is used. We define the normalized syllable duration as the average of the intrinsically normalized durations of the phones constituting the syllable, where the duration of an elided phone is zero.
- A prosodic constituent boundary marked by *intonation* can be found at the end of the word lieber in Figure 4.4. Note that although being a syntactic clause boundary and although there is a strong pitch rise at the end of the word it is perceived as a minor prosodic boundary.
- A further prosodic constituent boundary can be perceived after the word dann in Figure 4.3. It is marked as well by *higher F0* and by *lengthening*; however, both are not very distinct.
- Another prosodic constituent boundary is after the word Sache in Figure 4.4; it is only marked by a slight *continuation–rise*, expressed by an F0 above the utterance's base value.

Sometimes boundaries are additionally marked by laryngealizations: [Kie97] found that prosodic phrase boundaries more often cooccur with laryngealizations than other word boundaries.

In addition to the boundary markers some authors report on a characteristic slope of the pitch contour for certain types of prosodic phrases [Lad86, Pie80, Ste89]. However, it has not been proven yet if this holds for spontaneous English speech as well as for read speech, and it is questionable, if it is valid also for German [Fér93, p. 106], where in general the F0 dynamic of utterances is much lower than in English [Adr91].

Rhythm

The compound attribute *rhythm* in German and in English is often defined by the number of accented syllables per time; both are so called *stress timed* languages: the time interval between accented syllables tends to be equal. In contrast, in languages like French the number of syllables per time is about to be constant during an utterance (*syllable timed* language). In the case of spontaneous speech rhythm is especially not well investigated yet. Nevertheless, rhythm does not just seem to play a role in read speech: an analysis of accent placement related to rhythmic structures in discourse is presented in [Bec90]. The strategies for keeping the time interval between accented syllables constant are different between English and German. In both languages unaccented syllables can be reduced. In English additionally the phenomenon of *stress shift* can be observed, where a syllable within a word is accented which does not carry the lexical accent [SH92, Ros92]. This might to a smaller extent also occur in German. A theory for rhythmic patterns of spontaneous speech has been introduced in [Ume94]. In their English speech material they identified a hierarchy of rhythmic patterns including the repetition of amplitude–accents (*beats*) in addition to pitch–accents both at regular time intervals. For rhythmic reasons also a word which is not semantically important might carry an accent.

Hesitation

Hesitation is a further compound prosodic attribute. It differs from the attributes described so far, because it does not contribute to the meaning of an utterance, but it marks a conflict between speech planning and speech production. It signals that the speaker is still thinking and wants to continue. Therefore it is a mean to hold the floor, that is, it avoids an interruption by the dialog partner. Basic prosodic attributes which characterize hesitations are pauses and extraordinary lengthening of syllables or words. Filled pauses can be used to mark a pause explicitly as a hesitation. In Figure 4.3 the word den is hesitated, which is indicated by the extraordinary lengthening and probably also by the rather low and slightly falling F0. This special case of hesitation is called *hesitation lengthening*. Because it can be confused with prosodic phrase boundaries and in accordance with the labeling system defined in [Rey94], these prosodic attributes will be labeled with B9.

Form versus Function

A general problem for research on prosody and for the use of prosodic information in ASU is that the different compound attributes influence each other especially when they occur very close in time. This and speaker–dependent variations make it possible for the same *function* to be realized by different *form* and vice versa the same form can have different functions.

For example, in Figure 4.4 the form of the F0 contour of the words gar and nicht is very similar, however the function is different: in the first case it marks an accent and in the second case it marks a continuation–rise. Other examples of different realizations of accents or boundaries by basic prosodic attributes were already mentioned earlier in this section (pages 104, and 107).

Examples for the same function realized by different forms can be found in Figure 4.4: The words lieber and verschieben both are accented and are immediately succeeded by a prosodic clause boundary; lieber carries a continuation–rise whereas the pitch at verschieben is falling. This has consequences for the use of pitch for marking the accent: the pitch on the first syllable of lieber, which carries the lexical accent, is relatively low and slightly rising; in contrary, the pitch on the first syllable of verschieben is relatively high and on the second syllable, which carries the lexical accent, it is falling.

Note that also coarticulation affects the F0 contour, but not the overall (perceived) pitch contour. For example, the small F0 fall at the very end of the word lieber is due to contextual effects (the preceding unvoiced plosive) and does not have any linguistic meaning; the F0 is on the average rising from the beginning of the word to the end. If one takes into account that the F0 on the first syllable is rather low due to the accent marking, the pitch contour relevant for the boundary marking can be described as slightly rising, that is, as continuation–rise [Bat96a].

The divergence between form and function continues also when considering the next level, the linguistic roles of the compound prosodic attributes. Accentuation can especially have very different functional roles as outlined in Section 4.2. A discussion of this topic can be found in [Cut83] and [Kie97, Sec. 3.2].

4.2 The Functional Roles of Prosody in Human–human Communication

4.2.1 Remarks on the Importance of Prosody

The compound prosodic attributes as described in the previous section are the carrier of the functional roles of prosody in human–human communication. The main functional roles are

- the structuring of turns into clauses and smaller phrases,
- the marking of the focus,
- the determination of the sentence mood, and
- the transmission of emotion.

These roles can contribute to the disambiguation of syntactic and semantic structure, and the interpretation of the dialog act. In the following sections, examples will be given where prosody helps to disambiguate on these different levels. Note that the examples are presented in isolation. When they would occur in real–life speech, in most cases ambiguity can probably also be resolved either from the situational or from the linguistic context. Thus prosody is often redundant leaving it at a discretion of the speaker to use prosodic cues. However, prosody usually will support the *intelligibility* of a turn. This can, for example, be concluded from the results presented in [Sil93], which show that humans understand synthetic speech much better, if an appropriate prosody model is used even in cases where syntax and semantics are relatively simple as in a name–and–address information service. It is perhaps easier to resolve ambiguities based on prosodic rather than on contextual information or as [dP93] puts it: "*Appropriate prosody facilitates the communication process ... by adding redundancy ..., thus reducing the cognitive load on the listener*". Also in [Hou92] it is stated that the acceptability of synthetic speech depends to a great deal on the appropriateness of its prosody.

Infants respond to different prosodic patterns in adult utterances directed to them at an age of around three months, that is, long before lexical comprehension starts [Pap81, Cry79]. This also indicates that prosody plays an important role in human–human communication, because otherwise infants would not "spend effort" on learning prosody. Moreover, prosody plays an important role in the therapy of very young hearing disabled children. Speech uttered by them is often hard to understand, which is usually more due to prosodic than to phonetic misproductions [Nor94c, Nor95].

In any case, it can be expected that a speaker wants to be understood and therefore he would try his best to support the intelligibility of his utterance. But the fact that prosodic information often is redundant causes a lot of variation among speakers in the use of prosodic information [Hel85], which has to be taken into account if using prosody in ASU. However, we believe that in contrast to many publications about prosody, the more frequent use of prosody in speech and thus the most interesting aspect concerning the integration into ASU is not the discrimination between really ambiguous interpretations of words or sentences but the role it plays concerning the general intelligibility. That means, prosody can "just" become one more source of — as we believe important — information, which might in ASU allow for a faster or more precise analysis.

4.2.2 Sentence Mood

Sentence mood is often determined by grammatical cues, for example questions can be indicated by certain particles (*Wh*–questions) or by the word order, where the verb is at the first position in a sentence. Apart from this there are cases, where such indicators are not present. This is especially true for elliptic clauses, which are very frequent in spontaneous speech, cf. [Wei75, Kie93]. In these the sentence mood can be disambiguated by prosodic means only. More specifically, the intonation contour at the end of the utterance is the most important indicator of sentence mood; other prosodic attributes like duration or voice quality might play a role as well, but they have not yet been investigated in this context.

Altmann has elaborated a very detailed model for the sentence mood in German, which consists of a hierarchy of categories characterized by a set of grammatical and intonational features [Alt87, Alt93]. For this book we only want to consider sentence mood determined by intonation and we restrict ourselves to the three major categories of contours, which are *fall*, *rise*, and *continuation–rise*, cf. Section 4.1.2, page 103. If no syntactic cues mark the sentence mood

- a *falling* intonation marks an utterance as a *statement*,
- a *rising* contour indicates a *question*, and
- *continuation–rise* signals that there is a (prosodic) clause boundary, but that the speaker has not finished his turn yet.

The latter occurs often at positions, where there would be a comma in written language, for example at boundaries between main and subordinate clause or between the elements of an enumeration.

Let us consider the following two subsequent turns from a dialog from the VERBMOBIL corpus of dialogs between German speakers:

A: In der ersten Mai–Woche is' noch jeder Nachmittag frei — in meinem Terminkalender. So ab vierzehn Uhr. (4.1)
In the first week of May there is still each afternoon available — according to my calendar. After about two pm

B: Gut. Sollen wir's dann gleich am Montag, den — dritten Mai machen? Vielleicht um halb vier?
Okay. Shall we do it right on Monday, the — third of May? Maybe at half past three?

The speech signal of the first part of the turn of speaker B is depicted in Figure 4.3, page 101. The prosodic attributes realized in the speech signal have been roughly translated into punctuation and are inserted in the word chains above. The speaker did mark the second syllable of Montag (*Monday*) with a strong continuation–rise (above indicated by a comma), which signals that despite the subsequent pause the turn is not over, but the pause has to be interpreted as a hesitation, because in this case the speaker needs time to figure out the date of the Monday he has in mind. The hesitation in this example is continued by the extraordinary lengthening of the word den (*the*). Another example for continuation–rise can be found in Figure 4.4, page 102 at the word lieber (*prefer*), where it indicates a boundary between main clause and subordinate clause. Note that this is redundant from the point of view of syntax, because the subordinate clause starts with the conjunction wenn (*if*). However, the function of the continuation–rise at this boundary is also to hold the floor.

Continuation–rise can also affect the meaning of turns. Consider examples (4.2) and (4.3), which have the same wording and the same phrase structure. The meaning differs only depending on whether the pitch contour on the same word is a fall or a continuation–rise:

Peter hat geträumt. Paul hat einen Brief an Maria geschrieben. (4.2)
Peter dreamt. Paul wrote a letter to Mary.

Peter hat geträumt, Paul hat einen Brief an Maria geschrieben. (4.3)
Peter dreamt that Paul wrote a letter to Mary.

In the first example a falling contour on geträumt (*dreamt*), indicated by a point, splits the turn into two independent sentences. In the second example a continuation–rise, marked by a comma, indicates that the second phrase is a subordinate clause specifying what Peter dreamt about.

The other two types of sentence mood can also be found in Figure 4.3, page 101. The end of the word gut (*okay*) is marked by a falling contour indicating a statement, and the word machen (*do*) carries a rising contour emphasizing that the utterance is a question, which is already determined by the verb–first word order: sollen (*shall*) is at the beginning of the sentence. Especially in the case of elliptic clauses, such as the word gut in this example, the determination of the sentence mood can only be done by prosodic means and it affects the meaning of the utterance:

- gut., as in the example, means "that is fine", whereas
- gut? would mean "Really? That's fine for you?"

4.2.3 Phrase Structure

The most important role of prosody within the scope of this book is the structuring of turns into *prosodic phrases*: words which "belong together" from the point of view of meaning are grouped into phrases, and – more important – it is widely agreed upon that there is a high but no full correspondence between prosodic and syntactic phrase and clause boundaries respectively [Fér93, pp. 59–60], and [Ste89, Ste92, Wig92c]; for an evaluation of corpora with respect to this cf. Sections 5.1.4 and 5.2.7. Note that this also depends on the underlying syntactic theories. Nevertheless, the prosodic boundaries identified in Figures 4.3 through 4.5 correspond to syntactic boundaries in any theory, cf. Section 4.1.2. However, in Figure 4.3, page 101, the pause after Montag (*Monday*) and the continuation–rise on the last syllable of Montag together signal a prosodic clause boundary; from the point of view of syntax, there should only be a minor boundary between constituents. A minor prosodic boundary is for example in Figure 4.4, page 102, after the word Sache (*thing*) where a pitch rise is perceived due to a F0 which is a little higher than the average in the utterance. It separates the two constituents die ganze Sache (*the whole thing*) and auf Mai (*to May*).

In the following we will have a closer look at the communicative function of prosodic structuring of speech. Let us start with two turns with the same wording, which differ by the boundaries between two clauses:

Peter sagte, Paul ist dumm. *Peter said, Paul is stupid.* (4.4)
Peter, sagte Paul, ist dumm. *Peter, said Paul, is stupid.*

In written language these clause boundaries are indicated by commas. In speech prosodic boundaries can take the part of the commas. Thus the boundaries on the one hand separate clauses from each other and on the other hand they group words into (prosodic) phrases. Note that this is a typical example as it is often given in the literature to "prove" the need of prosody, because of the "obviously" different meaning depending only on the position of the clause boundaries. However, depending on the context even here prosodic information can become redundant; for example, after the following question, only the first of the two alternatives can follow:

Was hat Peter gesagt? *What did Peter say?* (4.5)
Peter sagte, Paul ist dumm. *Peter said that Paul is stupid.*

However, it is probably the case that a listener gets the meaning of an utterance easier or faster if prosody is used by the speaker in an appropriate way. Comparing examples (4.6) and (4.7) it becomes more obvious what is meant with "prosody supports the intelligibility of an utterance" (from [Blo94]):

Ich brauche eine Verbindung nach München abends. (4.6)
I need a connection to Munich in the evening.

Ich brauche eine Verbindung nach München. Abends möchte ich dort sein. (4.7)
I need a connection to Munich. In the evening I want to be there.

Until the word abends both utterances are phonemically identical. Depending on the continuation of the turn, there is a main clause boundary before or after abends. There is no real ambiguity between these examples but already while a person speaks the listener usually starts retrieving the meaning [Car86, pp. 44–51]. When the word abends has been uttered the listener does not know about the continuation, so that at this point the interpretation is ambiguous and when only the information about the spoken words is available the meaning disambiguates after some more words have been uttered (so called *"garden path problem"*). An appropriate prosodic boundary marking would support the understanding, because it disambiguates earlier. Moreover, if accidentally the boundary were marked at the wrong position it could irritate a listener.

The following sentence is also an example, where prosody can help to structure, so that it might be easier to get the intention:

Ich möchte morgen früh mit dem Zug um zehn Uhr von Hamburg über Frankfurt nach München fahren. (4.8)
(I want tomorrow morning with the train at ten o'clock from Hamburg via Frankfurt to Munich to go.)
Tomorrow morning at ten o'clock, I want to go by train from Hamburg to Munich via Frankfurt.

In this example, syntactic ambiguity lies in the structuring of the adverbials (morgen, früh) and in the attachment of the prepositional phrases (henceforth *PP–attachment*). For the different parts of the sentence the grouping into phrases is given in the following; all combinations of the two alternative structures of morgen früh

1. (morgen) (früh)
2. (morgen früh)

with the five alternatives of mit dem Zug um zehn Uhr von Hamburg über Frankfurt nach München fahren

1. (mit dem Zug) (um zehn Uhr von Hamburg über Frankfurt nach München fahren)
2. (mit dem Zug um zehn Uhr) (von Hamburg über Frankfurt nach München fahren)
3. (mit dem Zug um zehn Uhr von Hamburg) (über Frankfurt nach München fahren)
4. (mit dem Zug um zehn Uhr von Hamburg über Frankfurt) (nach München fahren)
5. (mit dem Zug um zehn Uhr von Hamburg über Frankfurt nach München) (fahren)

are possible and yield ten alternative structures.

The syntactic structuring, especially the PP–attachment can also affect the semantics of the sentence, for example:

(mit dem Zug) (von Hamburg . . . fahren)

versus

(mit dem Zug von Hamburg) (fahren)

In the first case von Hamburg is attached to fahren that means the person wants to depart in Hamburg. In the second case von Hamburg is attached to Zug, that is, the person wants to take the train which leaves Hamburg and not for example Bremen, but he wants to depart somewhere else, for example, in Hannover. In most real–life situations where this sentence will be uttered the actual prosodic phrasing will probably not affect the meaning, because it is already disambiguated by the context. Nevertheless, we believe that in human–human communication structuring supports the general intelligibility or acceptability also in this case. If the sentence were spoken monotonously without prosodic boundary marking it would probably be more difficult, that is, it would take more time, to get the intention. If at all, these boundaries are likely to be marked as prosodic constituent boundaries, cf. page 107. The prosodic phrasing, however, might not correspond to the "correct" syntactic structure underlying the intended meaning. We consider the following phrasing as the default case, which we expect most likely to be realized (cf. the ERBA prosodic constituent boundary labels, Section 5.1.3):

(Ich möchte morgen früh) (mit dem Zug) (um zehn Uhr) (von Hamburg) (über Frankfurt) nach München fahren. (4.9)

In spontaneous speech elliptic clauses occur frequently. These as other free phrases (cf. Section 5.2.1) cause a lot of ambiguities, because it is often not clear from the point of view of syntax if a phrase is an elliptic clause on its own or if it is part of a clause, and moreover to which clause it is bounded if it is at the boundary between two clauses. For example the word chain

ja zur Not geht's auch am Samstag (4.10)

taken from a VERBMOBIL corpus can be structured into clauses in many ways; Table 4.2 shows the different parts of the turn together with possible functions and respective translations into English. This results in 36 alternatives, all of them being meaningful [Nöt94a]. In the following we will consider four examples in a possible context (A and B name different speakers; in Appendix B.1 all 36 cases are given):

1. **A:** Können sie diese Woche auch noch zu einem anderen Termin?
   Do you have another date available this week?
   B: Ja zur Not geht's auch am Samstag.
   Well, if necessary, Saturday is also possible.

2. **A:** Könnten wir uns am Freitag treffen?
   Could we meet on Friday?
   B: Ja. Zur Not. Geht's auch am Samstag?
   Okay. If necessary. Is Saturday possible as well?

3. **A:** Dann bleibt uns nur noch das Wochenende übrig.
   Then only the weekend remains.
   B: Ja zur Not geht's auch. Am Samstag.
   Okay, it's possible, if necessary. On Saturday.

parts of the turn		function	translation	punctuation
ja	1	elliptic statement	*yes*	.
	2	elliptic clarifying question	*okay*	?
ja ...	3	turn initial particle	*well*	-
zur Not	1	elliptic statement	*if necessary*	.
	2	elliptic clarifying question	*really*	?
zur Not ...	3	sentence adverb	*if necessary*	-
geht's	1	elliptic question	*it' possible*	?
... geht's	2	statement	*... it's possible*	.
... auch	1	question	*even*	?
	2	statement	*as well*	.
am Samstag	1	elliptic statement	*on Saturday*	.
	2	elliptic question	*on Saturday*	?

Table 4.2: Ja zur Not geht's auch am Samstag: Functions of the different parts of the turn.

4. **A:** Naja, zur Not geht's auch gegen Ende der Woche.
   Well, if necessary the end of the week were okay as well.
   B: Ja? Zur Not geht's? Auch am Samstag?
   Okay? It's possible, if necessary? Even on Saturday?

In these examples, the syntactic structure and the sentence mood are indicated by punctuation. Depending on this, the identical word sequences get completely different meaning. In speech the only way to disambiguate this is by prosodic means.

The phrase structure of turns does not only play a role for syntactic analysis, but it has also relevance for the discourse structure. Turns often consist of several dialog acts. It is important for the continuation of the dialog to be able to classify the dialog acts which have been uttered previously [Mas93]. In the second of the above examples the turn consists of three clauses and two dialog acts: According to [Jek95] the two clauses Ja. Zur Not. together are to be classified as *acceptance of a date* and the subsequent utterance Geht's auch am Samstag? is a *suggestion of a date*. Before the dialog acts can be classified, the discourse structure, that is, the boundaries between the dialog acts, has to be determined. They coincide with clause boundaries in a way that every dialog act boundary is a clause boundary as in the example above. It is difficult to determine the dialog act boundaries based on a syntactic analysis only. Therefore, a few studies investigated the prosodic marking of such boundaries [Cah92, Swe92, Hir94a]. They independently from each other found that discourse unit boundaries are marked extraordinary strong as prosodic boundaries, that is, the F0 variation is greater, and the pauses are longer. Furthermore, often the F0 range is greater and the F0 maxima and the intensity maxima are higher at the beginning of such units. These results hold for dialog acts in spontaneous speech as well as for topic shifts within elicited monologues. In [Swe92] it is reported that humans can reliably perceive such boundaries even in unintelligible speech. (The test samples had been made unintelligible by band–pass filtering.)

4.2.4 Focus and Scope

Despite its contribution to the structuring of utterances as mentioned earlier (page 105) the most important roles of accentuation are the marking of the *focus* of an utterance and of the *scope* of particles. Both are semantic terms.

Focus is usually defined as the important or *new* information with respect to the *given* information, cf. [Ban85, Nöt91a, Fér93] for an overview. The strongest accent in a sentence lies on one of the words which is in the focus. This is called the *focal accent* [Bat89]. Thus accentuation makes the new or important information more prominent with respect to the given information in an utterance. This holds for all the accented words in Figures 4.3 to 4.5, which were explained in Section 4.1.2. As a side effect of accentuation the important information is produced more clearly and therefore it is better intelligible than the not accented parts of an utterance; this has been proved by perception tests [McA91]. In [Kie97] an analysis of a spontaneous speech corpus showed that accented words are less likely to include laryngealizations than unaccented words, so, it can be concluded that the voice quality in general is better in accented words.

In this section we will consider the following examples:

A: Treffen wir uns? *Do we meet?* (4.11)
B: Treffen wir uns in Ulm. *Let us meet in Ulm.*

In the second sentence the word Ulm is in the focus and thus will usually be accented. If more than one word is in the focus normally the last content word carries the focal accent:

A: Treffen wir uns? *Do we meet?* (4.12)
B: Treffen wir uns um acht in Ulm. *Let us meet at eight in Ulm.*

In the second sentence the phrase um acht in Ulm (*at eight in Ulm*) is in the focus; the word Ulm will usually carry the focal accent. In both examples the focal accent coincides with the default accent position, because it is on the last content word of the sentence. [Fér93] argues that if a sentence is spoken out–of–the–blue, the entire sentence is in the focus, and thus the default accent can also be viewed as a focal accent, cf. page 105. In the context of

A: Wann treffen wir uns in Ulm? *When do we meet in Ulm?* (4.13)

only the word acht (*eight*) is in focus and will thus carry the focal accent:

B: Treffen wir uns um acht in Ulm. *Let us meet at eight in Ulm.* (4.14)

Note that these are constructed examples. In real dialogs answers are often elliptic and concentrate on the relevant parts, for example, instead of the full sentence in example (4.14) only (um acht).

In out–of–the–blue sentences the position of the focus can be ambiguous if only the wording is considered, but a speaker can disambiguate the focus position by putting the primary accent on a certain word; this can also affect the meaning of a sentence. For example

Treffen wir uns in Ulm? (4.15)

means *"Do we meet each other in Ulm and not in ..."*, whereas

Treffen wir uns in Ulm? (4.16)

could mean *"Do you think we will have time to meet each other, when we will be in Ulm?"* or it could mean something like *"Do we meet each other in Ulm, or do we talk on the phone?"*

Concerning the scope of particles, we will consider the following example: Given the context sentence

Peter weint. *Peter cries.* (4.17)

in the subsequently spoken sentence

Peter weint nicht *Peter does not cry* (4.18)

Peter as well as weint are given information. Thus the focus is on nicht (*not*), which can be accented. In this case the particle nicht negates the entire statement. However, also either Peter or weint (*cries*) can carry the focal accent in order to indicate the scope of the negation particle. If Peter is accented, a meaningful continuation of the sentence could be but Paul cries; on the other hand, if weint were accented then a meaningful continuation could be he laughs.

In this case accentuation contributes to the meaning of the utterance, provided there is no further context information given. In many real–life situations probably the position of the focus can be determined from either the dialog or the situative context.

As [Bos95] states, the position of the focus can also change the *presupposition* of a sentence rather then directly contribute to its semantic meaning. A presupposition is an implicit condition of a sentence, which is considered as being true. The following two sentences have different presuppositions:

Dann müssen wir noch einen Termin ausmachen. (4.19)
Then we still have to fix a date.

Dann müssen wir noch einen Termin ausmachen. (4.20)
Then we have to fix another date.

The semantics of both of the sentences is the same. However, the position of the focus changes the the scope and thereby the presupposition of the utterances, which results in a different translation of the particle noch (*still,another*). Note that it might depend on the semantic theory if presupposition is considered to be part of the meaning of a sentence or not. In any case it does not contribute to the semantic structure as described in Section 3.4 in the context of the semantic construction module of the VERBMOBIL system.

In [Bat91] the intonational marking of double versus single and broad versus narrow focus in read speech has been compared. For further topics as the focus being spread over different phrases and the related accent structure cf. [Cha76, Jac88, Bat88].

4.2.5 Pragmatics and Discourse

Besides its semantic function concerning the disambiguation of focus position and of scope, accentuation plays an important role in pragmatic and discourse interpretation [Hir93b]. The interaction between these levels has not much been investigated yet.

Here we want to concentrate on the modal and discourse particles. Often their interpretation is ambiguous, because the same orthographic word form can be used for different functions, so called homonyms. For example, the word well can be an adverbial or it can be a discourse particle. A comprehensive analysis of English speech data concerning the use and function of *cue phrases*, which include discourse particles, has been reported in [Hir93c]. A main result is that prosody is critical for the disambiguation of these phrases. Accentuation seems to be especially important in this context but also the marking of prosodic boundaries plays a role.

A similar analysis has been conducted for German on the VERBMOBIL speech data [Rip96c]. A large number of dialogs and particles has been investigated. The problem of the correct interpretation of particles seems to be even higher for German than for English, because the latter contains fewer particles. In German these particles are often homonyms that can have syntactic and/or semantic function as well as pragmatic and discourse function resulting in very different interpretations. The most frequent particle is ja. In the following we will give three examples for the particle also taken from the VERBMOBIL corpus, which show three different functions.

- Most frequently also is in this data used as discourse particle. Discourse particles usually are at the beginning of utterances, but they can also occur in the middle of utterances at the beginning of new dialog acts or topics. As a discourse particle also means well:

also ich dachte noch in der nächsten Woche ... (4.21)
well I thought still during the next week ...

- As a conjunction also means therefore or so and initiates a consequence as in

... bis ich wieder da bin also nicht vor sieben ... (4.22)
... until I'll be back again, so not before seven ...

- It rarely occurs as a modal particle where it can introduce a confirmation, an explanation, or a summary. It is difficult to translate also into English in this case; the best lexical translation might be then, if translated at all. The following is a typical example:

das war also Freitag dreiundzwanzigster Oktober (4.23)
that was Friday 23rd of October (then)

The interpretation of such particles can depend on the current or the preceding dialog act, the prosodic–syntactic structure, the sentence mood and the accentuation. According to the data investigated in [Rip96c] prosody seems to be a major cue in this context, however, in what manner it contributes and in general how such particles have to be interpreted is not yet well understood and is a complex problem in the case of spontaneous speech.

Therefore, we will explain the role of prosody in this context using constructed though plausible minimal pairs where prosody alone allows for a disambiguation of the syntactic function of the homonym. For example in

Das müssen wir wohl bedenken. *We really have to think about it.* (4.24)

the word wohl functions as modal particle being best but not equivalently translated as really whereas in

Das müssen wir wohl bedenken. *We should think about it carefully.* (4.25)

the accentuation marks wohl an adverb meaning carefully. In the following two examples the function and meaning of gut depends on the presence of a subsequent sentence boundary, which in speech has to be marked prosodically:

Gut. Machen Sie das. *Okay. You can do that.* (4.26)

Gut machen Sie das. *You do that well.* (4.27)

In example (4.26) the boundary after gut identifies it as a discourse particle, whereas in example (4.27) it functions as an adverb.

Finally, we want to have a look at the following two examples:

Finde ich schon. *I really believe that.* (4.28)

Finde ich schon. *I'll find it certainly.* (4.29)

In both cases the word schon functions as a modal particle. In the first case (4.28) where it is accented, it has contrastive function; it should be translated by really. In the second case (4.29) it expresses affirmation and is best translated as certainly.

Thus prosody does not alter the function but it alters the pragmatic interpretation of the particle.

The importance of accentuation for discourse is not limited to the interpretation of particles. For example, the role of accentuation of pronouns and proper names in discourse was investigated by [Nak93]. She found that pronouns, though representing given information, can be accented in order to "shift" or to "refocus" attention. This can often be observed after sub–dialogs. Furthermore, prosody seems to be an important cue for discourse structure as outlined in Section 4.2.3.

4.2.6 Contrastive Accent

If a speaker wants to make clear that a syllable, word, or constituent is in contrast to the given information he puts a somewhat stronger accent on it:

A: Wo bist Du hingegangen? *Where did you go to?* (4.30)
B: Ich bin ins Kino gegangen. *I went to the movies.*

A: Wie war's in der Oper? *How was the opera?* (4.31)
B: Ich bin ins **Ki**no gegangen. *I have been in the **mov**ies.*

In (4.30) Kino carries an "ordinary" focal accent. In (4.31) the speaker wants to make clear that he was not in the opera, so he puts a stronger accent on Kino.

If just a syllable is in contrast to the given information, this syllable can be accented even if it does not carry the lexical accent:

Ich meine Homberg nicht Homburg *I mean Homberg not Homburg* (4.32)

As for investigations concerning contrastive accent and how it differs from focal accent cf. [Ban85, Haa95b].

4.2.7 Emphatic Accent and other Paralinguistic Roles

Extraordinary strong accentuation is called *emphatic accent* [Ban85, Nöt91a]. It occurs in situations like the Das ist ja UNERHÖRT (*This really is* UNBELIEVABLE) or Das tut VERDAMMT WEH (*It* HURTS *like* HELL). In these cases the accentuation does not have the linguistic function as described above, but it rather carries emotional information. Often not only one syllable, but all syllables in a word or even more than one word in a phrase are accented.

For the sake of completeness we will also mention the important role prosody plays in the transmission of other emotions. A detailed analysis of acoustic–prosodic correlates of the emotional impression listeners get from utterances is presented in [Tis93]. Such emotions are anger, joy, rage, love, excitement, tension, self confidence, sensitivity, comfort. These are directly expressed in terms of variations of the basic prosodic attributes vowel duration, F0 and intensity. Mainly, the overall variance of these attributes within utterances and the contour of F0 and intensity within syllables, especially the position of the maxima, are considered as being important carriers of emotional information. Also [Fuj90] found that para–linguistic information as exhortation or disbelief can be partly distinguished by intonation and moreover the intonation is different from a "neutral" marking of accent or boundaries. In this work we do not consider this aspect of prosody any further. Note however that also emotion can sometimes contribute to the meaning of utterances. For example if the sentence

I am proud of you (4.33)

is uttered ironically, the irony should be interpreted as a negation on the semantic level.

4.2.8 Disfluencies

When speaking spontaneously people have to think about what to say and how to say it while they are speaking. This can cause asynchronies between the speech production and the intended thought, which causes disfluencies in the production process. These often take the form of hesitations, cf. Section 4.1.2, but also repetitions or interruptions followed by a repair or by a restart are frequent. An interruption can occur between words or even within a word. After the interruption the preceding word or several words can be corrected; this is called a *repair*. The correction can be in a way that a few or all words are replaced by others or that no word is corrected, but that new words are inserted. With *restart* one denotes phenomena, where the speaker discards the current syntactic construction totally and starts anew from scratch. For detailed categorizations of disfluencies and a frequency analysis of corpora cf. [O'S92a, O'S92b, Bat94b].

All cases of disfluencies cause problems in current ASU systems. Hesitations can disturb speech recognition, because words are extraordinarily lengthened or pauses occur at unpredictable, that is, non–boundary, positions. Word interruptions are a problem as well, because at the present state–of–art a word recognizer only uses models of lexical items. In syntactic analysis including stochastic language modeling with an n-gram repetitions and repairs cause severe problems, because they are sometimes hard to recognize on the basis of syntactic knowledge and it is often ambiguous which of the words spoken before the repair are corrected. This becomes obvious when looking at the following example taken from a VERBMOBIL corpus:

<Schmatzen> ja ist in Ordnung Montag | Sonntag den fünften und Montag den sechsten Dezember dreiundneunzig hab' ich denn als zweitägigen Termin hier in meinem Kalender notiert (4.34)
(<smack> yes is all right Monday | Sunday the fifth and Monday the sixth December ninety-three have I then as two–day date here in my calendar written)
Yes, it is ok Monday | Sunday the fifth and Monday the sixth of December ninety three. So, I wrote this as a two–day meeting in my calendar.

The vertical bar indicates the disfluency. The word Montag has been corrected by the speaker by Sonntag den fünften und Montag. When considering only the wording, at the position of the vertical bar could also be a sentence boundary. If the preceding Montag is interpreted as a right dislocation both sentences would be syntactically correct, and they would be meaningful if interpreted independently. When trying to get an interpretation of the entire turn it becomes likely that these are not two sentences but that the turn is one sentence with a disfluency. However, in a left–to–right processing based on the word chain alone the disfluency will be detected rather late.

In view of these problems caused by disfluencies it would be of great help, if they could be recognized in advance. Luckily several studies have shown that they are marked by prosodic means, which offers the potential to detect them very early in the speech recognition and understanding process [O'S93, Hir93d, DK95]. Repairs are often marked by hesitations and/or the substitute word or the new word being inserted carries an extraordinary strong accent.

In [Lic92] results of perception experiments are presented: humans detected 85% of all disfluencies; only 20% of the disfluencies were recognized before the word after the interruption started, and 6% of the disfluencies were recognized after the first word succeeding the interruption has been perceived. The remaining 74% of the disfluencies which were recognized at all were detected by the listener before the first word after the interruption has been uttered completely. This can only be due to prosodic markers put on the first word after the disfluency.

First approaches regarding the recognition of disfluencies in the VERBMOBIL corpora are presented in [Kie97]. The use of this information within word recognition or syntactic analysis will be left to future work, and will not be considered any further in this book.

4.3 Prosody in Man–machine Communication: A Literature Survey

This book addresses the problem of using prosodic information on different levels of ASU. Although the number of publications in the field of prosody increased very much in recent years, most researchers addressed the problem of detecting prosodic attributes, but only little work has been conducted towards the use of these attributes in man–machine communication. This section presents a literature survey, cf. also the summary given in Table 1.1, page 12.

4.3.1 Word recognition

In ASR prosodic information is usually not used at all, except for the energy, which is a component in the acoustic feature vector. To our knowledge, no work has been done to include for example F0 in the feature vector, because F0 is no direct acoustic correlate of phones, although the acoustic features are affected by prosody. The mel cepstrum features which are usually used in ASR encode the spectral structure mainly related to the formants but no information about pitch. Because the height of the formants depends on the actual pitch [O'S87, pp. 68], Singer tried to normalize the cepstrum features with respect to F0 [Sin92]; he achieved a slight improvement on a very small speech database. However, [Zot94] was not able to repeat this result on a large and more realistic database.

The state transition probabilities of HMMs imply an exponential distribution of the duration associated with one state [Nol87]. This is not well suited to model phone durations since these are Gamma distributed [Cry82, Wig92b]. Thus some researchers found that omitting the HMM state transition probabilities completely improves the word recognition rate [Ney93, Ken94]. Other researchers tried in different ways to introduce explicit duration modeling into HMMs. Work for isolated word recognition has been reported in [Lev86, Gue90, Ken91, Ljo94]. For continuous speech recognition Pierre Dumouchel compared different models [Dum93, Dum94b, Dum94a, Dum95]. He was able to reduce the errors by about 6% on speech read by one speaker by imposing constraints for the minimum and maximum phone duration, whereas a Gaussian mixture model for the phone duration did not improve the recognition [Dum95]. Furthermore, he incorporated a model of micro–prosody in a speech recognizer. It consists of Gaussian distributions modeling F0 and energy, which are attached to diphone segments. On data from three speakers extracted from books–on–tape this reduced the error rate by about 8% [Dum94a]. Explicit duration modeling by gamma densities was also used in the HMM continuous phone recognition experiments reported in [Lev89]. [Bis92] used different phone models for lexically stressed and unstressed syllables, which differed in the parameters of an explicit duration model. Experiments were conducted on the Wall Street Journal (WSJ, [Pau92]) corpus, which is a large collection of newspaper paragraphs read by untrained speakers. A reduction in error rate of 10% could be achieved. Similar improvements were achieved by [AD92] on the DARPA Resource Management speech corpus. HMMs as they are currently used in word recognizers might in general not be adequate to model prosody, because they basically model sequences of frames of constant length, whereas prosodic attributes are related to larger entities of variable length like syllables. The continuous variable duration HMMs described in [Ril95] and explicit stochastic segment models as developed by different researchers [Rou87, Ost89, Dig90, Leu91, Zha94, DM96b] might be better suited for the modeling of prosody, but this has not been investigated yet.

Therefore other researchers developed preprocessors or postprocessors to standard word recognizers, which use prosodic information. In [Wai88] as well as in [Aul84] experiments are described where stress patterns have successfully been used to reduce the lexicon size in an isolated word recognition task. However, this has not been embedded in a full system, so it is an open question, if this reduction in lexicon size would improve a subsequent word recognition step.

Phone duration models which depended on the lexical accent in addition to the phone context were used in [Ana95] to rescore the n-best sentence hypotheses. By this the word error on the best sentence hypothesis could be reduced by 10% for the WSJ task.

In [Nöt88a, Nöt89], cf. also [Nöt91a], prosodic filters for word lattices hypothesized by an HMM word recognizer were proposed: each multi–syllabic word where due to empty transitions in the HMM entire syllables were deleted was rejected. Furthermore, every function word hypothesized over a segment of speech which was automatically detected as accented was deleted from the lattice. This resulted in an increase of recognition rate by 5 to 10 percent points depending on the generated number of hypotheses.

Important work has been done by the group of Mari Ostendorf concerning the post–processing of the n-best sentence hypotheses computed by an HMM word recognizer [Ost93b, Vei93a, Vei93b, Ost94]. Each of these hypotheses is rescored by prosodic information. Rather than predicting a single prosodic label, probabilities are computed for all prosodic classes by computing for each word a set of syntactic features like the part–of–speech or the type of syntactic boundary after the word. The features are determined from an automatic parse of the word chain. Based on these feature vectors classification trees were trained to perform prosodic classification. Rather than a hard decision into classes at each word boundary probabilities for each of the prosodic classes were combined with the probabilities computed by an acoustic–prosodic model for the same classes, which yields an overall prosodic probability for the utterance. This in combination with the acoustic and the n-gram scores computed by the word recognizer is used to rescore the n-best word chains. By this the rank of the correct word chain could be improved on a corpus of ambiguous sentences read by radio news speakers [Pri91] as well as on a subset of ATIS, which is a spontaneous speech database for air travel information and flight reservation [Pol91]. In the experiments on ATIS the subset of utterances was chosen, where the correct sentence was among the top ten hypotheses computed by the MIT word recognizer. One parse per sentence hypothesis was picked at random among the alternatives. The average rank of the correct sentence hypotheses could be improved by 23%.

4.3.2 Linguistic Analysis

In ASU so far no work is known to the author on the use of prosody on the semantic or dialog levels, although it was proposed already two decades ago [Lea75], cf. also [Lea80b, Vai88]. [Hub90] proposes rules for the direct transfer of intonation units from English to Japanese. On the syntactic level work has only recently been undertaken by a few people, which are again the group around Mari Ostendorf [Pri89, Ost90, Pri90, Vei92, Vei93a, Ost93b] and by Andrew Hunt [Hun93, Hun94a, Hun94b, Hun95a, Hun95b].

In their first work Ostendorf et al. extended grammar rules by prosodic "break indices", so that at each word boundary a subset out of seven levels of breaks could occur. For the spoken word chain each word was classified into one of these break indices on the basis of an acoustic feature vector. These break indices were introduced in the word chains, which then were parsed using the extended grammar. This approach resulted in a decrease in the number of parses by up to 25% [Ost90, Pri90, Bea90]. Later they used the same approach as for rescoring the n-best sentence hypotheses, cf. Section 4.3.1, also for the rescoring of parses, which is flexible and rather independent from a specific grammar, and it is a stochastic method. All the experiments reported so far by this group concerning the use of prosody in parsing were conducted on pairs of ambiguous sentences read by professional radio news speakers. When using automatically determined prosodic boundary and accent information, in up to 73% percent of the cases the model decided for the correct out of two given parses, which were selected manually a priori [Vei93a, Ost93b]. Note that so far no reports on the application of this parse scoring algorithm on the ATIS (spontaneous speech) have been published, though the n-best sentence hypotheses rescoring has already been conducted (as described in Section 4.3.1), and the same algorithms were used for both tasks.

Hunt developed a similar approach: he as well computes for each word acoustic–prosodic features and syntactic features. The syntactic features are determined based on a parse of a word chain using the *link grammar*, which is a special kind of grammar developed at CMU [Sle91]. As Ostendorf et al. Hunt correlates the syntactic features with the prosodic features. However, he does not do this by computing probabilities for pre–defined prosodic classes; rather he directly computes correlations between these feature vectors using multivariate linear statistical analysis methods. With this he can score different parses of the same word chain without requiring a manually labeled training database. On the same corpus used by Ostendorf et al. (cf. above) with his approach 74% of the parses were recognized correctly in a leave–one-out training and testing method [Hun94b].

Note that due to the computation of the syntactic features the approaches of both Ostendorf and Hunt require that an entire sentence hypothesis has been parsed before the prosody model can be applied. Prosodic information is not incorporated directly into the search for the optimal parse.

4.3.3 Stochastic Language Models

As in other areas of ASR the use of stochastic language models can be useful. The group of Mari Ostendorf developed an approach which can be viewed as a mixture of bigram and trigram models to predict prosodic phrase boundaries [Vei90]. Given a word chain, this model can be used to generate alternative sequences of boundary markers. In 50% of the cases the boundaries actually produced by radio news speakers were consistent with one out of the top ten predictions. In previous work we used polygrams to score word chains where prosodic boundary labels were introduced between each pair of words [Kom94a]. This method was not used to rescore n-best sentence hypothesis, but it was used for the purpose of prosodic boundary classification; thus for each utterance the chains differed only in the prosodic labels not in the words, which in this case were the spoken words. The labels were alternatives scored by an acoustic–prosodic classifier. According to these scores the n-best chains were selected by an A* search. On a corpus of read speech 90% of the boundaries could be correctly recognized (11% false alarms). A similar approach, presented in [Leh95], uses trigrams based on word categories in a Viterbi search to predict prosodic phrase boundaries given the spoken word chain. For the two–class problem boundary vs. no–boundary on a subset of the VERBMOBIL spontaneous speech data, cf. Section 5.2.1, at 90% of the word boundaries the reference label was correctly predicted. Note that this includes the turn final boundaries. The reference labels were manually created on the basis of perceptual evaluation. Note that Viterbi search works only when using an n-gram with $n = 2$ or with some approximation also with $n = 3$.

The use of classification trees for the prediction of prosodic phrase boundaries given a word chain was first introduced by [Wan92]. For each word boundary features like the part–of–speech of the four surrounding words, the distance from the beginning and end of the utterance, the sentence type (for example indirect question), the utterance duration, the type of certain syntactic constituents, and a priori from acoustic data predicted accent values for the two words surrounding the boundary are computed. On a subset of the ATIS database a recognition rate of 82% for prosodic phrase boundary vs. no–boundary was achieved on a balanced subset of boundaries. Note that these also include utterance final word boundaries. Classification trees were in the same manner used for the prediction of prosodic accents on the basis of the text only [Hir93a]. A recognition rate of 85% for accented versus not–accented words was achieved on an ATIS subset.

This approach based on classification tree was adopted by [Ost93b] for the purpose of parse scoring, cf. Section 4.3.2. As an alternative in [Ost94] a hierarchy of prosodic units and a new statistical model is proposed, with which the probabilities for the partioning of a certain unit into subunits can be estimated given a labeled training corpus. However, the performance is below the one of the classification trees.

[Hun95a, Chap. 5] proposes the use of prosodic constraints in n-gram language models for word recognition; in the case of bigrams this would mean to estimate the probability of a word given the preceding word and its prosodic label. Preliminary analysis showed that this might result in an improved language modeling for ASR. Note that for this approach a considerably higher amount of training data is needed.

With regard to the use of prosody in ASU it might be useful to develop models for entire phrases instead of classifying single word boundaries, because the different prosodic attributes within a phrase influence each other. In [Jen94] different classes of intonation contour shapes of phrases have been modeled using HMMs. A continuous speech recognizer has been built similar to the ones used for word recognition. This model however does not incorporate any durational or energy features.

4.3.4 Other fields of ASU

In order to integrate prosody in ASU reliable classification of prosodic feature vectors is an important prerequisite. The features sets and classifiers developed by Andreas Kießling [Kie97] are used in the research presented in this book. [Kie97] also gives an overview of related work. A valuable contribution to the state–of–the–art has been done by the group of Mari Ostendorf [Wig92a, Wig92b, Wig94].

The voice source signal is a superposition of F0 and its harmonics. The vocal tract can be viewed as a filter giving the voice source signal its phone characteristics. For the purposes of pitch or laryngealization detection, which are voice source phenomena, it can be useful to inversely filter the speech signal in order to obtain an approximation of the original voice source signal. Different kinds of linear inverse filter models have been proposed in the literature [Mar72, Cum90, Alk92b].

To train statistical prosody models manually transcribed data are needed. Most research relies on labels created with the ToBI system [Sil92a]. It is based on the theory of tone sequences established by J. Pierrehumbert [Pie80], cf. also [Pie90]. The labels are combinations of high and low tones attached with symbols differentiating between boundary and accent tones as well as a hierarchy of break indices. An utterance is transcribed on the basis of perceptual evaluation, visual inspection of F0 contour and speech signal and by applying phonological and linguistic knowledge to the word sequence underlying the utterance. This system considers duration only implicitly as a factor contributing to phrasing but not to accentuation. A similar labeling system has been proposed by [Hir94b]. It has also been adapted to German by [Rey94]; cf. also [Rey93, Rey95a] and Sections 5.2.2, 5.2.3.

In [Pri91, Wig92b] a scheme for the labeling of purely syntactic boundaries with the purpose of training prosodic classifiers is described. For six types of syntactic boundaries different labels, called *break indices* were defined. Different sentence internal boundaries are distinguished, for example parentheses. Boundaries between two main clauses are not considered. This scheme can be used to label written language manually. It has been applied to a small corpus consisting of 35 pairs of syntactically ambiguous sentences.

4.4 Summary

The term *prosody* comprises speech attributes which are not bound to phone segments. We distinguish between *basic* and *compound* prosodic attributes. Basic prosodic attributes are *loudness, pitch, voice quality, duration, speaking rate* and *pause*. Variations of these over time constitute the compound prosodic attributes, which are *intonation, accentuation, prosodic phrases, rhythm,* and *hesitation*.

Intonation is the distinctive usage of pitch: general patterns like falling, rising, high or low pitch contribute to the meaning of the utterance. These can be used to directly determine the sentence mood, or they can among other attributes contribute to the marking of accents and phrase boundaries. In the case of sentence mood we distinguish between fall, rise, and continuation–rise. Intonation is important to indicate sentence mood in the absence of syntactic markers as is the case for elliptic clauses. A fall marks a statement, a question usually is determined by a rising contour, and at boundaries between main clause and subordinate clause a continuation–rise is frequent. The latter is also used to hold the floor, that is, especially before pauses it signals “I am not finished yet”.

Accentuation refers to syllables in an utterance, which are more prominent than others. This can be expressed by a change in duration or intonation or by an increase in loudness. An accent can be realized by all of these attributes at the same time or by a sub-set Depending on the context, the speaker, and the strength of the accent. As for the strength of accents there is a continuum. Often three categories are distinguished: phrase (primary) accent, secondary accent, and emphatic accent. For the strength of accentuation no absolute measurement can be defined, but different accented syllables within an utterance or a dialog have to be compared. The default accent position within a phrase is the last content word. This supports the structuring of utterances into prosodic phrases, cf. below. Accentuation differing from the default in position or strength has various functions: the most important one is the marking of the focus, which is the new information in an utterance. Furthermore, the scope of quantifiers can be identified by accentuation. On the discourse level the accentuation of particles can determine their function, which can be discourse particle on the one hand and either modal particle or adverbial on the other hand. Furthermore, extraordinary strong accent can be used to put a word or a syllable in contrast to some given information.

Utterances are segmented into phrases by the prosodic marking of their boundaries. For this book we will distinguish between prosodic clause and prosodic constituent boundaries. Prosodic clause boundaries usually are marked by intonation and by phrase final lengthening and optionally by pauses. They mark major boundaries. Prosodic constituent boundaries are minor boundaries usually solely marked by intonation. There is a strong correspondence between prosodic and syntactic clause boundaries; this holds also for constituent boundaries but to a lesser extent. Prosodic boundaries in the first place support the intelligibility of utterances, because they structure the utterance into meaningful segments. Furthermore, they can disambiguate between different readings of a turn.

Hesitation is a phenomenon expressed by extraordinary lengthening and/or by filled pauses. It is used by a speaker to signal a conflict between speech planning and speech production; therefore it prohibits the turn taking of a dialog partner. Such conflicts can also result in repetitions, corrections or even restarts. These phenomena are marked by prosodic means, especially by intonation.

With respect to the use of prosody in ASU it is important to distinguish between form and function of the different attributes: the same compound attribute can be expressed by different basic attributes or by different contours of the same attribute. On the contrary, the same form of a basic attribute can mark different compound attributes. The actual marking depends on the context and on the speaker. For the context it is important to note that the different compound attributes influence each other. For example an accent close to a phrase boundary may be differently marked than an accent in the middle of a phrase.

We conclude that the modeling of prosodic phenomena cannot be solved by deriving rules from a small amount of sample data, but statistical models are needed, whose parameters are trained on large amounts of data. Therefore, we developed methods to annotate large text corpora with prosodic–syntactic labels. These and the corpora used in the experiments presented in this book will be described in the next chapter.

Chapter 5

Prosodic Labeling and Speech Corpora

For the training and evaluation of stochastic prosodic models a large amount of sample data is required. Data here means a collection of speech signals which are transliterated, that is, the spoken words are given for each turn and furthermore the transliteration has to be annotated with prosodic labels. Note that for the experiments described in this book no manual time alignment of prosodic labels is required. This chapter describes the speech corpora and the prosodic labeling systems used within the research presented in the remainder of this book. At the beginning of this research only ERBA, a large data base of read speech, was available. This was used for initial studies. Later on we switched to the VERBMOBIL spontaneous speech corpus for the experiments. A few experiments were also conducted on read elliptic time of day expressions. As for the prosodic annotation, new schemes for the prosodic–syntactic labeling of large text corpora have been developed in course of our research. This chapter focuses on the description of these labels and an evaluation with respect to perceptive labels. Note that in this book we distinguish between

- *syntactic labels*, which are based on a (manual) syntactic analysis of text corpora,

- *prosodic–syntactic labels*, which are syntactic labels subcategorized according to expectations about the prosodic marking, and

- *prosodic labels*, which are based on the perceptual evaluation of the speech and/or the visual inspection of the F0 contour. They should be independent from the syntactic structure of an utterance.

Most important for the training of classifiers used in the VERBMOBIL system are the acoustic–prosodic B3 labels, cf. Section 5.2.2, and the prosodic–syntactic M3 labels developed for spontaneous speech, cf. Section 5.2.5. The labels based on the tone sequence scheme (Section 5.2.3) are only given for completeness and are not further used in this book. Several subclasses of M3 are defined in Section 5.2.5; these are also not further used in this book. However, because this is a new labeling scheme, we described also these labels. Furthermore, the fine labels help to understand the labeling scheme in general. The success of the M3 labels becomes obvious by the comparison with the acoustic–prosodic and the syntactic labels, cf. Section 5.2.7, and in particular by the results obtained with classifiers trained on the basis of these labels, cf. Sections 6.4.3 and 7.2, which eventually were integrated in the VERBMOBIL system, cf. Section 8.3.2.

The idea for the M3 labels resulted from the prosodic–syntactic phrase boundaries for the read speech corpus ERBA which were successfully used to train classifiers, cf. Section 6.4.3 and 7.1. These labels are described in the following section.

This chapter also describes different acoustic–prosodic accent labels for ERBA and VERBMOBIL. However, so far in the classification experiments presented in Sections 6.4.3 and 7.1 only accented and unaccented syllables or words were distinguished. In the VERBMOBIL system so far accents play a less important role than prosodic phrase boundaries.

5.1 ERBA

5.1.1 The Speech Corpus

A few years ago there was a need for a large domain–dependent training database for ASR within the BMFT[1] funded project ASL[2], within the European SUNDIAL project and for the parameter training of the first prototype of the EVAR system (cf. Section 3.3). A spontaneous speech database is difficult to obtain. Either one needs a fully operational prototype system or one has to perform time consuming Wizard of Oz experiments as has been described for example in [Hit89, Kra90, Fra91, Pol91, Cor93, Haa95a]. Therefore, the ERBA corpus (Erlanger Bahn Anfragen — Erlangen Train–Table Inquiries) was developed. The goal behind it was to achieve a corpus of many different domain–dependent sentences, so that a high degree

[1] At that time: Bundesministerium für Forschung und Technologie (Federal Ministry for Research and Technology).

[2] Architekturen von Systemen zur integrierten Analyse von Sprachlauten und Sprachstrukturen (Architectures of Systems for the integrated Analysis of Speech and Language).

Wann muß ich in Zeil abfahren, um um ein Uhr in Frankfurt anzukommen?
When should I leave Zeil, to arrive at one in Frankfurt?

Um gegen acht in Ed zu sein, wann muß da ich losfahren?
To be before eight in Ed, when do I have to leave?

Guten Morgen, ich möchte zwischen drei und acht Uhr nach Finsterwalde fahren.
Good morning, I want to go to Finsterwalde between three and eight.

Ich brauche eine Auskunft über eine Direktverbindung von Aachen nach Hamm über Hamburg und zwar am Montag, den siebenundzwanzigsten Mai, um ein Uhr fünfundzwanzig.
I need some information about a direct train connection from Aachen to Hamm via Hamburg on Monday, the twenty seventh of May, at one twenty five.

Wann fährt der letzte Zug nach Dortmund?
When goes the last train to Dortmund?

Wie lange dauert die Fahrt nach Riebnitz–Damgarten West?
How long does it take to Riebnitz–Damgarten West?

Figure 5.1: Sentences from the ERBA corpus.

of phonetic variability can be achieved. The sentences were read by semi–naive speakers. Based on this, the first prototype was then used to collect spontaneous speech material, which thereafter helped to improve the system [Eck96][3].

The ERBA text corpus contains sentences from the domain of train time table inquiries. It was generated with a manually defined stochastic context–free grammar, which consists of 38 sentence templates being directly derived from the start symbol S and 833 further production rules excluding the production having a train station name on the right–hand side [Rie93, Rie94]. The templates specify the coarse structure of the sentences to be generated. Figure 5.1 gives examples of sentences generated with the ERBA grammar, Figure 5.2 shows an example sentence template and a few production rules. With this subset of the grammar the first sentence of Figure 5.1 can be derived.

Each production rule is associated with a probability, which controls the fre-

[3] This approach of the iterative improvement of dialog systems starting with an automatically generated domain–dependent corpus was afterwards adopted for the design of dialog systems in different languages and domains within the research project SQEL, funded by the European Commission and managed by the Lehrstuhl für Mustererkennung, Universität Erlangen–Nürnberg (cf. for example [Ipš95]).

production rule				*P*
S ⟶ WHEN muß ich [spätestens]0.03 [in STAT]0.5 LEAVE , ARRIVE ? WHEN *should I* [*at the latest*]0.03 LEAVE [STAT]0.5 , ARRIVE ?				5/170
WHEN	⟶	wann [genau]0.005	*when [exactly]*0.005	4/6
WHEN	⟶	um wieviel Uhr	*at what hour*	1/6
WHEN	⟶	um welche Uhrzeit	*at what time*	1/6
STAT	⟶	S_EXPRESS		1/2
STAT	⟶	S_RAPID		1/2
S_EXPRESS	⟶	Frankfurt		1/154
S_RAPID	⟶	Zeil		1/417
LEAVE	⟶	abfahren	*to leave*	1/2
ARRIVE	⟶	um T in STAT anzukommen	*to arrive* T *in* STAT	1/8
ARRIVE	⟶	damit ich T in STAT bin	*that I'll be* T *in* STAT	1/8
T	⟶	um NA Uhr [MIN]0.2	*at* NA [MIN]0.2	3/18
T	⟶	zwischen NA und NA Uhr	*between* NA *and* NA	2/18
T	⟶	um QUART NB	*at* QUART NB	3/18
NA	⟶	ein	*one*	1/12
NB	⟶	eins	*one*	1/12
QUART	⟶	halb	*30 to*	1/5
WANN	⟶	um wieviel Uhr	*at what time*	1/6

Figure 5.2: Excerpt of the ERBA grammar together with English translations. Non–terminal symbols are written with capitals only; *P* denotes the probabilities of the production rules.

quency with which it is considered during sentence generation. These probabilities were as well as the grammar rules defined manually. For example, the non–terminal symbol S_EXPRESS meaning "InterCity train station" is expanded to the city name Frankfurt with a probability of 1/154; at that time there existed 154 InterCity stations in Germany. To keep the grammar specification compact, optional parts denoted with "[.]P" are introduced; these parts are expanded in $P \cdot 100\,\%$ of the cases where this specific rule is applied. Furthermore, the grammar was only designed as to generate syntactically correct sentences, which not necessarily had to be meaningful. Examples for semantically or pragmatically not meaningful sentences are given in Figure 5.3; in the third the route is quite unrealistic. These meaningless sentences could influence the way the speakers pronounce them; this was not regarded to be a problem for the training of word recognizers, what ERBA was initially intended for.

ab etwa vier Uhr bis vier Uhr
from about four o'clock until four o'clock

von Frankfurt nach Frankfurt
from Frankfurt nach Frankfurt

von Stuttgart über Hamburg nach München
from Stuttgart via Hamburg to München

am einunddreißigsten Dezember um Mitternacht
at the thirty first of December at midnight

in drei Wochen ... ankomme
within three week ... arrive

zwischen sechs Uhr morgens und neun Uhr abends ... abfahren
to leave between six a.m. and nine p.m.

Figure 5.3: Semantically meaningless parts of ERBA sentences.

The sentences consisted of a single main clause in some cases preceded by an elliptic clause and preceded or succeeded by a subordinate clause. Turns consisting of multiple sentences were not defined within the grammar. For an introduction into stochastic formal grammars (as context–free grammars) cf. [Nie83, p. 273].

With this grammar an arbitrary number of sentences can be generated. 10,000 of them were used for speech recording. To give an impression of the complexity of these sentences, Table 5.1 summarizes a few numbers: the vocabulary size is 1,517, including 571 names of train stations; the length of the sentences varies between 4 and 26 at an average of 12 words. The test set perplexity of the ERBA_TEST sub–corpus is 10.1 for a polygram model where we limited the size of n-grams to be modeled to $n = 5$. The test–set perplexities, cf. equation (2.59), were determined using 6,400 ERBA_TRAIN sentences for training and 500 ERBA_TRAIN sentences for cross validation; all names of train stations, months, days of the weeks, and numbers were represented by a category, respectively. For a description of these sub–corpora cf. below. Another interesting figure is that if all names of train stations, all numbers, all names of months, and all days of the week are replaced by a separate but unique symbol, there are still 8,648 sentences pair–wise different.

The corpus was read by 100 *semi–naive* speakers, who had no formal speaker training. This yielded 14.7 hours of speech data. Each speaker read 100 sentences. They were recorded at Daimler Benz (Ulm), Philips (Aachen), Universität Bielefeld, and Universität Erlangen–Nürnberg. With these different recording sites we

WORDS IN THE CORPUS			
# word types	# word tokens	# station types	av. sec per word
1,517	127,243	571	0.44

PERPLEXITIES			
bigram	trigram	4–gram	5–gram
16.6	10.8	10.3	10.1

SENTENCE BASED FIGURES			
	minimum	maximum	average
# words	4	26	12
duration (secs)	2.3	12.4	5.3

Table 5.1: Some figures characterizing the ERBA corpus.

could assume that the speakers came from different dialectal regions. The sentences presented to the speakers included punctuation as shown in Figure 5.1. The recording was stopped if the speakers left a pause of more than 500 msecs, that is, they were forced to leave only short pauses. After the recording of one sentence the speakers had to listen to the recorded signal, and were asked to repeat the recording if it differed from the presented sentence. The recording was done in normal office environment with a close talking microphone connected to a Desklab A/D device from Gradient. The speech signals were sampled with 16 kHz and linearly quantized with 14 bits.

The ERBA speech corpus was divided into different subsets according to Table 5.2. ERBA_TRAIN was used for the training of prosodic classifiers; ERBA_TEST was used for their evaluation. ERBA_LISTEN served for listener judgments concerning prosodic accents and boundaries and was used for cross–validation of the automatically generated prosodic labels described in Section 5.1.4. In order to get realistic listener judgments the semantically not meaningful sentences had to be excluded. To include the same number of sentences from each speaker, 50 sentences per speaker were used. ERBA_INF was used for preliminary experiments on the use of prosodic clause boundaries in a parser (cf. Section 8.3.2). It is the subset of ERBA which contains all sentences with infinitive clauses, which could be parsed by the parser, and for which word graphs could be made available by Daimler Benz, Ulm. Note that 182 sentences out of ERBA_INF are contained also in ERBA_TRAIN. The sub–corpora ERBA_TRAIN, ERBA_TEST and ERBA_LISTEN were spoken by different speakers.

	ERBA_TRAIN	ERBA_TEST	ERBA_LISTEN	ERBA_INF
# female speakers	25	9	5	34
# male speakers	44	12	5	49
# sentences	6,900	2,100	500	242
# word tokens	80,919	24,483	6,792	3,098
minutes of speech	380	117	49	22

Table 5.2: ERBA sub–corpora

5.1.2 Automatic Labeling of Syntactic–prosodic Phrase Boundaries

For the training of statistical classifiers large amounts of labeled training data are needed. Prosodic labeling on the basis of perception tests is very time consuming. Furthermore, they reflect not exactly what is needed during a syntactic analysis of speech. Therefore, we developed a scheme for the automatic labeling of the ERBA corpus. We already published a summary of this scheme in [Bat93b, Bat95b]. A detailed description will be given in the following. Furthermore, a new interpretation of the comparison between these labels and listener judgments is given in Section 5.1.4. This comparison and the good recognition rates presented in Section 6.4.3 justify our approach.

Syntactic clause and constituent boundaries were marked in the ERBA grammar and included in the sentence generation process. The majority of the syntactic clause boundaries are expected to be prosodically marked, cf. the discussion in Sections 4.1.2 and 4.2.3. Regarding constituent boundaries it depends very much on the position within a sentence whether they will be prosodically marked or not. Therefore, these are further subdivided so that we distinguish four types of boundaries; Table 5.3 gives an overview.

As for B3 three different types of boundaries can be distinguished: First, there are boundaries between elliptic clause and clause, for example,

guten Morgen B3 ich möchte gerne ... (5.1)
good morning B3 *I would like ...*

The second type are boundaries between main and subordinate clause, for example,

... einen Zug B3 der sehr früh fährt (5.2)
... a train B3 *that leaves very early*

B3	mark clause boundaries
B2	are constituent boundaries likely to be marked prosodically
B1	are boundaries that syntactically belong to the normal constituent boundaries as B2 but that are most certainly not marked prosodically because they are "close" to a B3 boundary
B0	every other word boundary

Table 5.3: Syntactic–prosodic boundary labels in ERBA.

Finally, B3 occur at coordinating particles between clauses:

ich möchte um acht Uhr nach München fahren B3 und möglichst früh ankommen (5.3)
(I would–like at eight o'clock to Munich to–go B3 *and possible as–early–as arrive)*
I would like to go to Munich at eight o'clock B3 *and I want to arrive as early as possible*

The beginning and end of a sentence are implicitly labeled as B3.

In the grammar we marked all the types of constituent boundaries with B2 which in the generated sentences occur in some context where they could be marked prosodically. Examples (5.4) and (5.5) show examples of such boundaries.

in der Nacht B2 mit dem InterCity B2 nach Ulm (5.4)
during the night B2 *with the InterCity* B2 *to Ulm*

zwischen Ulm B2 und Stuttgart (5.5)
between Ulm B2 *and Stuttgart*

Certain boundaries belonging to these syntactic class are not prosodically marked depending on their position in the sentence. Specifically, if they are "close" to a B3 boundary, they most probably will not be marked for rhythmic reasons. The text corpus together with the B3 and B2 was generated. A postprocessor with a few simple rules was then applied to convert those constituent boundaries which were close to a B3 boundary from B2 into B1. The definition of "close" cannot be defined in terms of a fixed number of words, but it depends on the type of the phrase being between the B3 and the constituent boundary. We applied the following rules:

- For the first constituent in a clause: if there are only function words, auxiliary verbs or modal verbs contained in the constituent the B2 boundary is turned into B1.

- For the final constituent in a clause: B2 boundaries before a final verb or a final verb / auxiliary verb combination are turned into B1.
- No other B2 was converted into B1.

The following ERBA sentence shows examples for B1 labels:

wann muß ich B1 in Bielefeld B1 abfahren B3 um B1 ab etwa sechs Uhr B2 bis zwei Uhr B2 in Wolgast B1 zu sein (5.6)
(*when have I* B1 *in Bielefeld* B1 *to–leave* B3 *in–order* B1 *from about six o'clock* B2 *until two o'clock* B2 *in Wolgast* B1 *to be)*
when do I have to leave Bielefeld to be in Wolgast between six o'clock and two o'clock

At a B1 boundary we hypothesize a prosodically clitic, weak constituent that integrates with the succeeding or preceding stronger constituent into a greater prosodic phrase. Note that B1 can also succeed each other as in example (5.6): The first B1 is close to the beginning of the sentence and the preceding constituent consists of a function word (pronoun) and an auxiliary verb. The second B1 is close to the B3 boundary separating main and subordinate clause. The third B1 in the sentence marks again a constituent boundary being close to the beginning of a sentence. Examples (5.7) and (5.8) show that there can also B2 boundaries be the closest boundaries to B3. They were not converted to B1, because the rules given above could not be applied. In fact, these boundaries are likely to be prosodically marked.

hat B1 der Sonderzug B2 um drei Uhr B2 nach Northeim B2 einen Liegewagen (5.7)
(*has* B1 *the special–train* B2 *at three o'clock* B2 *to Northeim* B2 *a sleeping–car)*
does the special train at three o'clock to Northeim have a sleeping car

bitte geben Sie uns B2 die Zeit B2 des letztmöglichen Zugs B2 zwischen Prora B2 und Fürth (5.8)
(*please give you us* B2 *the time* B2 *of-the last–possible train* B2 *between Prora* B2 *and Fürth)*
can you give us please the departure time of the latest train from Prora to Fürth

Table 5.4 shows the frequencies with which the different boundary labels occur in the ERBA sub–corpora. After these labels were generated, we compared them

	ERBA_TRAIN	ERBA_TEST	ERBA_LISTEN	ERBA_INF
# B3	2,673	788	396	244
# B2	15,595	4,700	1,173	357
# B1	13,088	3,940	1,216	618
# B0	49,563	15,055	3,553	1,635

Table 5.4: Frequency of boundary labels for the different ERBA sub–corpora; sentence–final boundaries are not considered.

with listener judgments for a small corpus in order to justify our labeling scheme. Before we come to this in Section 5.1.4 we will describe the automatic generation of accent labels in the following section. Finally, it should be noted that the text read by the speakers did not contain the B2 boundary markers. However, it included the B3 boundaries in the form of commas.

5.1.3 Automatic Labeling of Syntactic–prosodic Accents

As in the case of boundaries there is a need for large quantities of data labeled with accents. We again solved this problem by a labeling scheme, which can be applied automatically. The scheme consists of a set of rules which assign accent labels to syllables based on the word chain in which the boundary labels described in Section 5.1.2 are already inserted. We already published a summary of this scheme in [Kie94a]. A detailed description will be given in the following. Furthermore, a revised interpretation of the comparison between these labels and listener judgments is given in Section 5.1.4. This comparison and the good recognition rates presented in Section 6.4.3 justify our approach. So far the classifier only distinguishes two classes, accented and unaccented syllables.

For the assignment of accents it has to be decided, which words in a sentence are accented, and which syllables in a word carry the accent. In words with more than one syllable, normally one of these syllables bears the word accent, exceptions will be discussed below. Factors that influence whether or not a word is accented include the word category, for example content word versus function word, its position in a larger prosodic context, and the speaking rate. Rhythmic constraints can influence the location of accents within words as described in Section 4.1.2. In our approach we only consider the word category and the position of a word within a phrase. Furthermore, since we deal with sentences read out of context we do not have to care about contrastive or emphatic accents and we can expect that accents are placed in default position, cf. Section 4.1.2.

Assigning the Accent Label to a Word within a Phrase

The first step is to decide which words within a phrase are accented. We assume that in each prosodic phrase bounded by B1, B2, or B3 one and only one word is more prominent than the others. Note that for the generation of the accent labels also the beginning and the end of a sentence is assumed to be a B3 boundary. In German, the phrase accent is normally positioned according to the rightmost principle as described in Section 4.1.2. In the following example the syllables to be expected as accented are underlined:

> ich möchte B1 am nächsten Dienstag B2 zwischen drei B2 und sechs Uhr B2 von Hamburg B2 nach Ulm B1 fahren (5.9)
>
> (*I would like* B1 *next Tuesday* B2 *between three* B2 *and six o'clock* B2 *from Hamburg* B2 *to Ulm* B1 *to go)*
>
> *I would like to go next Tuesday between three and six o'clock from Hamburg to Ulm*

This example contains the two most important exceptions in our database: The word Uhr (*o'clock*) and verbs such as fahren (*to go*) are usually not accented though formally being content words, because they are rather predictable in the domain of train table inquiries and therefore semantically weak or clitic. Therefore, in the last two phrases not the verb fahren but the city name Ulm is expected to be accented. This is the case for all verbs, except for the so called particle verbs ankommen (*arrive*) and abfahren (*leave*) that might be accented. In general in German it seems to be often the case that in a verb phrase the right most argument of the verb is accented [Uhm91]. Interrogative pronouns such as was (*what*), wann (*when*) are function words, which obviously are semantically and pragmatically strong words in this domain; these are expected to be accented. Based on these observations the following rules were formulated:

> For each constituent bounded to the right by symbol Bx and to the left by By (x,y ∈ {1,2,3}) look for
>
> 1. the rightmost content word, which is not a function word, a verb, an auxiliary verb, an interrogative pronoun or the word Uhr, or
> 2. if not found, look for the rightmost verb, or
> 3. if not found, look for the word Uhr, or
> 4. if not found, look for an interrogative pronoun, or
> 5. if not found, look for an auxiliary verb, or
> 6. if not found, take the rightmost word, regardless its category
>
> and mark this word by symbol Ax, where x is identical to the x in Bx.

After applying this rule to a sentence, in each phrase one and only one word is marked by an accent label Ax ($x \in \{1,2,3\}$). All other words are implicitly marked by A0. With this, accent labels are achieved which are distinguished by the type of phrase; for example, an A2 label marks the accented word in a prosodic constituent, that is, a phrase bounded to the right by B2, and A3 is used for the accented word in a prosodic clause. The strength of the accentuation is expected to decrease from A3 to A0. In order to take into account that there are semantically weak words occurring in short phrases before a B3 boundary, we have to add another rule:

IF		the word labeled with A3 is not a content word
	AND	the phrase is bounded on the left by B1
	AND	it is bounded on the right by B3
	AND	there is a content word on the left hand side of the B1 boundary
THEN		exchange the accent labels of these two words.

This rule changes example (5.10) into (5.11); the accent label has been placed before the accented word:

...nach (A1)Ulm B1 (A3)fahren B3 (5.10)
...*to* (A1)*Ulm* B1 (A3)*to go* B3

...nach (A3)Ulm B1 (A1)fahren B3 (5.11)
...*to* (A3)*Ulm* B1 (A1)*to go* B3

Applying the rules to the sentence from example (5.9) results in the following labeling:

ich (A1)möchte B1 am nächsten (A2)Dienstag B2 zwischen (A2)drei B2 und (A2)sechs Uhr B2 von (A2)Hamburg B2 nach (A3)Ulm B1 (A1)fahren (5.12)

Special treatment is necessary for certain compound words that occur very frequently in our application, for example, city names. In our lexicon these words are characterized by a linking hyphen or dash. Following the rightmost principle, we marked the rightmost word by Ax, and all other words of the compound word by Axi, denoting that there is an *implication* from left to right, that is, if any word of the compound word is accented, all its right hand neighbors are accented as well, cf. examples (5.13) and (5.14). It has to be noted that this rule is rather straightforward and does not take into account other possibly relevant factors as, for example, rhythmic constraints, which in certain prosodic contexts might cause one of the Axi syllables to be the only accented syllable in the word.

Assigning the Accent Label to a Syllable within a Word

When a word is marked as accented it has to be decided, which syllable(s) carry the accent. This is based on the labeling of the lexical accent in our lexicon. These labels were created manually; we decided in favor of a rather broad labeling, that is, we only distinguished accented from unaccented syllables. Secondary accents are not labeled because in a canonical citation pronunciation, these differences might be produced and perceived systematically but not in a more casual pronunciation as is the case in continuous speech.

The actual accent assignment depends on the following factors:

1. If the word has only one syllable marked as the (lexical) word accent in the lexicon, this syllable inherits the symbol Ax from the word.

2. For some words different alternative positions for the lexical accent exist due to regional differences. All alternatives are marked in our lexicon. In the automatic labeling, all these syllables get the symbol Axa, denoting that they are real alternatives, and that it is to the discretion of the speaker, which of those alternatives is actually accented.

3. If there is no lexical accent at all for this word, which is usually the case for function words, the first syllable in the word gets the symbol Axn, so that these labels can be separately investigated. This simple rule can of course not be applied to all German function words but it works reasonably well within our lexicon.

These rules apply in the same way to single words and to the parts of the compound word marked by implication labels as shown in the following two examples, where (5.13) contains a conventionalized phrase used as a greeting and (5.14) is the name of a train station:

Grüß_Gott (5.13)
Axi Ax

Riebnitz–Damgarten–West (5.14)
Axi A0 Axi A0 A0 Ax

Examples for alternative accents are numbers as

zweiundzwanzig (5.15)
Axa A0 Axa A0
(two and twenty)
twenty two

	ERBA_TRAIN	ERBA_TEST	ERBA_LISTEN	ERBA_INF
# A3	11,881	3,628	1,035	566
# A2	17,987	5,460	1,336	385
# A1	13,089	3,940	1,216	618
# A0	94,226	28,860	7,853	3,538

Table 5.5: Frequency of syllable–based accent labels for the different ERBA sub–corpora.

and certain city names as Erlangen, where either the first or second syllable can be accented.

By applying all these rules to the whole ERBA database of 10,000 sentences in total 199,078 syllables were marked. Table 5.5 shows their frequency of occurrance in the different ERBA sub–corpora.

The accent and boundary labels described so far are based on expectations about the prosodic marking. We developed labeling schemes which allowed to generate large amounts of labeled data with rather little effort. The following section will tell us how these labels relate to listener judgments.

5.1.4 Automatic Labels versus Listener Judgments

In order to verify our expectations concerning the prosodic marking, perception experiments with "naive" listeners were conducted at the L.M. Universität München. These were mostly students having no special prosodic or linguistic education. The listeners were given the sentences of the ERBA_LISTEN corpus in orthographic form without any punctuation marks and without any of the prosodic–syntactic labels. Then they could listen to each of the sentences three times. Ten subjects were asked to mark the space between two words if they felt it separated two different "chunks" of speech. The listeners were instructed not to rely upon their knowledge of sentence structure, although its influence cannot be ruled out altogether. A second group of ten subjects was asked to mark each syllable they perceived as accented. So, each word boundary and each syllable got a perception score from 0 (no mark) up to 10 (all 10 subjects in the test perceived is prosodically marked accent or a phrase boundary).

Boundaries

In Figure 5.4, the results of the perception experiments are given for the four different boundary types. The distribution of the B0, B1, and B3 boundaries meet

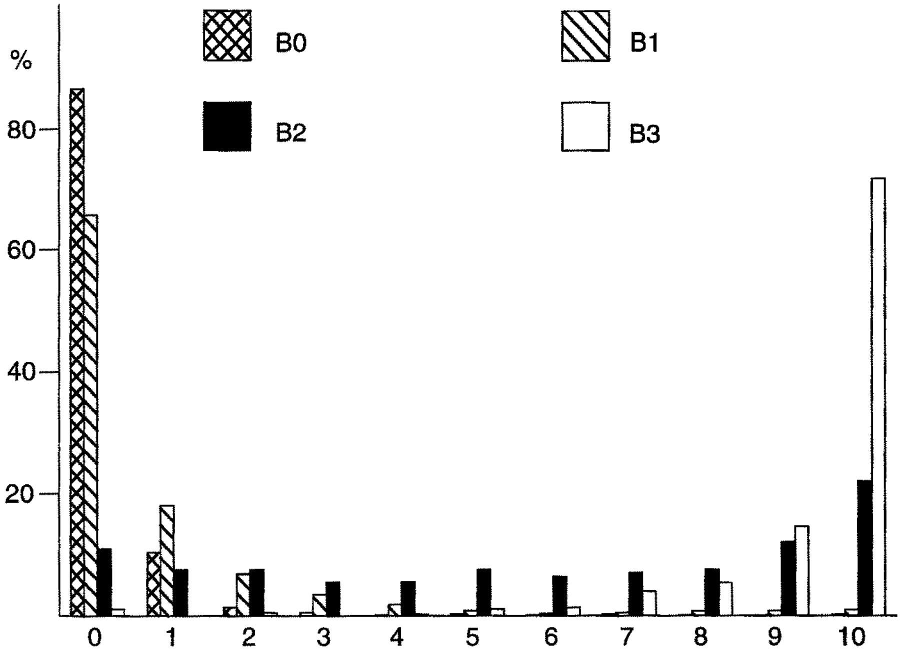


Figure 5.4: Perception result of ERBA boundaries: frequency (in %) of the scores for the different boundary classes.

our expectations and cluster at the left end (very few scores for B0 and B1 boundaries) or at the right end (many scores for B3 boundaries). The B2 boundaries behave differently: only 63% were marked by the majority of the subjects. One explanation might be that it is to the discretion of the speaker if he wants to mark these boundaries. Another explanation might be that the B2 boundaries are just less clearly marked prosodically. This would coincide with the fact that B2 are syntactically weaker boundaries than B3. Since most of the scores of 2 or higher coincide with B2 or B3 and vice versa most of the scores of 0 or 1 coincide with B0 or B1, it makes sense to put the borderline between prosodically marked and unmarked boundaries between the scores 1 and 2. With this borderline, in 85% of the cases where at least two listeners perceived a boundary there was an automatically generated reference boundary (B2, B3). On the other hand, in 94% of the cases where less than two listeners perceived a boundary we automatically generated a no–boundary reference label (B0, B1).

This leads us to the conclusion that the automatically generated reference boundaries are adequate and can be used to train and test acoustic–prosodic classifiers. In fact, the recognition rates presented in Section 6.4 are better than we expected for this material, because our impression was that many people were

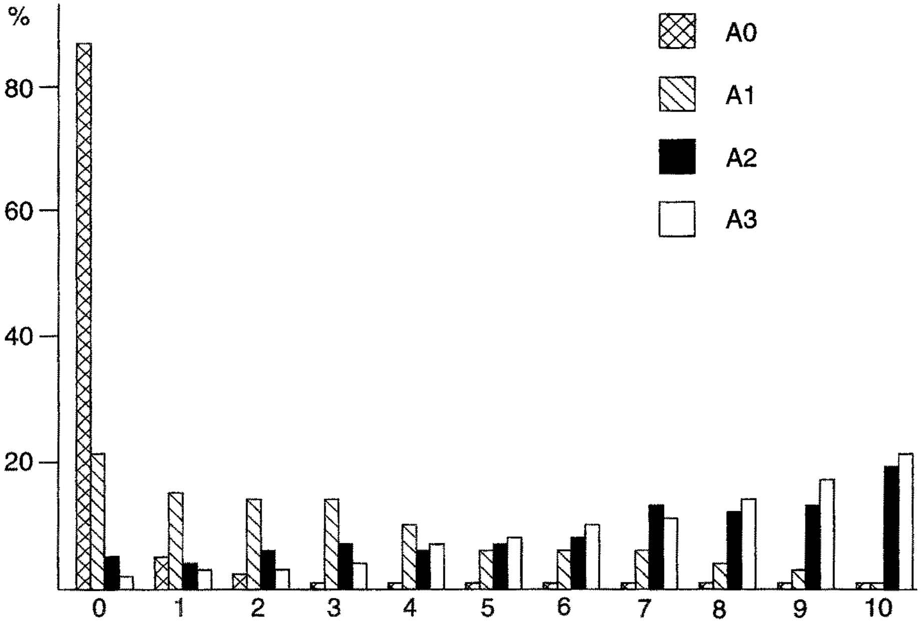


Figure 5.5: Perception result of ERBA accents: frequency (in %) of the scores for the different accent classes.

reading rather monotonously.

Accents

The perception data were compared with the automatically labeled places of phrase accents. The 500 sentences contain 71 types of function words with 3,346 tokens and 588 types of content words with 3,396 tokens. Function words got an average score of 1.4 with a minimum of 0 and a maximum of 8; 10% were above 5 and 36% above the mean. Content words got an average score of 7.4 with a minimum of 0 and a maximum of 10; 14% were less than 5 and 47% were less than the mean.

In Figure 5.5 the frequencies of the perceptual scores are plotted for the different accent classes. In accordance with the prosodic marking of the boundaries most of the A0 and A1 labels got low scores, whereas the majority of the A2 and A3 labels got high scores. This result meets our expectation. Since most of the scores of five or higher coincide with A2 or A3 and vice versa most of the scores of

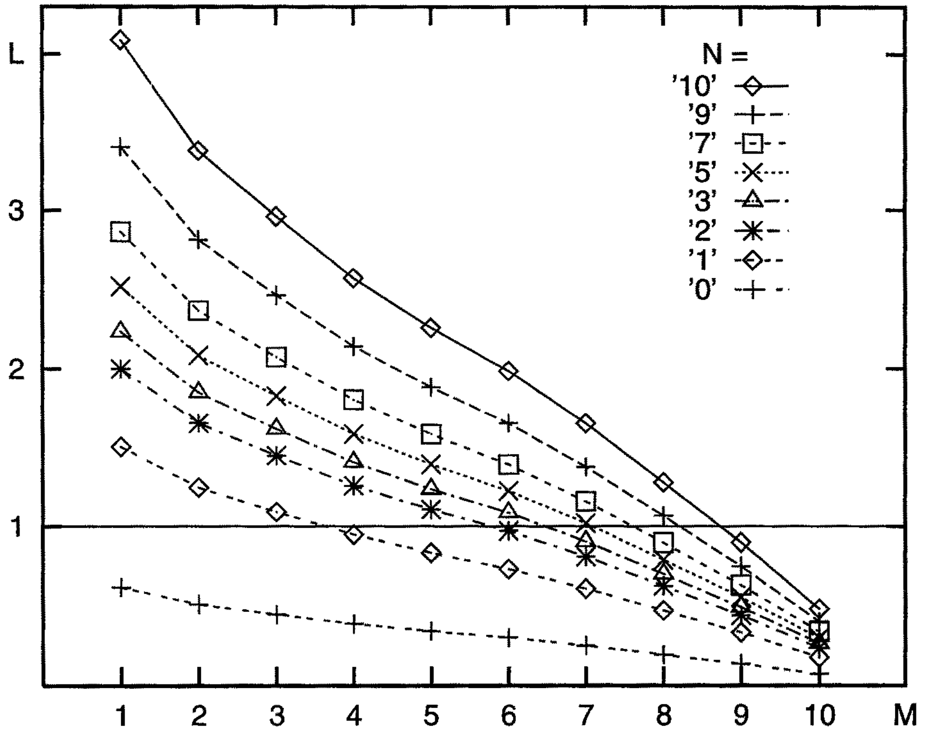


Figure 5.6: The relationship between perceived accent and boundary scores.

four or less coincide with A0 or A1, it makes sense to put the borderline between accented and unaccented syllables between the scores four and five. However, the behavior or A1, A2 and A3 at the scores 4, 5 and 6 is not clearly distinct so in fact the borderline could be anywhere around these scores.

In general the scores for the accents are not as clear–cut as for the boundaries. In contrast to B2 versus B3 the labels A2 and A3 got very similar scores: the scores of A2 cluster more to the right end than the B2 scores and A3 does not cluster as close to the right end as the B3 scores. This is in contradiction to the theoretical assumption often made in the literature that the rightmost phrase accent in a sentence is the most prominent one, cf. Section 4.1.2 and [Koh77]. Similarly to B1 versus B0 the label A1 got higher scores than A0. As mentioned above A1 in general got low scores, however, much more A1 labels got high scores than the B1 labels.

To investigate this further, we examined the relationship between perceived boundaries and accents. The result is depicted in Figure 5.6:

- The abscissa represents a threshold M, partitioning the perceived accent scores (PAS) into two classes: if $PAS \geq M$, the syllable is defined to be accented, else it is not accented.

- Each of the curves marked with $N \in [0 : 10]$ represents a threshold, partitioning the perceived boundary scores (PBS) into two classes: if $PBS \geq N$, the word boundary is defined to be a phrase boundary.
- If the variables M and N are assigned particular values the listener judgments are binarized and mapped to *accent* versus *no–accent* or to *boundary* versus *no–boundary*, respectively. Then the average number of accent labels within one phrase (L) is given at the ordinate in the figure.

In theory it is usually assumed that in each phrase one accented syllable occurs, cf. Section 4.1.2. This is represented by the horizontal line at $L = 1$ in the figure. We are now interested in the points where this line crosses the curves. It turns out that it crosses the curve which corresponds to $N = 2$ exactly at the abscissa value of $M = 6$. Therefore, if we assume the theory about the number of accents per phrase is correct, we have to draw the borderline between *boundary* and *no–boundary* between the listener scores of 1 and 2 and the borderline for *accent* versus *no–accent* is between 5 and 6. This corresponds exactly to the borderlines which we proposed above for different reasons. Note that if these borderlines were slightly moved, for example, if $N = 3$, and $M = 7$, the average number of accented syllables per phrase is still roughly at 1.

This section explained how we can create prosodic–syntactic phrase boundary and accent labels based on text corpora without conducting perceptual evaluations of utterances. The comparison with these labels with boundaries and accents marked by listeners showed a very high agreement and thus justified our approach. The main focus of the experimental work of this book is the spontaneous speech data of the VERBMOBIL project. The findings of this section motivated us to develop a similar scheme for the prosodic–syntactic labeling of spontaneous speech. This will be described in the following section.

5.2 VERBMOBIL

In this chapter we dealt so far with elicited speech, where the observed phenomena are rather controlled, that is, the sentences are grammatical, and one can assume a default prosodic marking. This allows for relatively easy creation of prosodic reference labels, cf. also the discussion in [Bat93c]. Furthermore, since the speech is more regular it is easier to train good models for speech recognition. However, if the goal is to build ASU systems which can be used in real–life environments with "naive" users the models have to be developed with spontaneous speech data. Several studies showed that the phonetic and prosodic properties of read and spontaneous speech differ strongly, and furthermore that models trained on read speech perform badly when tested on spontaneous speech [Bla91, Sil92b, Kom93a, Dal92, Dal94b, Bat95a, Cam95a]. In this section, we will investigate the VERBMOBIL corpus, which is a large spontaneous speech corpus. We will describe recording conditions, type and frequency of spontaneous speech phenomena as well as methods for the creation of prosodic reference labels, which are needed to train statistical classifiers. The different types of labels are acoustic–prosodic accent and boundary labels, syntactic boundary labels as well as prosodic–syntactic boundary labels. These are compared in Section 5.2.7. This comparison, the classification results presented in Section 6.4.3, and the improvement of the syntactic analysis in the VERBMOBIL system described in Section 8.3.2 show the success of the new prosodic–syntactic labeling scheme. Different classes of accents and boundaries are defined in this section. However, the classifiers described in the following chapters are mainly trained to detect clause boundaries or to discriminate between accented and unaccented words. These tasks were identified to be most promising with respect to an integration in the VERBMOBIL system. Tone labels are only described for completeness.

5.2.1 The Speech Corpus

For the training and testing of stochastic ASR models and for the evaluation of the knowledge based linguistic modules large spontaneous speech databases have been collected within the VERBMOBIL project by Universität Bonn, Universität Karlsruhe, Universität Kiel and L.M. Universität München [Koh94, Hes95, Til95]. The data collection is still ongoing. Meanwhile 12 CDs have been made available by the Bavarian Archive for Speech Signals (BAS). All data comprise human–human dialogs of the appointment scheduling domain of VERBMOBIL, cf. Section 3.4. Most of these dialogs and all the dialogs investigated in this book are in German. These are contained in the CDs one through five, seven and CD 12. The other dialogs are in English and few are in Japanese.

The CDs contain altogether 637 German dialogs, which consist of 12,033 turns. They comprise 307,982 word tokens and 6,573 word types. Table 5.6 gives a few further numbers. The perplexities are test–set perplexities, cf. equation (2.59), of a word category based polygram (602 categories) determined on the sub–corpus BS_TEST using 5,714 turns of the sub–corpus M_TRAIN for training and the remaining 500 turns for cross–validation; for the definition of the sub–corpora cf. below. Table 5.7 shows how many dialogs are contained on each of the CDs. The number of dialogs and the number of turns is not directly related, because sometimes the

WORDS AND NON–VERBALS IN THE CORPUS			
# types	# tokens	# word tokens	av. sec per token
6,573	307,982	263,775	0.32

PERPLEXITIES			
bigram	trigram	4–gram	5–gram
106	90	88	87

TURN BASED FIGURES			
	minimum	maximum	average
# words	1	226	26
duration (secs)	0.3	74.2	8.3

Table 5.6: Some figures characterizing the VERBMOBIL corpus.

CD	1	2	3	4	5	7	12
# dialogs	63	81	45	72	101	68	207
# turns	1840	1537	1214	1517	2154	1737	2034

Table 5.7: Overview about the currently available VERBMOBIL CDs.

speakers had the task to fix more than one appointment per dialog.

Different sub–corpora have been used in the experiments presented in this book for the training and testing of classifiers or for the evaluation of the combination of the prosody module with different linguistic modules. Table 5.8 gives an overview. The last column shows how many dialogs were taken from each of the CDs 1, 2, 3, 4, 5, 12. CDs 7 and 12 were not used for training purposes because they were just recently released. Part of CD 12 has been used for testing. The following test corpora have been used for different experiments:

- BS_TEST is usually used for the test of prosodic classifiers. It was spoken by six different speakers (three male, three female); it contains one dialog per recording site. In all other corpora about one third of the speakers are female[4].

- LING_BIG has been defined for the evaluation of the syntactic and semantic

[4] Note that the BS_TEST sub–corpus differs from the test data we used in previous publications as [Kom95b].

modules by the people working on the VERBMOBIL syntactic and semantic modules.

- LING_SMALL contains five dialogs from CD 12; it was used for the evaluation of the VERBMOBIL parsers with respect to size of word graphs. This is a subset of 268 turns of CD12 which are used for cross–validation in the 1996 word recognition experiments. All VERBMOBIL speech data except these 268 turns are used for speech recognizer training.

- The LING_TEST corpus has been chosen for final linguistic evaluation. These turns had to be taken from CD 12 so that they are real test data for the linguistic modules as well as for the prosody module. A further criterion was that the vocabulary contained in the turns should be covered by the VERBMOBIL lexicon. This was fulfilled by 1038 CD 12 turns. In alphabetical order the first 594 out of these were chosen by DFKI Kaiserlautern and Siemens München to make up the LING_TEST corpus. 116 of these turns are contained in the word recognizers cross–validation data. This is the only VERBMOBIL sub–corpus, which does not contain all turns of each of the dialogs included in the corpus. The turns are taken from 122 dialogs and 223 speakers.

- The corpus D_TEST has been defined by the people working on the VERBMOBIL dialog modules for classification experiments with hand–segmented dialog acts.

The different training corpora contain the maximum available data for the different label types described in the following sections:

- BS_TRAIN contains all turns annotated with prosodic labels except for the ones contained in BS_TEST, cf. Section 5.2.2.

- M_TRAIN are the turns labeled with prosodic–syntactic boundaries as described in Section 5.2.5. The labels are also available for BS_TEST, BS_TRAIN, and LING_BIG. These sets are not contained in M_TRAIN.

- S_TRAIN is the subset of BS_TRAIN, for which syntactic labels were created, cf. Section 5.2.4.

- D_TRAIN is labeled with dialog act boundaries as described in Section 5.2.6. It is a subset of the conjunction of BS_TRAIN an M_TRAIN.

The speech recording has been conducted in quiet but not noise–free rooms. Two subjects got the task to schedule one or more appointments for business meetings. The purpose of the meetings and the required amount of time was given. Each

	dia-	turns	min-	word	dialogs per CD					
	logs		utes	tokens	1	2	3	4	5	12
BS_TEST	3	64	11	1,513	2	1				
LING_BIG	14	315	33	4,806	8	1	2	2	1	
LING_SMALL	5	56	6	955						5
LING_TEST	122	594	62	9,482						122
D_TEST	31	453	58	8,432				31		
BS_TRAIN	30	797	96	13,145	27		1	1	1	
M_TRAIN	293	6,214	869	132,452		80	43	70	100	
S_TRAIN	21	583	66	8,222	21					
D_TRAIN	96	2,459	327	51,543		51	45			

Table 5.8: Overview about the different VERBMOBIL sub–corpora.

of the speakers had a personal calendar, where business and private appointments had already been inserted. In about one third of the dialogs the speakers had to press a button while they were speaking to prevent that they were talking at the same time. Close talking microphones were used for the recording. The signals were digitized with a stereo DAT recorder with 48kHz sampling frequency per channel. Later the speech signals were low–pass filtered and downsampled at 16 kHz. They were stored turn–wise in files.

A *turn* is defined as one chunk of speech uttered by one person without "very long" pauses and without being interrupted by a different person. A turn can contain more than one clause or clause like unit. Note that in spontaneous speech the grammatical clause as defined in grammars for written language, as for example [Eng77], cannot be applied in general, cf. [Nak92]. An orthographic transliteration of these turns has been conducted by the above mentioned institutes following the guidelines defined in [Koh94].

In [Tro94] a part of the VERBMOBIL corpus consisting of 50 dialogs, 641 turns has been analyzed with respect to such spontaneous speech phenomena. He subsumes structures as left and right dislocation, extraposition, and ellipsis under the term *free phrase*. They have in common that they are either not within the verbal brace or that they are not bound to any matrix clause at all. In our prosodic–syntactic labeling scheme as described in Section 5.2.5 free phrases were taken into account.

Since the VERBMOBIL recordings were not obtained in wizard–of–oz experiments as for example the ATIS data described in [Hit89, Kra90, Fra91, Pol91, Cor93, Haa95a] or with an ASU system as the data described in [Eck96] but were

spontaneous speech phenomenon	label	once per # of words	once per # of turns
begin of reparandum	+/	95	4.5
end of reparandum	/+	166	7.8
word fragment at end of reparandum	=/+	191	8.9
word fragment and restart	=/–	2,878	135.1
syntax fragment and restart	/–	431	20.2
hesitation lengthening at word end	...<Z>	28	1.3
hes. lengthening at word beginning	<Z>...	294	13.8
turn initial pauses, breathing	<Pause> <Atmung>	13	0.6
filled pauses	<äh>, <ähm> <hm>, <häs>	37	1.7
other human non–verbal sound	<...>	42	2.0
non–human noise	<#...>	32	1.5

Table 5.9: Disfluencies labeled in the VERBMOBIL transliteration.

obtained from unconstrained human–human dialogs, we can state without giving a detailed comparison that its *degree of spontaneity* is very high, that is, spontaneous speech phenomena are frequent: Only 13% of the turns investigated in [Tro94] consist of a single and grammatically complete sentence. At least one free phrase occurs in 71% of the turns, and on the average there are 1.5 free phrases per turn.

Other spontaneous speech phenomena are disfluencies, cf. [Bat94b] and Section 4.2.8, non–verbal articulations, as laughter, lip smack or coughing, and all kinds of noise including the slamming of doors. These agrammatical and non–speech phenomena are represented in the transliterations of the turns, where <.> denotes non–verbal or incomprehensible human sound events and <#> denotes non–human noise. Within the brackets the name of the specific events is given. Disfluencies are also labeled in the transliteration. For typical turns containing these disfluency labels cf. examples (5.16) and (5.17). Table 5.9 gives an overview of the labels found in the transliteration and shows their frequency relative to the number of words and turns, respectively. This was evaluated on the M_TRAIN corpus, which is described below. The number 4.5 in row "begin of reparandum" and column "once per # of turns" of the figure means that once every 4.5 turns there is a repair. The part of the turn being corrected is the *reparandum*. The beginning of a reparandum is always labeled with +/ whereas the end should either be marked by /+ or by =/+. *Repair* refers to interruptions, where the preceding few words (or a word fragment) are corrected by the subsequent words, which together with the words preceding the reparandum build a syntactically well formed word sequence, as da ich auch den ganzen November in (5.16). The term *restart* refers to a full interruption, after which a completely new syntactic construction, for example, a new sentence, is started. To conclude we will give two examples for typical VERBMOBIL turns including the special spontaneous speech labels:

<#Klopfen> da kann ich leider nicht <Atmung> <Schlucken> ich hätte +/ höchstens noch<Z> /+ <Pause> höchstens noch<Z> <Pause> tja <Pause> <Schmatzen> +/ den Sa =/+ den Sonntag Montag was anderes geht bei mir nicht <#> da ich auch +/ im<Z> /+ <Schmatzen> den ganzen November ab neunzehnten bis neunundzwanzigsten leider verhindert bin <#Klicken> (5.16)
<#knocking> there I unfortunately have no time <breathing> <swallowing> I had +/ only available<Z> /+ <Pause> only available<Z> <Pause> well <Pause> <smacking> +/ the Sa =/+ the Sunday Monday I don't have other dates available <#> because I have +/ in<Z> /+ <Schmatzen> all the November from the nineteenth to the twenty ninth other plans unfortunately <#click>

ja <Pause> vierzehn Uhr denn <Pause> am Freitag den zwanzigsten <Pause> +/ Herr<Z> /+ darf ich noch mal Ihren /– <Lachen> Herr<Z> <ähm> <Pause> Wieland (5.17)
yes <pause> four p.m. because <pause> on Friday the twentieth <pause> +/ Mister<Z> /+ may I [get] again your /– <laughter> Mr.<Z> <ehm> <Pause> Wieland

5.2.2 Acoustic–prosodic Boundary (B) and Accent Labels

In [Rey94], an inventory of prosodic labels intended to label the VERBMOBIL speech data along the lines of ToBI, cf. [Sil92a, Bec94], has been defined. In the following we will give an overview of the different labels. Examples for the different boundary and accent categories are also given in Figures 4.3 to 4.5.

Inspired by the labels defined for ERBA the following boundary labels were defined for VERBMOBIL:

- B3: prosodic clause boundary
- B2: prosodic phrase boundary
- B9: irregular boundary, usually hesitation lengthening

- B0: every other word boundary

A boundary is labeled as B9 if it is either marked by an extraordinary lengthening, that is, a hesitation lengthening, or if it does not correspond to any syntactic boundary. For a definition of these terms cf. Section 4.1.2, especially page 105. Unfortunately, the same names have been chosen for these labels as for the ERBA boundary labels despite the fact that the ERBA labels denote prosodic–syntactic labels whereas the VERBMOBIL labels describe perceptive phenomena. Henceforth, we will refer to these labels as the B labels.

The following prosodic accent labels were defined in [Rey94]:

- PA: The most prominent (primary) accent within a prosodic clause. There should be only one PA per phrase, but more than one can be marked if there are several words being equally prominent.
- NA marks all other accented words as carrying a secondary accent.
- EK: emphatic or contrastive accent.
- UA: unaccented words.

For a description of the different terms again cf. Section 4.1.2. The definition of the accents labels for VERBMOBIL differs much from the ERBA accent labels. This is due to the fact that VERBMOBIL is spontaneous speech. Therefore emphatic and contrastive accents may occur and in contrast to theory there can be more than one accent per phrase.

Until December 1995 the Universität Braunschweig labeled 33 dialogs from the VERBMOBIL corpus cf. also [Rey95b]. The labeling was conducted by students and corrected by an expert. It was based on perceptive evaluation of the speech signal as well as on visual inspection of F0 and energy contours. The labelers should not rely on their syntactic knowledge during labeling, however, in practice this cannot be avoided in all cases. At first a turn was segmented into prosodic clauses. Afterwards for each of the prosodic clauses the other boundaries and accents were labeled. Then the accented syllables were marked. It turned out that, except for 13 emphatic accents, within the accented words the syllable carrying the lexical accent received the perceptual label. Finally, the intonation was labeled as described in the next section. Table 5.10 gives a few numbers concerning the labels. In the case of the boundary labels the turn final labels have not been taken into account, because they are almost always labeled as B3 and they were not considered in the experiments described in this book.

corpus	boundaries				word based accents			
	B3	B2	B9	B0	PA	NA	EK	UA
BS_TRAIN	1,523	885	557	9,381	3,116	1,848	161	8,018
BS_TEST	165	126	76	1,082	407	277	6	823

Table 5.10: Frequency of all prosodic boundary and accent labels currently available for VERBMOBIL. The numbers for the boundaries do no take the end of turns into account.

5.2.3 Tone Labels

In the literature the ToBI labeling system is often used to describe the intonation of utterances. The VERBMOBIL sub–corpora BS_TRAIN and BS_TEST were annotated with these labels by Universität Braunschweig [Rey94]. For reasons given in the discussion below these labels are not considered any further, however, for completeness we give a short introduction.

Recall that intonation is the distinctive usage of pitch, cf. Section 4.1.2. That means that only general patterns which might contribute to the meaning of utterances are relevant. The ToBI labels are a symbolic description of the pitch contour in terms of these general patterns; based on the tone sequence theory of [Pie80, Pie90] it has first been developed for English [Sil92a] and was first adapted for German as described in [Rey94]; a consensus German ToBI labeling system was developed by the collaboration of several groups [Gri96]. ToBI basically distinguishes between high (H) and low (L) tones and combinations of them. Furthermore, these labels are combined with diacritics denoting one of three functions: clause boundary (%, called intonational boundary in ToBI), constituent boundary (–, called intermediate boundary in ToBI), and accents (*). Table 5.11 gives an overview of the different tone labels. The H of the first five basic accent tones can optionally be preceded by a fourth diacritic (!) marking a *downstep* if this !H tone is somewhat lower than the preceding H tone. This results in a total of 11 accent tone labels.

The tone labels are intended to describe the *form* of the intonation. However, in fact they are a mixture of perceptual, formal and functional labels: During labeling first the function of prosodic attributes is determined on the basis of perceptual evaluation, that is, accent, weak (B2) and strong (B3) boundaries are distinguished. On this basis the tones are labeled. For example, in [Gri96] it is stated that the difference between H– and H–L% is "*not tonal but relates to the perceived boundary strength*".

We do not want to discuss these labels in detail, because they will not be used any further. The main reason for this is that the linguistic modules within

TONES AT ACCENTED SYLLABLES	
H*	high pitch in the accented syllable
L*	valley shaped pitch: normal, low accent
L+H*	low pitch in preceding syllable, strong pitch rise on accented syllable
H+L*	high pitch in preceding syllable, pitch fall on accented syllable
L*+H	pitch rise on syllable; "delayed" peak on succeeding syllable
H+!H*	"very early" peak; pitch fall already in accented syllable

TONES AT B3 BOUNDARIES	
L–L%	low, falling pitch
H–H%	high, rising pitch
L–H%	from low to high strongly rising pitch
H–L%	level or high and slightly falling pitch

TONES AT B2 AND B9 BOUNDARIES	
H–	high phrase tone
L–	low phrase tone

Table 5.11: VERBMOBIL intonation labels based on the ToBI system

VERBMOBIL rely on *functional* prosodic classes, for example, +/– accented or +/– prosodic clause boundary. By which form elements these are realized does not play a role in VERBMOBIL. Therefore, we want to train prosodic classifiers, which directly detect these functional classes. If form elements (tones) were classified, they would anyway have to be mapped to the functional classes. Furthermore, in general in German the different tones are not as relevant with respect to the meaning as they are for English; consider the following example, from [Ste89]:

Fred ate the beans (5.18)
H*L L+H*LH%

Fred ate the beans (5.19)
L+H* LH% H* LH%

In both sentences Fred and beans are accented, however, due to the different intonation the meaning is different: the sentence in (5.18) could be a response to the question Well what about the beans? Who ate them? whereas (5.19) could have been preceded by Well, what about Fred? What did he eat? In [Ste89] it is concluded that H* intonation marks new information whereas L+H* is used to accent given information. Such a difference in the meaning of the same wording based on different tone sequences seems to be frequent in English but not in German [Fér93, p. 171]. In German the information about the presence or absence of a boundary is more important than the specific tonal marking. The tones can therefore be seen as an intermediate representation level between the raw F0 contour and the actual functional boundary or accent marking. A further drawback of the tone sequence system is that it does not take durational variation into account, which in German is an important prosodic attribute with respect to boundary as well as to accent marking [Kie94b].

Nevertheless, in VERBMOBIL as well as in EVAR also the direct classification of intonation plays a role in certain cases. However, this is only relevant at clause boundaries and only the three major classes *fall* (F), *rise* (R), and *continuation–rise* (CR) as described in Section 4.1.2 are so far considered to be useful [Sch95a, Abb96].

5.2.4 The Syntactic S Labels

There are some drawbacks in the perceptual prosodic boundary labels, cf. Section 5.2.2, if one wants to use prosodic information in parsing:

- Prosodic labeling by hand is very time consuming, the labeled database up to now is therefore rather small. On the other hand large labeled databases are needed for training of statistical models, which in the case of word recognition, where statistical HMM models have been used for two decades, can be obtained at much lower cost than prosodic labels, cf. [Ost93a].

- A perceptual labeling of prosodic boundaries is not an easy task and not very robust. [Sil92a] states that a requirement for a prosodic labeling scheme should be that different transcribers agree in at least 80% of the labels. [Rey95a] compared prosodic boundaries labeled by five naive transcribers for the VERBMOBIL corpus; only boundary and non–boundary was distinguished. Depending on the speaker the agreement varied between 67% and 84% percent. In [Gri96] three experienced transcribers agreed in 86% of the word boundaries upon whether there is no phrase boundary or a weak or strong prosodic phrase boundary. This was evaluated based on partly elicited speech read by trained speakers. Regarding this consult also our own evaluation presented in Section 5.1.4.

- Finally, prosodic boundaries do not necessarily only mirror syntactic boundaries but are influenced by other factors as rhythmic constraints and speaker specific style. For example, politicians in television discussions tend to put strong prosodic boundaries after every few words.

The last item will be discussed based on the following example (cf. also Section 4.2.3):

tja B2 da hätte ich nur noch B3 am Dienstag morgen B2 um acht Uhr B3 <Atmung> <Pause> bis um neun Uhr Zeit B3 das wär' am siebenundzwanzigsten April (5.20)
well B2 *then I only have time* B3 *on Tuesday morning* B2 *at eight o'clock* B3 *<breathing> <pause> until nine* B3 *that's on April the 27th*

Here, the verbal brace comprises everything between da hätte ich and Zeit; the first two B3 boundaries (before and after B3 am Dienstag morgen B2 um acht Uhr B3) are within the verbal brace and, therefore, do not correspond to syntactic clause boundaries. For other examples of clashes between syntactic and prosodic boundaries, cf. [Fel95]. In the worst case, such clashes might be lethal for a syntactic analysis if the parser goes the wrong track and never returns. Note that the position of such an "agrammatical" prosodic boundary is not fully arbitrary as it occurs almost always at some sort of constituent boundary. Inside a constituent, prosodic boundaries are normally B9 boundaries. On the other hand there are syntactic clause boundaries, which are only marked by a prosodic constituent boundary or are not at all prosodically marked, cf. the discussion in Section 5.2.7 and the syntactic boundary between main and subordinate clause after the word lieber in Figure 4.4.

Earlier experiments on the ERBA corpus showed that the prosodic–syntactic labels described in Section 5.1.2 can be successfully used for the training of prosodic classifiers, cf. Section 6.4 and [Kom94a].

This result and the above mentioned problems motivated our colleagues from IBM to label *clause boundaries* solely on the basis of syntactic criteria [Fel95], that is, not considering prosodic marking. The 24 dialogs of the sub–corpora S_TRAIN and BS_TEST were labeled. An exact labeling scheme was developed which distinguishes between 59 labels. These take into account the syntactic function of the phrases before and after the boundary.

We do not want to describe this labeling scheme in detail, because they will only be used for comparison with the rather rough labeling scheme presented in the following section. We also only want to distinguish between the following categories:

- S3+: Word boundaries where for sure *a* syntactic clause boundary exists.

- S3–: Word boundaries where for sure *no* syntactic clause boundary exists.
- S3?: Ambiguous syntactic boundaries, that is, it cannot be decided on the basis of the word chain whether there is a clause boundary.

For examples cf. Section 5.2.7, where the different labeling systems are compared.

5.2.5 The Syntactic–prosodic M Labels

A few acoustic–prosodic recognition experiments similar to the ones for the B labels as described in Section 6.4 were conducted with the S labels; the recognition results for S3+ versus S3– were comparable to the results for B3 versus ¬B3 and are described in [Bat96b], whereas, the S labels are better suited for our purposes because they are faster to obtain and they better match the requirements of a syntax module. However, the S labeling scheme is still complicated and its application is rather difficult and time consuming.

These positive results and the urgent need for a larger training database for acoustic–prosodic classifiers and in particular for prosodic–syntactic models encouraged us to develop a new labeling scheme which allows for an even much faster labeling as in the case of the rather complicated but precise syntactic S labels. We identify the following requirements for such a labeling scheme:

- It should allow for *fast labeling*. Therefore the labeling scheme should be rather coarse, because the more precise it is the more complicated and the more time consuming the labeling will be. Furthermore, a “small” amount of labeling errors can be tolerated, since it will be used to train statistical models, which should be robust to cope with these errors.
- *Prosodic tendencies* and regularities should be taken into account. In this context, it is suboptimal to label a syntactic boundary that is most of the time not prosodically marked with the same label as an often prosodically marked boundary. Since large quantities of data should be labeled within short time, only expectations about prosodic regularities based on the textual representation of a turn can be considered. It is not worthwhile to perform perception tests.
- The specific characteristics of *spontaneous speech* have to be incorporated in the scheme.
- It should be *independent* of particular syntactic theories but at the same time, it should be compatible with syntactic theory in general.

context	label	class
main/subordinate clause	M3S	M3
non–sentential free element/phrase, elliptic sentence	M3P	M3
extraposition	M3E	M3
embedded sentence/phrase	M3I	M3
pre–/ post–sentential particle with <pause>/<breathing>	M3T	M3
pre–/ post–sentential particle without <pause>/<breathing>	M3D	MU
syntactically ambiguous	M3A	MU
constituent, prosodically marked	M2I	M0
constituent, prosodically *not* marked	M1I	M0
every other word (default)	M0I	M0

Table 5.12: Overview over the boundary M labels and the respective main classes; examples are given in Table 5.13.

According to these requirements the M labels were developed in close cooperation with Anton Batliner, L.M. Universität München. He applied this scheme and labeled 7,286 VERBMOBIL turns. The effort only amounted to approximately four months. Note that so far we were mainly interested in achieving a large amount of labeled training data; we have not evaluated yet if the scheme, as it is defined below, can be applied consistently by different labelers, though we believe this is the case. In order to verify the consistency of the labels provided by Anton Batliner we compare them in Section 5.2.7 with the other types of boundary labels available for VERBMOBIL.

An overview of the M labels is given in Table 5.12 where the context of the boundaries is described shortly, and the label and the main class it is attached to is given. Examples follow in Table 5.13 in the same order; for convenience, only parts of the turns are given. Table 5.14 shows the frequency of the currently available M labels for the different sub–corpora. We distinguish three main classes:

- M3: prosodic–syntactic clause boundary
- M0: no prosodic–syntactic clause boundary
- MU: ambiguous prosodic–syntactic boundary

Consider again the examples in Section 4.2.3 in which it cannot be decided from the textual representation of a turn alone where the boundaries should be. Further examples are given below. These boundaries are labeled with M3D and M3A, main class MU, in our scheme.

label	example
M3S	vielleicht stelle ich mich kurz vorher noch vor M3S <Atmung> mein Name ist Lerch *perhaps I should first introduce myself* M3S *<breathing> my name is Lerch*
M3P	guten Tag M3P Herr Meier *hello* M3P *Mr. Meier*
M3E	da hab' ich ein Seminar M3E den ganzen Tag M3S <Atmung> *there I have a seminar* M3E *the entire day* M3S *<breathing>*
M3I	eventuell M3I wenn Sie noch mehr Zeit haben M3I <Atmung> 'n bißchen länger *possibly* M3I *if you've got even more time* M3I *<breathing> a bit longer*
M3T	gut M3T <Pause> okay *fine <pause>* M3T *okay*
M3D	also M3D dienstags paßt es Ihnen M3D ja M3S *then* M3D *Tuesday will suit you* M3D *won't it / after all* M3S
M3A	würde ich vorschlagen M3A vielleicht M3A im Dezember M3A noch mal M3A dann *I would propose* M3A *possibly* M3A *in December* M3A *again* M3A *then*
M2I	wie sähe es denn M2I bei Ihnen M2I Anfang November aus *will it be possible* M2I *for you* M2I *early in November*
M1I	M3S hätten Sie da M1I 'ne Idee M3S M3s *have you got* M1I *any idea* M3S

Table 5.13: Parts of VERBMOBIL turns showing examples for the M labels.

In the classification experiments presented in this book, we distinguished only between the main classes, because these are the classes linguists within VERBMOBIL were interested in for the beginning. In particular, the automatic detection of M3 boundaries allows for much more efficient parsing of spontaneous speech as shown in Section 8.3.2. Nevertheless, the distinction of the ten classes during labeling was considered to be useful, because their automatic discrimination might become important in the near future. Also because they partly mark special spontaneous speech phenomena, they might be interesting for a theoretical corpus–based linguistic study, especially, since there exists only a small amount of literature on this topic, except for example [Wei75, Tro94]. Furthermore, these boundary classes might be prosodically marked in a different way. At least in the case of English radio news speakers an acoustic–prosodic discrimination of different syntactic boundary classes is possible [Ost93b]. In the end, it was considered to be no

corpus	M3					
	M3S	M3P	M3E	M3I	M3T	$\sum$
M_TRAIN	10,432	4,092	1,163	347	294	16,328
BS_TRAIN	925	409	225	14	22	1,595
BS_TEST	116	34	10	8	9	177

corpus	MU			M0		
	M3D	M3A	$\sum$	M2I/M1I	M0I	$\sum$
M_TRAIN	4,495	475	4,970	–	–	104,857
BS_TRAIN	502	184	686	–	–	10,067
BS_TEST	55	48	103	132	1,037	1,169

Table 5.14: Frequency of all prosodic–syntactic M labels currently available for VERBMOBIL. The end of turns are not counted.

significant extra work to label these boundaries differently.

As for the M2I and M1I it was assumed that it is easier to label them after the M3 labels have been placed. Furthermore, there was no immediate need in these labels. So far only the BS_TEST sub–corpus has been labeled with M2I/M1I where M2I and M1I are even not distinguished because we intend to automatically turn certain M2I into M1I as we did for the ERBA B1 labels.

Note also that the agrammatical labels as described in Section 5.2.1 have already been present in the transliteration prior to the labeling of the Ms and were also used for the disambiguation between alternative M labels. However, in very agrammatical passages, a reasonable labeling with M labels is almost impossible.

At this point, it should be noted that written language is much easier to label, because it is syntactically regular and much less ambiguous than spontaneous speech. Furthermore, in written language punctuation already marks most of the boundaries; this is in German much more the case than in English. Because the detection of boundaries in an ASU system has to rely on prosodic information, a labeling system for spontaneous speech should take prosodic regularities into account. To our knowledge, this is the first scheme for the labeling of prosodic–syntactic boundaries in transliterations of spontaneous–speech. In the remainder of this section, we will define the different labels more precisely.

Syntactic Main Boundaries: M3S

These boundaries can be found

- between main clause and main clause,

- between main clause and subordinate clause, and
- before coordinating particles between clauses.

The beginning or end of turns are implicitely labeled with M3S. M3S corresponds roughly to the B3 boundary in ERBA, cf. Section 5.1.2, that should not be confused with the B3 boundaries in VERBMOBIL.

Boundaries at Non–sentential Free Elements (Elliptic Phrases): M3P

Boundaries at *non–sentential free elements* functioning as elliptic sentences are labeled with M3P. Ellipsis is a rather poorly treated phenomenon in the literature, but cf. [Kle93] and for a qualitative and quantitative analysis of part of the VERBMOBIL corpus cf. [Tro94]. Normally, these phrases do not contain a verb. They might be frozen elliptic clauses, that is, conventionalized expressions, idiomatic performative phrases with a sort of fixed meaning as guten Tag (*hello*) and vocatives. A criterion that can tell apart such frozen construction from "normal, productive" elliptic sentences is that it is not clear what the full form of the sentence might be: should guten Tag, for example, be expanded to an explicit performative sentence as, for example, ich sage Ihnen hiermit 'guten Tag' (*I hereby say 'hello' to you*)? The main difference between productive elliptic clauses and frozen elliptic clauses might be that the productive ones could be incorporated into the matrix clause or expanded into a full clause but they are very often not in the case of spontaneous speech. Typically, frozen elliptic clauses are *not* incorporated into any other clause. In our approach, boundaries at both productive and frozen elliptic clauses are labeled with M3P.

nach vierzehn fünfzehn Uhr M3P wie Sie 's sich frei machen dann (5.21)
after two three o'clock p.m. M3P *it's really up to you then*

In example (5.21) nach vierzehn fünfzehn Uhr is considered to be a *productive ellipsis* because it can easily be expanded to form a full "correct" clause: Nach vierzehn, fünfzehn Uhr können wir uns treffen (*We can meet after two, three o'clock in the afternoon*). At phrasal coordination, no M3P boundary is labeled before und. If such an elliptic clause is between two main clauses, we try to allot it to the clause it belongs "more closely" to, for example, separate it with M3P from this clause but with M3S from the other clause. For example, if we can analyze it as a left dislocation, an M3P is labeled on its right. Admittedly, this is not always an easy decision to make.

Boundaries at Extrapositions: M3E

Boundaries between a sentence and a phrase to its right, which in written language normally would be inside the verbal brace, form a subclass of the M3P boundaries. This phenomenon can be called extraposition or right dislocation with or without a pro element. A pro element is a construction or a word, for example, a pronoun, replacing another element, to which it implicitly refers to. Extraposition is a rather complex and not very well understood phenomenon that has to be dealt with especially. Prosodically, these boundaries might be free variants: either they are prosodically marked or not.

We can define prototypical cases but as for the degree of extraposition there might be a continuum where somewhere on this continuum is the boundary between +/– extraposition. Consider the following examples:

treffen wir uns am Donnerstag den achtzehnten (5.22)
let's meet on Thursday the 18th

treffen wir uns am Donnerstag am achtzehnten (5.23)
let's meet on Thursday on the 18th

treffen wir uns am Donnerstag M3E und zwar am achtzehnten (5.24)
let's meet on Thursday M3E *that is on the 18th*

treffen wir uns am Donnerstag <Pause> M3E am achtzehnten (5.25)
let's meet on Thursday <pause> M3E *namely on the 18th*

treffen wir uns am Donnerstag <Pause> M3E und zwar am achtzehnten (5.26)
let's meet on Thursday <pause> M3E *that is on the 18th*

treffen wir uns am Donnerstag <Pause> M3E und zwar wäre es am achtzehnten am besten (5.27)
let's meet on Thursday <pause> M3E *in fact, the 18th would be best*

In these examples, if a distinct prosodic marking of this boundary can be perceived an M3E should be labeled. Since our scheme is supposed to be applied only on the basis of the transliteration, we have to rely on the marking of pauses and/or breathing and/or particles like und zwar in the basic transliteration. If this boundary is marked only by prosodic means the defining criterion is outside the realm of syntax proper. Possibly, we can deal with cases like these only in a syntactic theory that *incorporates* prosody but not in a syntactic theory that only *uses* prosodic information, compare the suggestions in [Ste92].

In cases where the extrapositions are syntactically not as much linked to the context as in the above examples, we label M3E independently from indicators like pauses, cf. example (5.28).

müssen wir noch den Termin im Oktober setzen M3E für die gemeinsame Besprechung M3S (5.28)
we still have to fix the date in October M3E *for our discussion* M3S

As a conclusion we want to emphasize that the marking of boundaries with M3E depends mostly on *expectations* about word boundaries where the speaker *can* produce a prosodically marked boundary but must not. If he does, it is a strong indication of an extraposition, if he does not, it is no indication at all, neither for a syntactic boundary nor for *no* syntactic boundary.

Boundaries at Embedded Sentences: M3I

Sentences or non–sentential free elements that are embedded in a sentence are labeled with M3I. These are typically parentheses as in example (5.29). Example (5.30) contains the short parenthetical expression glaub' ich (*I guess*). This one and other similar ones as, for example, wie gesagt (*as I said*) are not labeled because most certainly, they are integrated prosodically in the embedding sentence. However, words like Moment (*just a second*), are treated differently although they are just one word: they are a sort of discourse signal and are much more relevant for a correct interpretation of the meaning. Therefore it is rather likely that they are preceded or followed by a boundary, which we labeled with M3P.

Samstag morgens M3I bin ich ganz ehrlich M3I ist mir gar nicht recht (5.29)
Saturday morning M3I *I will be quite honest* M3I *doesn't suit me at all*

die Einzelheiten werden glaub' ich die Sekretärinnen abstimmen (5.30)
(the details will guess I the secretaries coordinate)
the secretaries I guess will coordinate the details

Boundaries at Pre–sentential and Post–sentential Particles: M3D and M3T

Very often in spontaneous speech, a turn begins with pre–sentential particles (German: *Satzauftaktpartikeln*), for example, ja, also, gut, okay. These are either discourse particles with no specific meaning but functioning as a sort of turn taking signal like well in English or they are elliptic clauses functioning as, for example, a confirmation. Note that discourse particles often cannot be directly translated into

English [Rip96c]. These pre–sentential particles can also occur inside a turn at the beginning of a clause or phrase as a sort of turn continuation signal in order to hold the floor, and rather seldom, they can be clause final or even turn final.

Prototypically, these particles might be separated by a prosodically marked boundary if they are used as confirmation, but as a discourse governing signal, they are not prosodically marked or are even clitisized. Ideally one should mark both boundaries separately, however, as said before we do not want to perform perception tests during labeling. Therefore, pre–sentential particles that are followed by a pause or by breathing are labeled with M3T, all others with M3D regardless of the function of the particle. However, the use of these particles as discourse particles seems to be the most frequent one and can thus be assumed to be the default, in the case of M3D. In the VERBMOBIL dialogs ja, for example, is in 90.6% in initial position, in 2.5% in second position; in these cases in more than 90% it functions as turn taking signal meaning well, cf. example (5.31); in the other cases it is a confirmation meaning yes, cf. example (5.32) and [Fis95]. On the other hand turn initial also, for example, is almost always a discourse particle and it is very often clitisized, cf. example (5.33). Within a turn it can also be a conjunction meaning *so*. The particle oh can be clitic or elliptic; in any case it signals a paralinguistic meaning (astonishment).

<Atmung> ja M3D Morgen M3P Frau Icks M3P ähm wann wär's Ihnen denn recht <#Klicken> (5.31)
<breathing> [well] M3D *good morning* M3P *Mrs. Icks* M3P *ehm when would it suit you <#click>*

und<Z> später noch mal <Pause> von vierzehn Uhr <Atmung> bis <Pause> siebzehn Uhr M3S <Atmung> okay M3T <Atmung> gut M3T <Atmung> <ähm> <Schmatzen> ja M3T <Pause> dann sehen wir uns also <Pause> Dienstag (5.32)
and<Z> later on again <pause> from two p.m. <breathing> until <pause> five p.m. M3S *<breathing> okay* M3T *<breathing> well* M3T *<breathing> <ehm> <smack> well* M3T *<pause> then we will see each other <pause> on Tuesday*

<Atmung> ja M3T <Atmung> also M3D für den eintägigen M3I <Pause> wenn wir den als <Lachen> erledigen wollten quasi M3I <Atmung> wäre mir ganz recht (5.33)
<breathing> yes M3T *<breathing> [well]* M3D *for the one day* M3I *<pause> if we want it <laughter> finish quasi* M3I *<breathing> I would prefer*

A post–sentential ja can be a tag meaning isn't it or it can be a modal particle, best translated with do, as in

ich weiß ja M3S daß das gut geht (5.34)
I do know M3S *that it will be okay*

or it can as well be a confirmation:

<hm> n nein M3P I lieber zehn Uhr M3P <hm> neun Uhr <hm> kann ich am Donnerstag glaub' ich doch nicht <Atmung> <hm> ja M3D zehn Uhr ist besser M3D ja <Atmung> (5.35)
<hm> n no M3P *better at ten o'clock* M3P *<hm> nine o'clock <hm> on Thursday I think I don't have time* M3S *<breathing> <hm> yes* M3D *ten o'clock is better* M3D *yes <breathing>*

Note that in this example ja could also be a discourse particle, but when listening to the speech signal it becomes clear that it is intended as a confirmation. No matter which function, ja is preceded by either M3D or M3T. So again we put the corresponding label depending on the presence of a pause. Inside a sentence or a phrase, these particles often function as a modal particle and are not labeled at all, cf. [Rip96c]. As for a thorough treatment of particles like these in German cf. [Wil88].

Syntactically Ambiguous Boundaries: M3A

These boundaries occur between two clauses when the position of the boundary cannot be determined for purely syntactic reason, as in the examples (5.36) and (5.37). Often there are two or more alternative word boundaries, where the syntactic boundary could be placed. It is therefore the job of prosody to disambiguate between two alternative readings. Consider the following examples:

fünfter geht jetzt M3A bei mir M3A nicht M3A aber <#Geräusch> neunzehnter M3A wär' in Ordnung M3S könnten wir das festhalten (5.36)
the fifth is okay M3A *with me* M3A *not* M3A *however <#noise> the nineteenth* M3A *would be alright* M3S *could we agree on that*

<Pause> genau M3A das wirft natürlich bei der Planung einige Probleme auf (5.37)
<pause> exactly M3A *that will of course cause some problems during planning*

In example (5.36), four M3A boundaries are labeled. The context might render some of these rather unlikely; the most likely boundary is the one after nicht. In cases like these, we try to label with M3A all reasonable positions but no rather strange positions. In example (5.37), genau could be a sentence adverb functioning as a confirmation or it could be an emphasizing adverb. Both readings are plausible; only in the first case there would be a clause boundary.

M3A overrides all other labels, compare the following examples:

also M3D der Zug geht morgens (5.38)
as I said M3D *the train leaves in the morning*

<Atmung> stimmt M3A also M3A der Zug geht morgens (5.39)
<breathing> you are right M3A *as I said* M3A *the train leaves in the morning*

The M3D from example (5.38) is replaced by M3A in (5.39), because at one of the M3A positions an M3S and at the other one an M3D should be labeled, but it is unclear from the text alone which reading is the intended one.

In the course of our rather fast labeling, some of the M3A boundaries could most certainly not be detected because they only show up in a slow, close reading. This fact and our non–strict syntactic labeling strategy might be the reason for a partly missing mapping of S3? with M3A, cf. Section 5.2.7.

Constituent Boundaries: M2I and M1I

M2I and M1I denote constituent boundaries. Their definitions are along the lines of the ERBA B2 and B1 boundaries, cf. Section 5.1.2: An M1I constituent boundary is in the vicinity of the beginning or the end of a clause and is normally not prosodically marked because of rhythmic constraints. An M2I constituent boundary is inside a clause or phrase, not in the vicinity of beginning or end, and it is rather often prosodically marked, again because of rhythmic constraints; cf. the following example:

M3S mir ist M1I sowohl M1I Sonntag der zwölfte M2I als auch M1I der neunzehnte M1I recht M3S (5.40)
M3S *with me* M1I *both* M1I *Sunday the twelfth* M2I *as well as* M1I *the nineteenth* M1I *are okay* M3S

Similar to the labeling in ERBA it might be possible also for the VERBMOBIL data to first label manually every constituent boundary with a single symbol and thereafter to automatically split these labels into M2I and M1I according to a few rules. The M2I boundaries are expected to often correspond to the perceptual B2 boundaries, however, the degree of freedom for the speaker is high in these cases, so that it often could also correspond to B0 or B3, cf. example (5.20). Also the labeling is not easy and might not be consistent. They constitute therefore a source of missing correspondence between B3 and M3.

It is still an open question whether ASU will benefit from such a labeling — but if it is reliable enough, it surely will. So far in VERBMOBIL a reliable detection of M3 had priority, therefore, for the time being, M2I is only labeled in three dialogs, and M1I is not labeled at all.

5.2.6 Dialog Act Boundaries: the D Labels

In Section 3.4 we mentioned as one task of the dialog module of the VERBMOBIL system the tracking of dialog acts. For this classifiers have to be trained on a corpus for which reference labels should be available. Since a turn can consist of more than one dialog act the labeling should include both, the dialog act names and a respective segmentation of the turns. In the following these dialog act boundaries will be denoted as D3, because they can be expected to highly correlate with M3. All word boundaries not labeled as D3 are denoted as D0. The sub–corpora D_TRAIN, D_TEST and BS_TEST were segmented into such units, and were labeled with one or more dialog acts. This work has been described in [Mas95b, Jek95]; we will give a summary in the following since in Section 8.6.1 we will describe recognition experiments with these boundaries.

The dialog acts are defined according to their illocutionary force for example ACCEPT, SUGGEST, REQUEST and can be sub–categorized on their functional role or their propositional content (for example DATE, LOCATION) depending on the application. In the VERBMOBIL domain 18 dialog acts on the illocutionary level and 42 dialog acts altogether were defined when including also the sub–categories [Jek95]. Since in spontaneous speech there are few grammatically complete and correct structures, it is not easy to give a quantitative and qualitative definition of the term dialog act. Criteria for the segmentation of turns based on the textual representation of the dialog were defined in [Mas95b] and were used in labeling German as well as English turns with dialog acts. Examples for such criteria are:

- All words that belong to the verb frame of a finite verb are part of the same dialog act segment.
- Conventionalized expressions as hello, good morning, and thanks are considered as separate segments even if they do not contain a verb.

Prosodic features were not taken into account in order to be able to label turns without having to listen to them and thus to reduce the labeling effort. However,

corpus	D3	D0
D_TRAIN	4,189	44,895
D_TEST	662	7,317
BS_TEST	113	1,336

Table 5.15: Frequency of all dialog act boundary labels currently available for VERBMOBIL. The end of turns are not counted.

for the automatic segmentation of dialog acts prosodic markers are very important cues, as has been discussed in Section 4.2.3 and as we will show in Sections 5.2.7 and 8.6.1. Examples (5.41), (5.42), and (5.43) each show a typical English turn segmented into dialog acts and labeled respectively.

no I have a class then 'till twelve	REJECT–DATE
eh can you make it like after twelve on the eleventh	SUGGEST–SUPPORT–DATE

(5.41)

eh Matt this is Brian here again	INTRODUCE–NAME
we have to discuss the documentation ehm sometime this month	SUGGEST–SUPPORT–DATE, MOTIVATE–APPOINTMENT

(5.42)

well I have a meeting all day on the thirteenth	SUGGEST–EXCLUDE–DATE
and on the fourteenth I am leaving for my bob sledding vacation until the nineteenth	SUGGEST–EXCLUDE–DATE
eh how 'bout the morning of the twenty second or the twenty third	SUGGEST–SUPPORT–DATE

(5.43)

Usually a dialog act segment corresponds to one dialog act as shown in (5.41), but sometimes it functions as more than one dialog act, cf. the second turn segment in example (5.42). In example (5.43) two subsequent dialog act segments carry the same dialog act label, SUGGEST–EXCLUDE–DATE. These segments are not integrated into one segment, because different dates are suggested.

Since the VERBMOBIL prosody module so far only operates on German speech, we will restrict ourselves in the following to the dialog act boundaries of the German VERBMOBIL corpora. Table 5.15 shows the frequency of these labels for different VERBMOBIL sub–corpora.

	#	S3+	S3?	S3–
M3	951	**84.3**	8.4	7.2
MU	391	79.3	9.2	11.5
M0	6,297	1.1	2.3	**96.6**

Table 5.16: Percentage of M labels coinciding with the different S labels.

	#	M3	MU	M0
S3+	1181	**67.9**	26.2	5.8
S3?	259	30.9	13.9	55.2
S3–	6199	1.1	0.7	**98.2**

Table 5.17: Percentage of S labels coinciding with the different M labels.

5.2.7 Correspondence between the Boundary Labels

In the previous sections four different types of boundary labels were presented, namely the perceptual B labels, the S labels for syntactic clause boundaries, the prosodic–syntactic clause boundary labels (the M labels), and the D labels for dialog act boundaries. Here we will investigate the degree of correspondence between these labels. With this we mainly want to justify the M Labels, because due to the simplicity of the labeling scheme and to the short time in which they were created, we could not be sure if the labels were adequate or contained too many errors or inconsistencies.

For the S_TRAIN corpus all of these four types of boundary labels are available. For these data Tables 5.16 to 5.20 give the figures of correspondence, which will be discussed in this section. In all of the tables the second column shows the frequencies of the labels out of the first column. All other numbers show the percentage of the labels at the left–hand side coinciding with the labels given at the top. For example, the number "84.3" in the second column, first row of Table 5.16 means that 84.3% of the word boundaries labeled with M3 are also labeled with S3+. Those numbers, where from the definition of the labels a high correspondence could have been expected a priori, are depicted with bold face. Note that turn final word boundaries are not considered in the tables, because these are in all cases labeled with S3, M3 and D3 and in most cases with B3.

A Comparison of M and S Labels

With respect to the justification of the M labels, our primary interest is the degree of correspondence between the M and the S labels, because the latter were based on a thorough syntactic analysis. Tables 5.16 to 5.18 show first of all that there is

	#	S3+	S3?	S3–
M3S	502	**94.2**	0.6	5.2
M3P	288	**93.4**	3.8	2.8
M3E	148	**34.5**	43.2	22.3
M3I	6	**50.0**	33.3	16.7
M3T	7	**85.7**	0.0	14.3
M3D	301	90.0	3.6	6.3
M3A	90	43.3	27.8	28.9
M0	6297	1.1	2.3	**96.6**

Table 5.18: Percentage of the subclasses of the M labels coinciding with the different S labels.

a very high correspondence between M0 and S3–: 96.6% of the M0 correspond to S3– and 98.2% of the S3– boundaries are also labeled as M0. Second, 84.4% of the M3 are also labeled as S3+. It should be noted that this number is higher than the correspondence between perceptive prosodic labels created by different subjects, cf. [Rey95a] and page 162. The only number relevant for the correspondence and being rather low is the 67.9% of S3+ labeled also with M3. Most of the remaining S3+ (26.2%) are labeled as MU. A closer look at the subclasses of the M labels shows that this is not due to errors, but it has systematic reasons resulting from different syntactic–theoretic assumptions. Mainly responsible for this mismatch is that the majority of the M3D labels, a subclass of MU, is labeled with S3+. This causes also the low correspondence between MU and S3?. Further the examples (5.44), and (5.45) show typical parts of turns, where S3+ systematically does not correspond to M3.

sagen wir lieber S3+/M2I vierzehn Uhr fünfundzwanzig (5.44)
let's rather say S3+/M2I *two twenty five p.m.*

aber S3+/M0 Donnerstag vormittag . . . wär' mir recht (5.45)
but S3+/M0 *Thursday in the morning . . . would be fine*

We believe that both labeling systems are correct, but the M labels might correspond to a rather general view of syntax whereas the S labels represent a syntactic theory, which is *traditional* in the sense that it does not or only to a small extent take into account spontaneous speech phenomena like extraposed phrases. This is the reason, why only 34.5% of the M3E labels, a subclass of M3, correspond to S3+. For a discussion of this topic cf. [Haf93]. The S labels were also developed with respect to a specific grammar formalism, whereas with the M labels we tried to meet a rather broad consensus about syntactic structure. This, especially, explains the differences in example (5.45). Furthermore, a systematic difference between the S and the M labels is caused by the fact that the latter take prosodic knowledge into account. In the case of the M3D it really depends on the prosody if there is a clause boundary or not. Also the M3E labels are to a great extent based on expectations about prosodic boundary marking. This is the reason why there is no consistent correspondence between the M3E and any of the S labels.

On the other hand the subclasses M3S and M3P correspond to S3+ in well over 90% of the cases. This meets our expectations, because these cases should be quite independent from the specific theory. Also here and in the rare S3–/M3 mismatches (1.1%) some systematic differences can be observed, however, we do not want to discuss these any further.

All in all we can draw the conclusion that the M labels are fairly consistent despite the simple scheme and despite the fast creation of the labels. Furthermore, they have the advantage of taking into account spontaneous speech phenomena and prosodic regularities. So they should be well suited to train statistical models. In fact, our results of Sections 7.2 and 8.3 show that they can be reliably predicted with stochastic n-gram models, and, furthermore, the use of these models improves the syntactic analysis in the VERBMOBIL system. These results, especially, are an argument for the consistency of our labeling scheme.

The Prosodic Marking of the M Labels

Since we claimed that the M labels reflect also prosodic knowledge we have to analyze the prosodic marking of the M labels, that is, the correspondence between the different M subclasses and the B labels, which is depicted in Table 5.19.

At first it can be noted that the sentence or clause boundaries M3S are mostly (87.8%) marked with a B3 boundary. This is not surprising, because as has already been discussed in Section 4.2.3 it is "well known" that there is a high correspondence between syntactic and prosodic boundaries. However, to our knowledge this is the first investigation of a large corpus of spontaneous German speech concerning this hypothesis. The 8% of the M3S which correspond to B2 are mostly boundaries between main clause and subordinate clause, where the speaker has prosodically marked the boundary only by a slight continuation rise. For subordinations it might especially be to the discretion of the speaker to which extent prosodic marking is used. In this context the overall speaking rate might play a role. In the following we give three typical examples, (5.46) to (5.48), where a prosodic marking of an M3S by either B3 or B2 is acceptable and even B0 occurs sometimes. The actual VERBMOBIL B and M labels are depicted in each case:

	#	B2	B3	B9	B0
M3S	502	8.0	**87.8**	0.0	4.2
M3P	288	10.4	**75.7**	0.3	13.5
M3E	148	11.5	**53.4**	0.0	35.1
M3I	6	0.0	**66.7**	0.0	33.3
M3T	7	0.0	**85.7**	0.0	14.3
M3D	301	32.9	24.6	0.3	42.2
M3A	90	16.7	35.5	1.1	46.7
M0	6297	4.6	2.8	3.7	**88.9**

Table 5.19: Percentage of M labels coinciding with the different B labels.

wenn ich da so meinen Terminkalender anschaue B3/M3S das sieht schlecht aus (5.46)
if I look at my calendar B3/M3S *that looks bad*

ich finde schon B2/M3S daß wir uns noch im März treffen sollten (5.47)
I really think B2/M3S *that we should meet before end of March*

<Pause> ja M3D ich sagte B0/M3S daß ich am <äh> Donnerstag den fünfzehnten April ... (5.48)
<pause> yes/well M3D *I said* B0/M3S *that I ... on <eh> Thursday the fifteenth of April*

The following is an example where a clause boundary has not been prosodically marked at all despite the fact that there is no subordination marking the boundary.

<Atmung> ich denke B0/M3S wir sollten das Ganze dann<Z> doch auf die nächste Woche verschieben (5.49)
<breathing> I think B0/M3S *we should move the whole thing to next week*

Nevertheless, from the syntax it is clear (in a left to right analysis already at the word wir) that there is a boundary and the first clause is a rather short conventionalized expression where the prosodic marking would not contribute much, if at all, to the interpretation of the utterance. Many of these B0/M3S "correspondences" occur after short main clauses like ich denke (*I think*), ich glaube (*I believe*), ich würde sagen (*I would say*), meinen Sie (*do you think*) or was glauben Sie (*what do you think*).

Also a high number (75.7%) of the M3P boundaries are marked as B3 but the correspondence is lower than between M3S and B3. Nevertheless, this number

	#	M3	MU	M0
D3	533	**91.9**	5.2	2.8
D0	7106	6.5	5.1	**88.4**

Table 5.20: Percentage of D labels coinciding with the different M labels.

is still within the range of agreements between different people labeling the B boundaries [Rey95a]. The lower correspondence of the M3P with respect to the M3S can be explained with the fact that M3P separate elliptic phrases. These often are quite short so that the same argument as above for the elliptic clauses holds for the short main clauses.

It meets our expectations that 35.1% of the M3E are not prosodically marked at all. We hope to be able in the future to split M3E into subclasses based on a detailed analysis of these cases. Example (5.50) shows two M3E boundaries, where only the first one is prosodically marked at all and it is also rather weak.

<Atmung> gut M3D dann sehen wir uns am Mittwoch dem fünfzehnten B2/M3E um sechzehn Uhr dreißig M3E in Ihrem B"uro <Atmung> <#Klicken> <Schmatzen> (5.50)
<breathing> fine M3D *then we see each other on Wednesday the fifteenth* B2/M3E *at four thirty p.m.* M3E *in your office <breathing> <#click> <smacking>*

The M3D labels coincide with B3, B2 and B0 without any clear preference. This can be expected, because the M3D mark ambiguous boundaries at rather short phrases. On the other hand at the positions of the M3A, the really ambiguous boundary positions between clauses, either a strong boundary marking (B3 in 35.5% of the cases) or no marking at all (B0, 46.7% of the cases) can be observed, which also meets our expectations.

Finally, from Table 5.19 it can be seen that in accordance with their definition almost all B9 boundaries do not coincide with major syntactic boundaries (M3).

From this analysis we can conclude that the M labels meet prosodic regularties to a great extent and should therefore be useful for the training of classifiers for boundaries on the basis of acoustic–prosodic features.

Difference between Dialog Act and M Boundaries

It is of further interest how M3 and D3 are related. Both M and D labels have been created rather rough and fast. Despite of this, the numbers in Tables 5.21 and 5.20 are consistent with our expectations: most of the D3 correspond to M3 and vice

	#	D3	D0
M3	951	51.5	48.5
MU	391	7.2	92.8
M0	6297	0.2	**99.8**

Table 5.21: Percentage of M labels coinciding with the different D labels.

	#	B2	B3	B9	B0
D3	533	6.2	**91.4**	1.5	0.9
D0	7106	6.4	7.7	3.2	82.7

Table 5.22: Percentage of D labels coinciding with the different B labels.

versa almost all of the M0 correspond to D0. Ideally, all M0 should be labeled with D0. Most of the mismatches result from general difficulties of labeling spontaneous speech especially in the context of disfluencies; some other are simply caused by labeling errors. Furthermore, only about half of the M3 correspond to D3, that is, a turn segment corresponding to a dialog act can consist of more than one clause or free phrase. Let us consider the following turn, which can be segmented into four clauses or two dialog acts:

ich muß sagen M3S mir wär's dann lieber M3S wenn wir die ganze Sache auf Mai verschieben D3/M3S <Pause> geht es da bei Ihnen auch <#Klicken> (5.51)

I would say M3S *I would rather prefer* M3S *if we would move the whole thing onto May* D3/M3S *<pause> does this suit you as well <#click>*

It does not surprise that only 7.2% of the MU labels coincide with a D3 boundary. Looking at the subclasses, 3.3% of the M3D and 20% of the M3A coincide with a D3 boundary.

Table 5.22 shows that 91.4% of the D3 boundaries are strongly marked prosodically, that is, with a B3 boundary. This number is much higher than the percentage of M3 boundaries corresponding to B3, which is 78.6% as can be determined on the basis numbers given for M3S, M3P, M3E, M3I and M3T in Table 5.19. It is even higher than the 87.8% M3S/B3 correspondence. This confirms the results of other studies which showed that boundaries at discourse units are extraordinary strongly marked by prosodic means, as has been discussed at the end of Section 4.2.3. In other words, dialog act internal clause boundaries are prosodically less marked than clause boundaries coinciding with dialog act boundaries.

5.2.8 Correspondence of the Different Boundary Labels with Pauses

We were often asked how much the different boundary labels correspond with pauses. If there were a high correlation with one of the labels, this would have two consequences:

1. We would not need to label this explicitly, because pauses are anyway marked in the transliteration of the VERBMOBIL corpora.
2. The automatic recognition of these boundaries would be trivial, because one would simply need a pause detector.

In Table 5.23 a few percentages concerning these correspondences are given. The numbers were determined on the VERBMOBIL sub–corpus BS_TRAIN which consists of 30 dialogs. In this evaluation breathing was considered as pause. It can be seen that in general all major boundary labels, that is, M3, D3, and B3 do not correlate much with pauses: only less than 70% of the pauses coincide with such a boundary and only between 30% and 40% of these boundaries are marked by a pause. With respect to the comparison of M3 and D3 above it is not surprising that more D3 than M3 boundaries are marked by a pause. From the comparison of pauses with the perceptual B labels we can conclude that pauses in 97% of the cases coincide with any of the B boundaries. However, this is due to the definition of the B9 boundaries which are prosodically marked boundaries being within syntactic constituents. It was assumed that a pause usually marks a prosodic boundary. These numbers also show that the 24% of the pauses coinciding with B9 are pauses not coinciding with any syntactic boundary.

We can conclude that pauses are an unreliable indicator of linguistic boundaries, especially, of the major ones. It might be the case that the marking of pauses in the transliteration which was used for this evaluation is not reliable. In this case one can expect that even more pauses would be labeled and only few of the existing pause labels would be deleted. However, it cannot be expected that the general conclusions drawn from Table 5.23 would have to be revised.

5.3 Time–of–day Expressions

For the training of classifiers for the sentence mood of elliptic sentences containing only time–of–day expressions we developed a small corpus of read speech. The text corpus was generated by an ERBA like grammar, where the time–of–day

	M3	MU	M0	D3	D0	B3	B2	B9	B0
% pauses coinciding with the label	63	3	34	47	53	69	4	24	3
% labels coinciding with a pause	30	4	3	38	4	35	4	32	0

Table 5.23: Correlation between pauses and different boundary labels in the VERBMOBIL sub–corpus BS_TRAIN. The beginning and end of the turns was not taken into consideration.

expressions were annotated with a label indicating the sentence mood. In the following we give a few examples:

4 Uhr 35 ?
20 Uhr 17 – (5.52)
22 Uhr 27 .

The word Uhr means o'clock. The sentence moods are statement (indicated by “.”), question (“?”) and feedback, denoted by “–”. The intonational marking corresponding to the functional class *feedback* is continuation–rise. From the text corpus we selected 30 sentences for each of these classes of sentence mood. Each pair of sentences out of the resulting set of 90 sentences contained different time–of–day expressions.

Four speakers, one female and three males, read the entire corpus of 90 sentences. They were told to produce the intonation as indicated by the label. The speakers were untrained but they were educated in the field of prosody. The recording was conducted in a quiet office environment with a close–talking microphone.

Because the sentences were read out–of–the–blue, in the case of questions and feedbacks the speakers had difficulties in producing the *correct* intonation, that is, the intonation marking the utterance in accordance with the labeled sentence mood. Based on perception tests we excluded 23 misproductions from the corpus. Other 15 utterances were excluded because of coarse errors in the automatically computed F0 contour. Note that this is the only corpus used in this book where utterances were sorted–out for such reasons.

Eventually, 322 utterances remained. In the following we will refer to them as the TIME–OF–DAY corpus. It consists of a total of 690 secs of speech; the average duration of an utterance is 2 secs.

5.4 The Munich SPONTAN Data

This corpus was recorded and digitized at the L.M. Universität München for the purpose of investigating the difference between spontaneous and read speech. Two "naive" persons were sitting at a table without seeing each other. They had the task to jointly solve a problem within a blocks world environment. The persons were not aware that the experiment was conducted to aquire speech signals. They were told that the experiment was part of a study about cooperative problem solving. As a consequence a speech database of high degree of spontaneity could be recorded. After nine months the same persons were asked to read the transliterations of the dialogs. Each person read his own part and the utterances of the partner.

In this way a speech sample from four speakers resulted, which consisted of 1329 utterances, 28 minutes of speech. One third of the data was obtained from spontaneous dialogs, two third are elicited speech. The sampling frequency was 10 kHz. In the scope of this book the corpus was used for the evaluation of algorithms for F0 determination.

5.5 Summary

The experiments presented in the following chapters were conducted using three different corpora: Initial studies were performed using ERBA, a corpus of 10,000 different sentences from the domain of train time table inquiries. They were read by 100 untrained speakers. In order to achieve a text corpus of so many different sentences we used a context–free grammar to generate the sentences. For most of the experiments we used the German part of the VERBMOBIL spontaneous speech corpus. It consists of dialogs between two partners who got the task to schedule an appointment. Since the tasks were rather unrestricted the data is rich in spontaneous speech phenomena. Such phenomena are constructions where the word order is agrammatical in the sense of written language, left and right dislocation, elliptic clauses, word interruptions, repairs, restarts, hesitations, filled pauses and all kinds of human non–verbal articulation and noise. The VERBMOBIL sub–corpora used in this book altogether consist out of 8,262 utterances. For few experiments, furthermore, a small database of elicited elliptic time–of–day expressions was used.

Since we use statistical classifiers for the recognition of prosodic accents and boundaries, large amounts of labeled training data are needed. The creation of these reference labels is an important issue. Note that all the approaches presented in the following do not need time–aligned prosodic labels. It is sufficient that the reference labels are introduced at the correct position in the transliteration of the utterances. The most frequently used approach for the creation of prosodic labels is based on the ToBI system. This relies on the perceptual evaluation of the speech and on the visual inspection of the F0 contour. Such a labeling is very time consuming and it is not as consistently applicable by different labelers as one might expect: agreements are below 85% percent.

Therefore, we developed methods or schemes for the creation of such labels based on text corpora. The goal was to achieve large amounts of labels within rather short time. In the case of ERBA the task was relatively easy, because the sentences are grammatically well–formed and because they were automatically generated. Therefore, after the corpus had been read we introduced labels for prosodic–syntactic constituent (B2) and clause boundaries (B3) into the grammar and generated the corpus again. Afterwards, based on a few rules those B2 boundaries being close to B3 and enclosing clitic constituents were turned into B1, which are in contrast to B2 and B3 not likely to be prosodically marked. Accent labels were afterwards placed according to the principle that the rightmost content word in a phrase is accented by default. On a subset of the corpus we found that there is an over 90% agreement between listener judgments and automatically created labels. These labels were thus considered to be adequate for the training of prosodic classifiers.

For the VERBMOBIL corpus we defined along the lines of ERBA a scheme for the labeling of prosodic–syntactic labels, called the M labels. Since within short time large amounts of data should be labeled, the scheme is based only on the transliterations. We defined ten classes of boundaries taking into account frequent spontaneous speech phenomena such as free phrases and knowledge about prosodic regularities especially concerning the boundary marking of such phrases. In the remaining chapters we only use the three main classes M3 (clause boundary), MU (ambiguous clause boundary) and M0 (no clause boundary). Constituent boundaries are included in M0. Meanwhile over 7,200 VERBMOBIL utterances have been labeled.

Other groups developed and applied different boundary labels: The ToBI scheme was applied to 861 VERBMOBIL utterances. It distinguishes several tone labels describing the form of the pitch contour, as well as functional boundary and accent labels. As for the boundaries prosodic constituents (B2) and clauses (B3) were labeled. A subset of these utterances was strenuously but exactly labeled with a fine scheme of 59 purely syntactic boundaries, the S labels. Eventually, the transliterations of over 3,000 VERBMOBIL utterances were labeled with dialog acts including the boundaries in between them, which we call D3. In comparing these labels with our M labels we found a very high agreement that led us to the conclusion that the M labels provide a scheme which allows for not only fast but also for consistent labeling.

In the remaining chapters we will use these labels for the training of statistical acoustic–prosodic and prosodic–syntactic models, which finally will be integrated in the EVAR and VERBMOBIL ASU systems.

Chapter 6

Preprocessing and Classification

A prerequisite for an integration of prosodic information in ASU systems is a robust classification of prosodic attributes such as sentence mood, accent and phrase boundaries. This includes appropriate preprocessing of the speech signal, pitch determination and feature extraction. The methods which were used in our experiments described in the remaining chapters were developed by Andreas Kießling [Kie97]; this chapter gives a short review. The classifiers were trained using the various label categories described in the previous chapter. Most important with regard to the experiments presented in Chapters 7 and 8 are the NNs described on pages 201–204 which classify prosodic clause boundaries.

6.1 Neural Inverse Filtering

Before we come to the review of the work done by Andreas Kießling we will describe a method for inverse filtering of speech signals which we developed alternatively to the approaches described in [Kie97, Sec. 6.1][1]. In Section 4.1.1 we described the speech production as a two–step process: The *voice source* (or *excitation*) signal (VSS) is generated by the larynx and contains the F0 and its harmonics; it is subsequently filtered by the *vocal tract* to give the signal the phone specific characteristics. The term *inverse filter* refers to the vocal tract, that is, one tries to find a filter which removes the phone information from the signal. The resulting signal should be as close as possible to the VSS. The determination of the voice source characteristics, which are F0 and laryngealizations, should be much more reliable based on this signal opposed to using the speech signal directly. In fact, F0 computation is almost trivial based on VSSs which were recorded with a

[1] It is based on an idea of Elmar Nöth.

laryngograph [Alk92a]. This is a device which measures the impedance between two electrodes put on both sides of the neck of the speaker. This impedance is strongly correlated to the opening of the glottis. Such a device can, of course, not be used in ASU systems. Therefore, one has to automatically compute this signal from the speech signal by an inverse filter.

Linear filters were early used for this task [Hes83], and were refined by [Str93, Kie97]. These filters are based on the assumption that the vocal tract resembles a linear filter. However, in general this assumption does not hold. It has already been shown in other applications that NNs can perform non–linear signal processing or function prediction [Lap87, Wid88]. This motivated us to develop a filter using an NN. This approach has two advantages: It can be automatically trained, and it performs a non–linear mapping. This work has been described in detail in the master's and bachelor theses [Den92, Kah93] and was already published in [Den93, Nie94a]. We will give a short summary in the following.

In our approach we map the speech signal directly to the voice source signal without any prior coding or feature detection. The NN is trained with pairs of speech signals and VSSs obtained with a laryngograph. The speech signal is low–pass filtered at 1000 Hz; the VSS is band–pass filtered from 20 Hz to 1000 Hz to remove high frequency noise and low frequencies which result from larynx movement during swallowing; both signals are downsampled to 2000 Hz. Note that the low–pass filtering of the speech signal alone is not sufficient to remove the vocal tract information. The speech signal is normalized to the range of $[-1, 1]$ and the VSS is normalized to the range of $[0, 1]$. Input to the NN is a frame of the speech signal. The NN has one output node, whose activation is considered as one sample value of the VSS. The input frame is shifted point–by–point through the speech signal; this results in a sequence of output values, the concatenation of which yields the automatically computed VSS, henceforth NN–VSS. This VSS is further smoothed iteratively by five different average filters whose width depends on the average pitch period determined from the unfiltered VSS.

During training the desired output of the NN is set to the VSS sample value at the center of the input frame. Entire speech signals are used for training including unvoiced and silent parts. We used quick–propagation for training. The NN is optimized with the usual mean squared error function (2.21). However, we found that for the comparison of two different NNs the mean squared error is not suitable. It seems not to provide an absolute measurement of the quality of the mapping learned by the NN. We were not interested in the exact reconstruction of the shape of the original VSS by the NN but we wanted to achieve an NN–VSS optimally suited for F0 determination and laryngealization detection. Therefore, we use the following error measure for the judgment of the quality of an NN originally trained

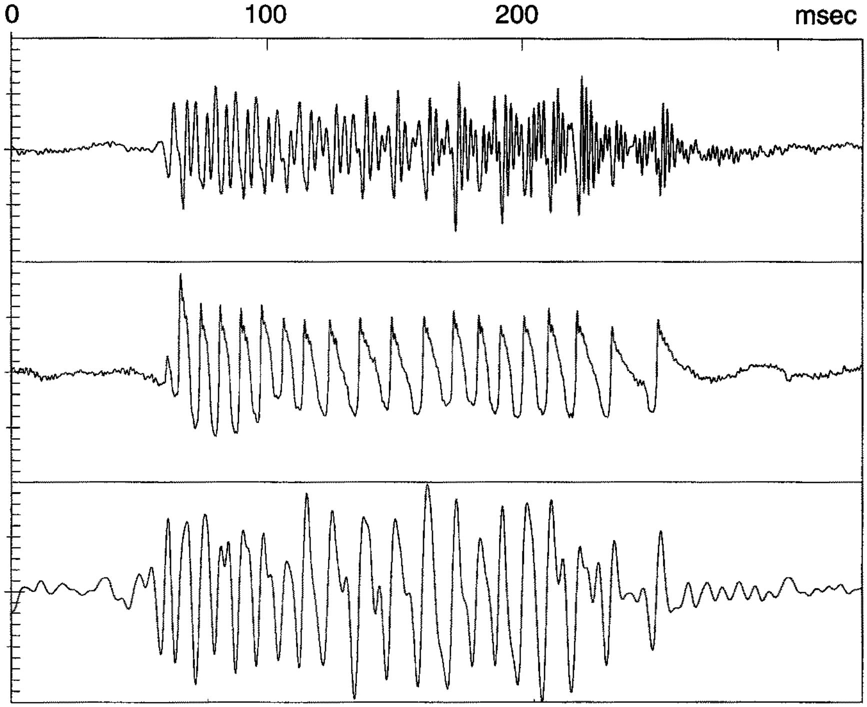


Figure 6.1: From top to bottom: speech signal, VSS determined by a laryngograph, VSS automatically computed by inverse filtering of the speech signal using an NN.

to minimize the mean squared error: for each 12.8 msec frame of the NN–VSS one F0 value is determined by a modification of the time–domain algorithm described in [Alk92a]. On an ideal noise–free VSS this algorithm computes the F0 without errors. Each of these F0 values is considered to be correct if it differs from a reference value by less than 30 Hz.

Speaker–dependent training and cross–validation was done using speech and corresponding VSS signals of 114 isolated time–of–the–day utterances (140 secs) from 3 male and 5 female speakers[2]. For training 35 utterances consisting of 40 sec of speech were used, that is, 68,620 training patterns. We evaluated the NNs on the entire data–base. We tried different NN topologies. Best results were achieved with a 39 msec input frame consisting of 78 sample values and using an NN consisting of four layers. Each of the hidden layers had 100 nodes. We yielded 3% F0 errors on the test data. Simple forms of recurrent NNs as described in [Jor86, Elm90]

[2] This database was kindly provided by Anton Batliner, L.M. Universität München.

were also tried resulting in 3.6% errors for NNs having the same number of parameters as the best NN. More general recurrent NNs as described in Section 2.2.4 were not tried. At the bottom of Figure 6.1 a typical example for an NN–VSS is shown. The corresponding speech signal is shown at the top. This is an example where it is not trivial to determine the F0 from the speech signal. There are strong higher frequencies which could be mistaken for the F0. Nevertheless, the result of the inverse filtering by the NN is very close to the VSS shown in the middle of the figure, which was determined by a laryngograph. Recall that the NN–VSS was smoothed by an average filter.

With the best NN we computed VSSs for the SPONTAN sample which is used in [Kie97] for the evaluation of F0 algorithms and for laryngealization detection, cf. also [Kom93a, Kie95, Bat95a] and Section 6.2. We computed F0 estimates by simply determining the maximum in the short–term spectrum of the NN–VSS. This approach yielded 10% less errors than the approach described in [Kie92] which is an older version of the algorithm described in Section 6.2. These NN–VSS of the SPONTAN sample were, furthermore, successfully used in [Gie95] for the detection of laryngealizations by an NN/HMM hybrid of the type presented in Section 2.7.1. Although we obtained promising results we did not use these NN–VSSs any further because the computation is more expensive than would be tolerable in an ASU system: since the NN has $(78+1)\cdot 100+2\cdot(100+1)\cdot 100+(100+1)=28,201$ weights and 301 hidden and output nodes, 28,201 multiplications and 301 evaluations of the sigmoid function have to be conducted per sample value of the speech signal. Real–time computation would require almost 60 MFLOPS, which is not possible with state–of–the–art workstations. However, this approach is promising if special hardware were available. In the future it might also be possible to adapt it for non-linear noise reduction or for the modeling of vocal tracts for the purpose of trainable feature extraction for speech recognition.

6.2 DPF0–SEQ: Incremental Fundamental Frequency Determination

Many algorithms were proposed for the determination of F0, for overviews cf. [Hes83, Nöt91a, Kie97]. These either compute an F0 estimate in the *frequency domain* by detecting the harmonics of the F0, or they directly try to find speech signal segments corresponding to fundamental periods. A main drawback of most of these algorithms is that F0 is computed only locally without taking into account context information. However, F0 errors can be considerably reduced when one makes use of the observation that neighboring F0 values do not differ very much, that is, the F0 contour is usually rather smooth. Therefore, an algorithm has been developed which searches for a smooth F0 contour in a matrix of alternative estimates computed over the time by dynamic programming (DP) [Bel57]. The algorithm is described in detail in [Kie97, Sec. 6.3.2]; preliminary versions of the algorithm have also been described in [Kom89, Kie90, Nöt91a, Kie92, Nie94c].

An overview of the basic DPF0–SEQ algorithm is given in Figure 6.2. The algorithm is based on the well–known fact that the frequency of the absolute maximum of the short–time spectrum of a voiced speech frame is a harmonic of the fundamental frequency; hence, this frequency divided by the fundamental frequency yields an integer. The problem is to find the correct integer divisor of the frequency of the absolute maximum. This problem is solved by determining several *candidate values* of the F0 for each frame and selecting the optimal ones by DP. The candidate values are determined on the basis of the average target value $F0'$ of the utterance and the maximum in the current frame's spectrum. The DP search minimizes the difference between two successive F0 estimates by keeping the resulting contour as close as possible to the target value $F0'_l$ of the voiced region. The target value for each of the voiced regions is computed by the combination of two standard approaches, a time–domain approach after [Ros74] and a frequency domain approach after [Sen78]. These are applied to a few frames in the part of the voiced region which has the maximum energy. It was observed that in these parts of the speech signal the F0 estimation is most robust. The DP algorithm searches for the path minimizing the weighted sum of the difference between consecutive candidates plus the distances of the candidates to a local target value. The path obtained in this way is the estimate of the fundamental frequency contour. It is usually even smooth at small laryngealized speech segments, which corresponds to the perceptual impression human listeners have. In practice we limit the range of F0 estimates to the interval from 35 Hz to 550 Hz. The F0 values are arbitrarily set to zero for unvoiced frames.

For simplicity the algorithm has been described in Figure 6.2 as if the utterance were first recorded and then processed as a whole. In fact the algorithm works on fixed sized increments of the speech signal thus allowing real–time F0 determination of speech signals of arbitrary length. The algorithm as described in the figure determines the F0 for each of the voiced regions in the increment and then moves on to the next increment. If an increment cuts a voiced region it is processed with the next increment. The average target value $F0'$ is dynamically updated using all target values $F0'_l$ of the past of the utterance.

This DPF0–SEQ algorithm was evaluated on 9 dialogs of the VERBMOBIL corpus, 24 minutes of speech, and on the SPONTAN sample data; it yields 4.7% and 6.0% coarse F0 errors, respectively. Recall that a coarse error is a difference

```
1. Preprocessing
Partition the digitized speech signal $f_j$ into frames $r_n$ of fixed size (e.g. 10 msec). The frames are numbered consecutively by the index $n \in \{1, \dots, T\}$.
For each frame perform a voiced/unvoiced decision; adjacent voiced frames are grouped to a voiced region $R_l$. Each voiced region is defined by an index tuple $(b_l, e_l)$ which gives the frame number of the beginning and end frame, respectively, of $R_l$. Between two consecutive voiced regions there is at least one unvoiced frame.
2. Short–time spectrum
FOR each voiced region $R_l$, $l = 1, \dots, L$ containing frames $r_n$, $n \in [b_l, e_l]$
    Low–pass filter the speech signal at 1100 Hz and perform a downsampling such that the sampling frequency $\geq$ 2200 Hz.
    An analysis window $s_n$ corresponding to a frame $r_n$ consists of the sample values in the three frames $r_{n-1}, r_n, r_{n+1}$.
    For each analysis window of a voiced frame compute the absolute value of the short–time spectrum $E_\nu$, $\nu = 0, 1, \dots, 127$.
    Determine the energy $Lh_n$ per frame from the spectrum.
3. Target values $\mathrm{F0}'_l$
FOR each voiced region $R_l$, $l = 1, \dots, L$ containing frames $r_n$, $n \in [b_l, e_l]$
    Determine a frame $r_\kappa$ for which F0 computation is assumed to be reliable:
    IF    $e_l - b_l + 1 \leq 5$
    THEN  $\kappa = (b_l + e_l + 1)/2$
    ELSE  Select $\kappa$ such that $Lh_\kappa = \max_{n \in [b_l+2, e_l-2]}\{Lh_n\}$
    Determine for this frame $r_\kappa$ the F0 by a standard algorithm. This F0 value is the target value $\mathrm{F0}'_l$ of the voiced region $R_l$.
Compute an average target value $\mathrm{F0}' = \frac{1}{L}\sum_{l=1}^{L} \mathrm{F0}'_l$.
4. Fundamental frequency candidates
FOR each voiced region $R_l$, $l = 1, \dots, L$
    FOR each frame $r_n$, $n \in [b_l, e_l]$, in voiced region $R_l$
        Determine the maximal value $E_{max}$ and the frequency $\xi_{max}$ of this value in the short–time spectrum $E_\nu$; set the integer divisor $n = \xi_{max}/\mathrm{F0}'$
        Five F0 candidates $F_{k,l,a}$ of frame number $k$ in voiced region number $l$ are defined by $F_{k,l,a} = \{\frac{\xi_{max}}{n+a}, a = -2, \dots, 2\}$; a candidate is undefined if $n + a \leq 0$
5. Fundamental frequency contour
FOR each voiced region $R_l$, $l = 1, \dots, L$
    Compute the optimal path in the matrix of F0 candidates by dynamic programming so as to minimize the distance between succeeding F0 values and to the target value $\mathrm{F0}'_l$
    Retrieve the F0 contour by back–tracking
```

Figure 6.2: Computation of the fundamental frequency (F0) contour after [Nie94c], adapted with respect to the DPF0–SEQ algorithm.

of 30 Hz from the reference. It is measured on the frames which are determined as voiced by both reference and automatic classifier. The voiced/unvoiced classifier is based on a few simple rules which compare energy and zero crossing features with thresholds. The thresholds were optimized with a coordinate descent algorithm and showed better results than automatically trained statistical classifiers [Str93]. On the VERBMOBIL data the voiced/unvoiced classifier yields 4.5% errors.

In the remaining chapters we are only interested in the coarse F0 contour. As the above error rates show this can be determined very robust with the DPF0–SEQ algorithm. The fine shape of the contour is not very informative with respect to accentuation and boundaries, cf. Section 4.1.2. In certain cases, for example in pitch synchronous feature extraction for speech recognition, cf. [Fla92], the exact F0 values and not only a frame–wise F0 determination but a determination of fundamental periods is necessary. For this in [Har94] an algorithm was developed, cf. also [Har95, Kie97]. It has been shown that this fundamental period determination can be done much more reliable if the robust estimates from the DPF0–SEQ algorithm are taken as a basis to restrict the range of the fundamental periods.

6.3 Classification of Sentence Mood

Classifiers for intonational sentence mood were developed to be able to determine the dialog act in the EVAR system in the case of isolated time–of–day expressions as described in Section 8.6.2. The three classes *fall, rise*, and *continuation–rise* were distinguished, cf. Section 4.1.2. They correspond to statement, question, and feedback. Earlier experiments using normal density classifiers were described in [Ott93] and published in [Kom93b, Nie93, Nie94c, Nie94b, Nie95, Kom94b]. Meanwhile better results were achieved by an improved feature set and using NNs [Kie97]; we will summarize these results in the following.

The experiments were performed on the TIME–OF–DAY sample data. Different features computed over a varying number of voiced regions were tried; the best result was obtained with the nine features illustrated in Figure 6.3 and listed in Table 6.1. The values of a regression line are given by

$$y_n = Rc \cdot t_n + C \tag{6.1}$$

The regression coefficient Rc and the constant C are determined over the non–zero F0 values in the considered time interval. The best NN topology was an NN with nine nodes in the first hidden layer, six nodes in the second hidden layer and three output nodes. Each output node corresponded to one of the classes; the desired output for the correct class of a training pattern was set to one, the other

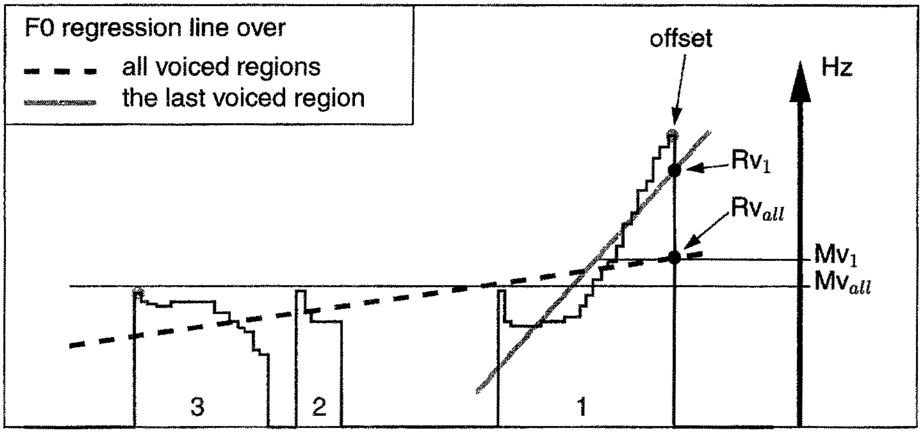


Figure 6.3: Illustration of the features for the classification of sentence mood obtained from the F0 contour, after [Kie97]. The F0 in this example is rising at the end of the utterance and, hence, corresponds to a question.

desired outputs were set to zero. Training and evaluation was done in a *leave–one–speaker–out* mode: three speakers were used for training, the fourth one was used for testing; this procedure was repeated so that each of the speakers was used once for testing. The results for the different classes are given in Table 6.2; they are the average of the recognition rates obtained for the individual speakers. The overall recognition rate (RR) is 92.9%. It turns out that falling F0 can be classified most reliably; continuation–rise shows the worst results. This is due to the fact that it is somewhere between the other two classes.

In the context of the Danish flight ticket reservation system [Dal94a, Bae95] this classifier for the sentence mood has been successfully adapted to Danish speech using also the algorithm for F0 computation as described in Section 6.2 [Brø96].

Note that the features as used in this classifier were especially developed for isolated time–of–day expressions as they occur frequently in train time table dialogs, cf. Section 8.6.2. Although this classifier has also been integrated in the VERBMOBIL research prototype the features are not optimal in this context because the utterances vary considerably in length, they often consist of more than one clause, and the classifier is not robust with respect to accentuation, especially, of a word in the final voiced region. However, since sentence mood classification so far is only used within EVAR but not yet within the linguistic processing of VERBMOBIL, further research was postponed until the second phase of VERBMOBIL, cf. also the investigations presented in [Dal92, Bat93c, Kom93a, Bat95a]. In Sections 7.3 and 8.6.2 we refer to the classifier described in this section. In these

feature	description
1. Rc_{all}	coefficient of the regression line over all F0 values in the utterance
2. offset–Rv_{all}	difference between the F0 offset and the value of the regression line (computed over the entire utterance) at the offset position
3. offset/Rv_{all}	quotient between the F0 offset and the value of the regression line (computed over the entire utterance) at the offset position
4. Mv_{all}–Mv_{all}	difference between the F0 average in the last voiced region and the F0 average in the entire utterance
5. Mv_{all}/Mv_{all}	quotient between the F0 average in the last voiced region and the F0 average in the entire utterance
6. offset–Mv_{all}	difference between the F0 offset and the F0 average in the entire utterance
7. offset/Mv_{all}	difference between the F0 offset and the F0 average in the entire utterance
8. Rc_1	coefficient of the regression line over all F0 values in the last voiced region
9. offset–Rv_1	difference between the F0 offset and the value of the regression line (computed over the last voiced region) at the offset position

Table 6.1: Intonational features for sentence mood classification, cf. also Figure 6.3.

		recognized class		
reference class	#	rise	fall	continuation–rise
rise	97	90.7	1.0	8.2
fall	118	0.0	98.3	1.7
continuation–rise	107	6.5	4.7	88.8

Table 6.2: Recognition results for the sentence mood using an NN in leave–one–speaker–out experiments (RR = 92.9%), after [Kie97].

cases we used the NN and the features as described above, however, the training was conducted on the entire TIME–OF–DAY corpus.

6.4 Recognition of Phrase Boundaries and Accents

In this section we describe the phrase boundary and accent classifiers, which are integrated in the approaches described in Chapters 7 and 8. The goal is

- to classify syllables or words as *accented* or *unaccented* and
- to assign word boundaries a prosodic boundary class.

They were developed by Andreas Kießling, a detailed treatment can be found in [Kie97, Secs. 7.3, 8.4], cf. also [Kom94a, Kie94b, Kie96c, Mas96]. In Sections 6.4.1 to 6.4.3 we will summarize the approach and the results. As classifiers NNs are used; in Section 6.4.4 we will therefore discuss a few aspects concerning the further use of the NN output activations in ASU systems.

6.4.1 General Considerations

Similar to the methods developed by the group of Mari Ostendorf [Wig92a, Wig94] in the approach described in the following features are based on the time–alignment of the phones corresponding to the standard pronunciation of the spoken or recognized words. The results described in this section were obtained using the spoken word chain to exclude the influence of word recognition errors. In Section 8.1 we show how the classifiers trained on the spoken word chain can be used on the basis of word hypotheses graphs.

Other prosodic accent or boundary classification experiments have been described where the classification is conducted for each 10 msec frame [Tay95] or on the basis of syllables detected in the energy contour without the help of phone models provided by a word recognizer [Nöt88b, Str95, Str96]. The use of the time–alignment of the words has a number of advantages:

1. Syllable or syllable nucleus detection can most reliably be done by using the information contained in the word models of a speech recognizer.
2. Not only syllable or syllable nucleus boundaries can be used, but the speech signal is segmented into phones.
3. As discussed in Section 4.1.2 durational information should be normalized by phone intrinsic parameters; this can only be done when using the word information.
4. When the time alignment of a word chain is used during training, the reference labels have only to be given word–by–word, that is, the position of the

labels on the time axis does not have to be provided by the labeler, which would increase the labeling effort, cf. Section 5.2.5.

5. In an ASU system a linguistic module should get the information if a specific word hypothesis is accented or not or if it is succeeded by a boundary. It is not very helpful in this context to know that a certain speech segment is accented, if this segment for example overlaps with word boundaries computed by a word recognizer.

The only disadvantage of using the time–alignment of the word chain is that in few applications this information might not be available. However, in the context of the VERBMOBIL system word information is available and therefore should be used. The time–alignment of the phones corresponding to a word is done by a standard HMM word recognizer, in our case the one described in Section 3.1. The word HMMs are built by concatenation of (context–dependent) phone HMMs; these are concatenated according to the spoken word chain underlying an utterance to yield the utterance HMM. The Viterbi beam search is used to determine the optimal state sequence, from which the segmentation of the speech signal into phones, pauses, and non–verbals can be derived. The phone segmentation also defines the syllable and word boundaries. Examples are given in Figures 4.3 to 4.5.

In [Cam92] the difference of hand–segmented phoneme boundaries and those determined by HMMs was examined. He found that 80% of the boundaries were less than 25 msec apart and that 60% were less than 15 msec apart. The comparison of the boundaries created by two different experts showed about the same results. Other researchers got similar results when evaluating the phone segmentation computed with an HMM word recognizer [Fer95].

6.4.2 Feature Extraction

In Section 4.1.2 we showed for a few examples depicted in Figures 4.3 to Figures 4.5 that accentuation and boundaries can be marked by a change in duration and F0 and we indicated that energy might play some role. In this context, a change in these prosodic attributes over a time interval corresponding to a few syllables or words is relevant. It is not convenient to use the whole F0 or energy contours over such intervals as input for classifiers; therefore features had to be found which encode the relevant information. A flexible tool was developed which allows to compute a large amount of features over different time intervals around a syllable, word, or boundary to be classified. For each task the optimal feature set was determined; therefore, the classifiers used in the remainder of this book all differ in the exact set of features used as input. Each of the features can be determined over a time interval specified in terms of numbers of syllables, syllable nuclei, or words to the left or to the right of the syllable or boundary to be classified. If the time interval covers several syllables but is restricted to syllable nuclei, only the syllable nuclei segments in this time interval are considered. Fixed time intervals specified in msec were used in preliminary experiments with less success. In Section A.4 the exact set of features for one particular classifier is given; Table 6.3 gives an overview of the principal types of the features.

The speaking rate is used as a feature itself and for the normalization of the phone durations. It can either be determined globally, that is, on the entire utterance, or in a local interval around the syllable/boundary to be classified specified in terms of $\pm n$ syllables. Although speaking rate can vary to a great extent within utterances, cf. Figure 4.3, in preliminary experiments no significant difference was found between both variants, presumably, because the determination of the speaking rate in a fixed sized local interval is suboptimal.

The phone duration is either determined in msec or it is normalized according to phone intrinsic mean and standard deviation and according to the speaking rate as proposed by [Wig94], cf. Section A.2. Context–dependent phone parameters can be used for the normalization, where the context information is either if the syllable carries the lexical word accent or additional if the syllable is the first, the final, or any other syllable in the word. In the figures in Section 4.1.2 the context–independently normalized duration of the syllable nuclei is depicted.

The energy is computed as shown in Section A.1. It is, furthermore, for each utterance linearly normalized to a fixed range. The mean, median, and maximum energy values can be phone intrinsically normalized similarly to the phone durations, except that the speaking rate is not considered. Note that the positions of the energy maxima mainly encode durational information.

The F0 is linearly interpolated in unvoiced regions and extrapolated from the onset and offset to the beginning and end of the utterance, respectively. This is useful because the voiced/unvoiced decision is independent from the word recognizer. With respect to the computation based on word graphs, cf. Section 8.1, it is more convenient to have an independent voiced/unvoiced decision. The F0 values are given in semi–tones, which is essentially the logarithm of the F0 measured in Hz. The semi–tones are normalized by the subtraction of the average semi–tone in the utterance or by the speech signal increments considered so far.

If a feature is computed over the syllable preceding the syllable to be classified, and the latter is the first syllable in the utterance, the specified time interval is undefined. In this case default values are used.

Related work has been described, for example, in [Cam94, Wig94, Str95,

DURATIONAL FEATURES
the *speaking rate*
the average phone *duration* in the interval

F0 FEATURES
the *mean* and/or *median* F0 in the time interval
the *minimum, maximum, onset* and/or *offset* F0 values in the time interval
the *positions* of these F0 values on the time axis relative to the end of the syllable/word to be classified or preceding the boundary
the *regression coefficient* of the F0 contour in the time interval
the root of the mean of the squared differences between the F0 values and the respective values of the regression line

ENERGY
the *mean* or *median* energy in the time interval
the *maximum* energy value in the time interval
the *position* of the maximum energy values on the time axis relative to the end of the syllable/word to be classified or preceding the boundary
the *regression coefficient* of the energy contour in the time interval
the root of the mean of the squared *differences* between the energy values and the respective values of the regression line

PAUSE INFORMATION
the *length* of the pause in msec before or after the syllable or word; if there is no pause the length is defined as being zero

LEXICAL FEATURES
an integer identifying the *class of the phone* being in syllable nucleus position
a flag indicating whether the syllable carries the *lexical accent*
a flag being one if the syllable is in *word final* position, zero else

Table 6.3: Types of features used for accent and boundary classification

Tay95], however, we believe that the prosodic feature extraction developed by Andreas Kießling [Kie97] and used within the research presented in this book is most flexible and exhaustive. It especially takes a large context into account.

6.4.3 Classification

For training and testing of classifiers for each syllable or word in an utterance a feature vector was computed consisting of features as described in Section 6.4.2. Initial experiments were conducted on the ERBA corpus. In a large series of experiments the features described in Section 6.4.2 were developed and their suitability for this classification task was evaluated. Later the approach was adopted for the VERBMOBIL corpus. In the following we will summarize the results mainly reported in [Kie97] as far as they are relevant for the remainder of this thesis. NNs were used as classifiers; they were trained on sample data which consisted of the same amount of patterns for each class. This was achieved by not using all of the available patterns for frequent classes and by using the patterns of rare classes more than once, cf. the discussion in Section 6.4.4. In general word boundary labels were assigned to the word or syllable preceding the boundary. In the following, boundary recognition rates never take the end of turns into account because their classification is trivial. In classification we always decide for the class corresponding to the output node with the maximum output activation.

Results on ERBA

Best results were achieved with an NN for integrated accent and boundary classification, which is not surprising because both prosodic attributes influence each other, cf. Section 4.1.2. For this task the boundary and accent labels described in Sections 5.1.2 and 5.1.3, respectively, were combined to classes like A[23]+B[01], which for example means that the syllable was originally labeled with either A2 or A3 with respect to the accents and with either B1 or B0 with respect to the boundaries. Each syllable in the corpus is assigned one of the following six reference classes:

- A[01]+B[01]: unaccented syllable not succeeded by a phrase boundary,
- A[01]+B2: unaccented syllable succeeded by a constituent boundary,
- A[01]+B3: unaccented syllable succeeded by a clause boundary,
- A[23]+B[01]: accented syllable not succeeded by a phrase boundary,
- A[23]+B2: accented syllable succeeded by a constituent boundary, and
- A[23]+B3: accented syllable succeeded by a clause boundary.

Syllables not being in word–final position are implicitly labeled as B0. Syllables in word–final position get the label of the succeeding word boundary. Although we

distinguished classes	RR	CRR
B[01]/B2/B3	90.3	90.6
A[01]/A[23]	94.9	95.6

Table 6.4: Recognition rates in percent on ERBA where three boundary classes and two accent classes are distinguished, after [Kie97].

ultimately are interested in either boundary or accent classification but not in the distinction of these six classes, in preliminary experiments it was found that such an explicit clustering of the training data improves the results.

An NN is used, where each of the six output nodes corresponds to one of these labels, and which had 60 nodes in the first and 30 nodes in the second hidden layer. The feature vector computed for each syllable contained 143 syllable or syllable nucleus based features obtained from an interval of ± 6 syllables around the syllable to be classified.

The NN was trained on ERBA_TRAIN and evaluated on ERBA_TEST. The recognized class is the one which corresponds to the output node with the maximum activation. Accentuation and boundary recognition was evaluated separately. The recognition rates (RR) and the average of the class–wise recognition rates (CRR) are given in Table 6.4. The recognition rates for accents were determined on all syllables; the boundary recognition rates were only determined on the word final syllables not taking into account the turn–final syllables. It turns out that the different classes can be reliably distinguished. This result also justifies our method for the generation of the reference labels as described in Sections 5.1.2 and 5.1.3. A separate evaluation of the different labels showed that for both accent and boundary detection the durational features are most important; F0 also plays an important role, whereas the influence of energy and pause is negligible [Kie94b]. Pauses are not relevant for this corpus because due to the recording conditions utterance internal pauses occur rarely and not always at boundary positions but sometimes the speakers were simply breathing at non–boundary positions.

Boundary classification on VERBMOBIL

Due to the smaller amount of VERBMOBIL training data it was found useful to train separate classifiers for accent and boundary classification. We first will consider the boundary classification; as for accents, cf. below. NNs with one output node per class were used as classifiers. In the following, NNs are characterized by "[M : H_1 : H_2 : N]" where M is the number of input nodes, H_1 and H_2 give the number of nodes in the first and second hidden layer, respectively, and N is the number

training labels	NN topology	*RR* (*CRR*) using as reference			
		B3/B[029]	M3/M0	MB3/MB0	D3/D0
B3/B[029]	[274:40:20:2]	**88.3 (86.8)**	86.4 (82.1)	**85.4 (81.6)**	83.3 (82.0)
M3/M0	[121:100:50:2]	81.5 (85.3)	**86.0 (86.7)**	**82.2 (83.3)**	78.1 (82.0)
D3/D0	[117:60:30:2]	76.6 (84.8)	88.1 (83.6)	87.4 (82.7)	**85.1 (82.6)**

Table 6.5: Recognition rates in percent obtained with different NNs for boundary classification, after [Kie97, Kie96b, Mas96]. The average of the class–wise recognition rates CRR is given in parentheses.

of output nodes. Using more than three layers did not improve the results. One feature vector was computed for each word.

For this thesis, in particular for the use in the VERBMOBIL system, we are mainly interested in the recognition of clause boundaries. In Section 5.2 we defined different types of clause boundaries:

- B3: (perceptual) prosodic clause boundaries,
- M3: prosodic–syntactic clause boundaries, which are strongly related to the B3 boundaries, and
- D3: dialog act boundaries, which are more or less a subset of M3.

For the relationship of these boundaries cf. Section 5.2.7. Depending on the task one is interested in the recognition of either the B, the M, or the D labels. Since the features of Section 6.4.2, which are used here as input to the classifiers, are acoustic–prosodic features except for the flags, it seems to be convenient to train classifiers on the prosodic B labels, regardless what type of clause boundary is to be recognized. Therefore, we train a classifier on the B labels and evaluate it on all of these types of labels. When training a classifier on the M labels one has to be aware of the fact that quite a percentage of M3 is not strongly prosodically marked, that is, they do not correspond to a B3 boundary, cf. Table 5.19. Therefore, a classifier trained on the B labels might be better suited to classify M3 boundaries. On the other hand, a classifier trained on the M labels might achieve better results even in recognizing the B labels, because of the much higher amount of training data. Similar considerations hold for the D labels.

Because of these considerations, three NNs are compared in Table 6.5; they are trained to distinguish between B3/B[029], M3/M0, or D3/D0 (indicated on the left–hand side of the table) and were trained on the sub–corpora BS_TRAIN, M_TRAIN, and D_TRAIN, respectively. For each type of classifier the optimal feature set and

NN	*RR*(B0)	*RR*(B2)	*RR*(B3)	*RR*(B9)
[274:40:20:4]	75.2	51.6	64.0	75.0

Table 6.6: Recognition rates in percent for the discrimination of four boundary classes on VERBMOBIL.

NN topology has been determined independently. In any case it was not found useful to consider more than ± 2 syllables and ± 2 words. Table 6.5 shows the recognition rates in percent determined on the sub–corpus BS_TEST evaluated on all word boundaries except the end of the turns. The different sets of reference labels are shown at the top of the table. The NN for M3/M0 was not trained on the ambiguous boundaries MU. However, with respect to the integration of these NNs in ASU systems a decision has to be made also on these boundaries, or a likelihood for the presence of a boundary has to be given. Therefore, the second to last column shows results on all word boundaries using the following compound reference labels:

- MB3: word boundaries either labeled with M3 or with MU and B3
- MB0: word boundaries either labeled with M0 or with MU and B[029]

Since the NNs are trained with the same number of training patterns per class and with respect to their use in Chapters 7 and 8, cf. the discussion in Section 6.4.4, we consider the average of the class–wise recognition rates (CRR) to be more important; it is given in parentheses in Table 6.5. In general, the NN trained to distinguish a certain type of labels also achieves best results in distinguishing these labels. Altogether best and almost the same boundary recognition results are obtained for B3/B[029] and for M3/M0 (CRR: 87%); D3/D0 can be recognized correctly in 83% of the cases. It could not be expected a priori that M3 and M0 labels can be as well distinguished as the perceptual prosodic B labels. We consider, therefore, the M labels as useful for the training of acoustic–prosodic boundary classifiers as the B labels. This proves that the labeling scheme developed in Section 5.2.5 is adequate, because it was expected that most of the M3 boundaries are prosodically marked and vice versa most of the M0 boundaries are not marked; therefore, the recognition rates could expected to be worse than for the B labels if the labels or the scheme were inconsistent. Recall that the Ms are much faster to label than the perceptual B labels because they are solely based on the transliteration of utterances. Note that the M3/M0 labels became available only recently so that these recognition rates might even be improved in the future.

In [Kie97, Kie96c] the relevance of the different types of features was investigated. In contrast to the (often monotonously) read speech of ERBA, on the VERB-

NN	trained on	RR	CRR
[179:80:40:2]	words	82.8	82.4
[71:40:20:2]	syllables	78.4	78.1

Table 6.7: Recognition rates in percent for the discrimination of accented and unaccented words on VERBMOBIL.

MOBIL data F0 features are more important than duration. Pauses contribute more to the recognition than in the case of ERBA but as expected by the evaluation presented in Section 5.2.8 pauses play only a minor role.

For comparison and because it will be used in Section 7.1, we also trained an NN to discriminate the four classes B0/B2/B3/B9. We used the same feature set as for the B3/B[029]–NN. The class–wise recognition rates are given in Table 6.6. The average recognition rate (RR) was 71.5%. The average of the class-wise recognition rates (CRR) was 66.4%.

Accent classification on VERBMOBIL

Andreas Kießling performed preliminary accent classification experiments on VERBMOBIL as in the case of ERBA using syllable based feature vectors and labels [Kom95b]. However, since the semantic module in VERBMOBIL so far is mainly interested in the information whether a word is accented or not, in the experiments reported in [Kie97, Kie96c] for each word one feature vector was computed and used for NN training and testing. The NN has two output nodes, one corresponding to UA (unaccented), and the other one corresponding to PA or NA or EK (accented), which is abbreviated as A. The NN was trained on BS_TRAIN, results on BS_TEST are given in Table 6.7. The 179 features were obtained on ± 2 syllables and ± 2 words around the word final syllable of the word to be classified. Taking into account more context did not improve the results. Specifying the intervals for feature determination with respect to the syllable carrying the word accent was not yet considered. This NN yielded a recognition rate of 82.8%. For comparison the table also shows the results for the NN trained on all syllables, but evaluated on the word level. This word level evaluation was conducted by classifying the word as accented if any of its syllables was classified as accented. The recognition rate on syllables for this NN was 82.5%; it drops considerably to 78.4% when evaluated word–wise.

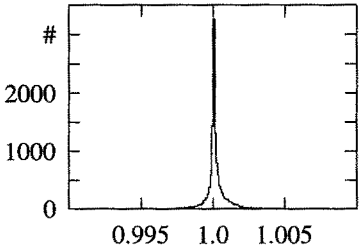


Figure 6.4: Distribution of the sum of the output activations of an NN. Abscissa: the different values of the sum of the output activations; ordinate: their frequency of occurrence.

6.4.4 Considerations Regarding the Integration of NNs in ASU

First boundary recognition experiments were reported in [Kom94a]. It was shown that polynomial classifiers outperform classifiers modeling the features with mixtures of multivariate normal distributions[3]. Note that most of the features described in Section 6.4.2 are not normal distributed. In [Sch93] it was found that also principal axis transformation or linear discriminant analysis do not improve the modeling with mixtures of normal densities. Later we found that with NNs about the same results can be achieved as with polynomial classifiers. With polynomial classifiers more features or cubic terms for all of the features could not be used due to numerical problems. Also with normal distribution classifiers the results could not be improved by using more features. In contrast, the recognition rates could constantly be improved using NNs (from 72% to 90% for the three boundary classes on ERBA) since the first results were published in [Kom94a].

For the integration of a classifier in an ASU system it is important that no hard decisions are conducted by the classifier. It should rather compute scores, and moreover, these scores should be probabilities so that they can be consistently combined with the probabilities computed by other knowledge sources. Polynomial classifiers compute scores which usually are not limited to the interval from zero to one [Kil93]. As discussed in Section 2.2.3, NNs in theory compute a posteriori probabilities, however, in practice it depends on the amount of training data available, on whether the NN is sufficiently large, and if the error converged to a global minimum during training. Since the above mentioned results indicate that NNs are best suited for the task of prosodic boundary classification, we examined the output activations of a particular NN in order to get some indications whether we can consider these as approximations of probabilities.

[3] The experiments with polynomial classifiers were conducted by Ute Kilian and Peter Regel–Brietzmann from Daimler Benz, Ulm.

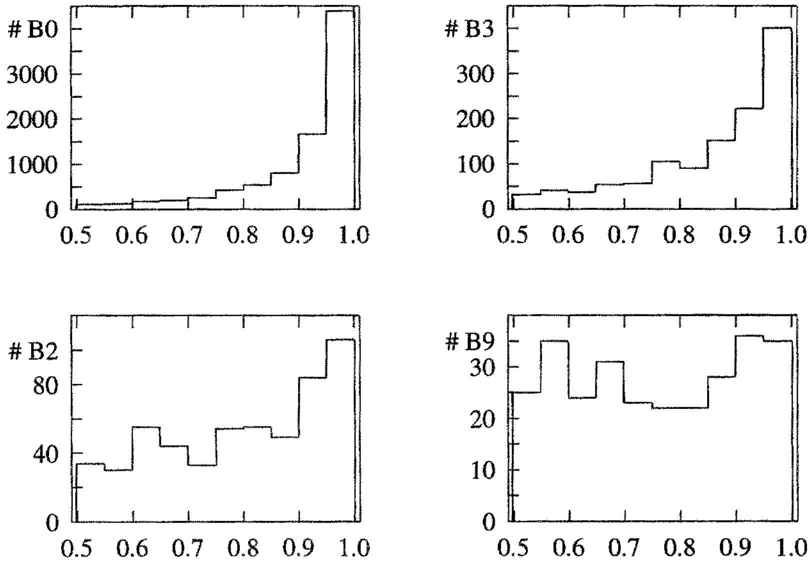


Figure 6.5: Distribution of the maximum of the output activations of the B3/B029]–NN for different reference labels; only correctly classified patterns are considered. Abscissa: maximum output activation; ordinate: frequency of occurrence for input feature vectors belonging to the indicated class.

The NN used is the B3/B[029]–NN whose recognition results are depicted in Table 6.5. In the following we only consider the outputs for correctly recognized patterns and do not take into account the end of turns. The evaluation depicted in the figures are based on the VERBMOBIL sub–corpus BS_TRAIN; separate evaluation on BS_TEST shows similar results. First of all we computed the sum of the output activations for each input pattern. The frequency of occurrence of the different values is shown in Figure 6.4. It can be seen that the sum of the values usually is very close to one although we use the normal sigmoid activation function and not the *softmax* function [Bri90b] which forces the sum of the outputs to be one.

Next, we considered the node with maximum output activation for each input pattern still using the B3/B[029]–NN; note that this NN has two output nodes so that maximum output activation is between 0.5 and 1. In Figure 6.5 the frequency of occurrence of these values is plotted separately for the different boundary classes. Again only correctly classified patterns were taken into account. It can be seen in the figures that in the case of B0 or B3 reference labels the NN is more "sure" about the decision, that is, values close to one are more frequent, than in the

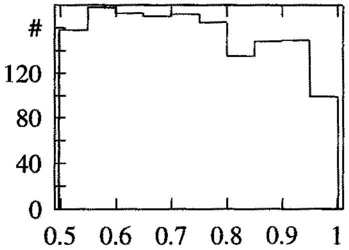


Figure 6.6: Distribution of the maximum of the output activations of the B3/B029]-NN for misclassified patterns. Abscissa: The different values of the output activations; ordinate: their frequency of occurrence.

case of B2 or B9. This is due to the fact that B2 is a weak boundary such that the features should be in between the class centers of B3 and B0. The irregular boundary B9 is usually even stronger prosodically marked than B3 but it is put into one class together with B0.

Finally, in Figure 6.6 the distribution of the maximum activation of the misclassified patterns is given. It turns out that the NN tends to be uncertain about the decision in these cases. A high probability is seldom assigned to a wrong decision.

From this analysis it can be concluded that NN output activations close to 1.0 can be interpreted as "very likely" whereas values close 0.5 seem to indicate that there is also some likelihood that the pattern would belong to the other class. For example, consider the case of a B2 input pattern, which causes an output activation of 0.7 at the node corresponding to B[029]. This means that this pattern is most probably not B3, but that there remains some likelihood that it could be B2. To conclude, we believe that this indicates that the NN output activations can be used as approximations of probabilities; we performed a similar analysis for a different application of NNs in [Kah93].

In our application it was observed during preliminary experiments that the a priori probabilities of the different classes influence the recognition results of the NNs too strong. Therefore, the training samples were designed such that the same number of patterns of each class was included. This has the consequence that the NNs compute probabilities $p(\boldsymbol{c}|\Omega_\kappa) = p(\Omega_\kappa|\boldsymbol{c})$. This is exactly what we need for the approaches developed in Sections 7.2, 7.3 and 8.1 where the NN probabilities are combined with a priori probabilities computed with a polygram model. The resulting classifier then estimates the a posteriori probabilities $p(\Omega_\kappa|\boldsymbol{c})$.

6.5 Summary

Using prosodic information in ASU requires preprocessing, feature extraction and classification. In the remainder of this book we use the methods developed by Andreas Kießling.

Basis of the F0 determination is the frame–wise computation of alternative candidates. These are appropriate divisors of the frequency with the maximum amplitude in the short–term spectrum of the speech signal. The F0 contour is determined for each voiced region by dynamic programming. It minimizes the difference between successive F0 estimates under the restriction that the resulting contour should be as close as possible to a target value, which is computed prior to the search with a robust time consuming standard algorithm.

Since F0 is a voice source phenomenon it is easier to directly determine the F0 from the voice source signal. In normal applications the voice source signal is not available. Therefore we developed a new method for the inverse filtering of speech signals by NNs. Input to the NN is a 39 msec frame of the speech signal. Output is one value of the VSS signal. The NN input frame is shifted point-by–point through the speech signal. The NN is trained with pairs of speech and voice source signals recorded in parallel. It was shown that F0 computation on this automatically computed voice source signal is very reliable (3% coarse errors), however, the method is too inefficient as to be used in an ASU system if no special hardware is available.

The sentence mood classification is based on features determined from the F0 contour itself and different regression lines. Three classes of sentence mood were distinguished: fall (statement), rise (question), and continuation–rise (feedback). On the TIME–OF–DAY corpus a recognition rate of 92.9% with an NN trained and tested in leave–one–speaker–out mode was obtained.

The classifiers for phrase boundaries and accents are based on features determined from the time–alignment of the (for the moment spoken) word chain. This has as main advantage that no manually time aligned reference labels are needed and that phoneme intrinsic parameters can be used for normalization. For each syllable or word to be classified a large feature set is computed consisting of phone durations, characteristic values obtained from F0 and energy contour, regression coefficients of these contours, pause length and flags indicating whether a syllable carries the lexical accent and whether it is in word final position. The same type of features is computed over different time intervals in the context of the syllable or word to be classified. NNs are used as classifiers.

On the ERBA corpus consisting of read speech, three prosodic–syntactic boundary classes on the word level and two prosodic–syntactic accent classes on the syllable level were distinguished; the recognition rate is 90.3% and 94.9%, respectively. The word–based recognition rate on VERBMOBIL for two prosodic accent classes is 82.8%. The increase in errors might be due to the lesser amount of training data and to the fact that spontaneous speech is not as regular as read speech.

As for the boundaries in the VERBMOBIL database, we were mainly interested in the detection of clause boundaries. Therefore, NNs were trained on the prosodic (Bs), the prosodic–syntactic (Ms), and the dialog act (Ds) boundaries. It turns out that both M3/M0 and B3/B[029] can be distinguished at a recognition rate of 87%; the recognition rate for D3/D0 is 83%. The NN trained on M3/M0 recognizes M3/M0 better than the NN trained on the perceptual B labels. These results show that the M labels are as useful for the training of classifiers as the B labels and, furthermore, that the labeling scheme we presented in Chapter 5 is adequate. Recall that the Ms are much faster to label than the perceptual labels because they are solely based on the transliteration of utterances.

With respect to the integration of these NN classifiers we investigated the distribution of the sum and the maximum of the output activations of an NN given a correctly classified feature vector. The NN was trained to distinguish between B3 and B[029]. We found that the sum only marginally differs from one. The maximum activation was separately evaluated for the different reference boundary labels; it is often close to one for B3 and B0 whereas the activations for B2 and B9, as expected, reflect an uncertainty about the class membership. This together with the theoretically findings summarized in Chapter 2 leads us to the assumption that these NNs approximate probabilities.

Chapter 7

Prosodic Phrase Models

As outlined in Section 4.1.2 the different prosodic attributes influence each other. Therefore, it seems to be promising to model entire phrases or at least to capture a certain context instead of classifying single isolated events as done by the NNs described in Section 6.4. Of course, these NN classifiers already take some context into account, because the feature vectors are computed over a few syllables or words in the context. However, the amount of context which can be considered in this way is very limited, and, furthermore, it might be better to extend the consideration of context to the level of decisions (classes).

With respect to this we developed different approaches, which will be presented in this chapter. However, it will turn out that these higher level models only yield satisfying results if they are based upon the NN classifiers of Section 6.4. As a first step we investigated NN/HMM hybrids (Section 7.1). The NNs model local information whereas the HMMs are used to model entire phrases. In another approach (Section 7.2), we developed a method for the combination of NNs and polygrams for the classification of prosodic attributes. In this approach the NN constitutes the speech (acoustic) model and the polygrams predict events on the basis of the spoken words, thereby being the language model. Finally, MSCTs were used for an integrated model of speech and language incorporating arbitrarily large context (Section 7.3).

With the currently available training data, the combination of NN and polygrams yielded the better results than the other approaches. This NN/polygram classifier was then used in the VERBMOBIL prototype system for the scoring of word graphs, cf. Sections 8.1 and 8.3.2.

phrase class	structure	# in ERBA_TRAIN
Phr1–short Phr1–long	(A[01]+B[01])* A[23] (A[01]+B[01])* B2	6,890 4,462
Phr2–short Phr2–long	(A[01]+B[01])* A[23] (A[01]+B[01])* B3	4,710 3,967
Phr3–short Phr3–long	(A[01]+B[01])* A[23]+B2	2,681 1,562
Phr4–short Phr4–long	(A[01]+B[01])* A[23]+B3	563 333

Table 7.1: Different phrase classes identified in the ERBA corpus. The four types Phr1 to Phr4 are distinguished by their structure. These types are further subdivided depending on the length of the phrases.

7.1 NN/HMM Hybrids

HMMs are widely used in ASR for the modeling of time sequences of feature vectors. For example, continuous speech recognition is done with word HMMs which generate an observation consisting of acoustic features every 10 msec. Therefore, it is obvious to establish prosodic phrase models by HMMs; for a definition of prosodic phrases cf. page 107. In our approach each particular class of prosodic phrases is modeled by one HMM. Continuous speech recognition, cf. Section 2.1, page 21, in this context means the modeling of sequences of different classes of phrases. However, we are not interested in the recognition accuracy on the phrase level, but we want to know how well we can recognize phrase boundaries and accents with the help of these models. This can be achieved by associating with each of the HMM transitions a boundary and accent class; during recognition a Viterbi search for the best transition sequence results in the sequence of recognized boundary and accent classes. The observations are feature vectors, where one vector corresponds to a syllable or a word. This approach was developed in [Kah94] using the ERBA corpus. In the following we will summarize this work, and in addition we will give results for the VERBMOBIL domain.

The Approach

First experiments were conducted using the sub–corpora ERBA_TRAIN and ERBA_TEST. The classes we intend to distinguish are A[01]+B[01], A[01]+B2, A[01]+B3, A[23]+B[01], A[23]+B2, and A[23]+B3. These are the same classes as used for the experiments described in Section 6.4, which were also done on ERBA.

Phr1: X X X X A2 X B2
Ich möchte in Bielefeld zwischen ...
(I want in Bielefeld between ...)

Phr2: X X X X A3 X X X X B3
Ich möchte in Bielefeld abfahren.
(I want in Bielefeld to leave.)

Phr3: X X X X A2+B2
Gibt es einen Zug nach ...
(Is there a train to ...)

Phr4: X X X X A3+B3
Gibt es einen Zug, der ...
(Is there a train which ...)

Figure 7.1: Examples for the different phrase types taken from the ERBA corpus. Each syllable in the text is labeled with the corresponding prosodic class. X is used as an abbreviation of the class A[01]+B[01]. Note that A2 and A3 are unified to the class A[23] in the HMM and in the NN models.

Preliminary experiments were conducted modeling feature vectors like the ones described in Section 6.4.2 directly with HMMs using mixtures of normal distributions. Different features sets were tried, however, this yielded low recognition rates, cf. below. Therefore, we used an NN with six output nodes, each corresponding to one of the classes mentioned above and investigated NN/HMM hybrids of the form as described in Section 2.7.

For each utterance the syllable–wise reference labels consisting of the above mentioned six prosodic classes were translated into a sequence of phrase classes. Each phrase covered a sequence of syllables. Different kinds of phrases were investigated. In most of the experiments four basic phrase types were used. In some experiments these were further distinguished by the length of the phrase: phrases consisting of more than six syllables are considered to be long phrases, all other phrases are short phrases. Table 7.1 gives an overview of these eight classes; Figure 7.1 shows an example for each of the four basic phrase types. In the first column of the table the name of the phrase class is given. The second column shows the structure of the corresponding phrases in terms of the sequence of the six prosodic classes; (A[01]+B[01])*, for example, denotes a sequence of class sym-

```
1. Initialization
Define number and structure of phrase classes.
Define for each of these phrase classes the topology of one HMM. This includes the association of HMM transitions with one of the prosodic classes.
FOR all utterances in training set
    Translate the syllable–based prosodic reference labeling into a sequence of phrase labels.
2. Training
For each syllable compute an acoustic–prosodic feature vector.
Train the NN weights using the syllable–wise labels by, e.g., quick–propagation.
Initialize the parameters of the HMM mixture normal densities on the basis of the syllable–wise reference labeling.
    FOR all utterances in the training set
        Build utterance model by the concatenation of appropriate phrase HMMs.
        Compute new estimates for the NN/HMM parameters with the algorithm of Section 2.7.1.
    The new NN/HMM parameters are the average of the estimates computed utterance–wise.
    Build the looped HMM for recognition.
    FOR different heuristic weights $\varpi$.
        FOR all utterances in the validation set
            Perform a Viterbi search for the optimal transition sequence.
        Determine the sequence of phrase classes from this optimal path.
UNTIL the recognition accuracy on the phrase level does not improve
      anymore.
3. Evaluation
On the optimal path obtained by the Viterbi search determine for each syllable in the test set the prosodic class associated with the HMM transition.
Determine the recognition rates separately for boundaries and accents using the sequences of syllable–wise class symbols.
```

Figure 7.2: Steps which have to be conducted for the training and evaluation of an NN/HMM hybrid for the modeling of prosodic phrases. Here it is assumed that feature vectors are computed syllable–wise.

bols A[01]+B[01] of arbitrary length including zero length. The motivation behind these types of phrases is that a major difference in the basic prosodic attributes is expected, whether

- the phrase ends with B2 or B3, and whether
- the last syllable in the phrase or any other syllable is accented.

The questions in the ERBA corpus are always determined by grammatical means. Therefore, they are usually not at all or rather weakly prosodically marked. Otherwise, it would have made sense to further distinguish phrases ending with B3 by their sentence mood. By developing these phrase structures the accent symbols Axi were mapped to A0 and the labels Axn were mapped to Ax, where $x \in \{1, 2, 3\}$; cf. Section 5.1.3, page 147, for an explanation of these labels. In the case of Axa, the right–most occurrence in a word was mapped to Ax, the other were mapped to A0.

In the first series of experiments we used the NN–Feat/HMM hybrid where the NN output is assumed to be a transformed feature vector, cf. Section 2.7.1. Later, NN–Prob/HMM hybrids were also investigated. In each case, one HMM corresponds to one of the phrase classes. The NN outputs were modeled with mixtures of multivariate normal distributions. We used transition–oriented HMMs, that is, observations are associated with transitions rather than with states. With each HMM transition we associate one of the six syllable labels. Since statistical independence of the NN outputs can be assumed, we use normal distributions with diagonal covariance matrices. Continuous speech training and recognition as described in Sections 2.3 and 2.7.1 was performed. The overall approach is summarized in Figure 7.2.

Experiments and Results

A large number of different experiments was performed. In particular, many different HMM topologies were systematically investigated including simple ones as shown in Figure 2.7. Most experiments were performed with the phrase classes defined in Table 7.1. Below we give results for three different HMM topologies. For each type of topology the following parameters were varied; as for details cf. [Kah94]:

- the number of densities per mixture distribution,
- ± tying of observation densities (tying means in this case that the normal distributions of the transitions within one HMM associated with the same syllable label are identified to enhance model robustness), and
- heuristic weighting of the transition probabilities as described in [Nor91].

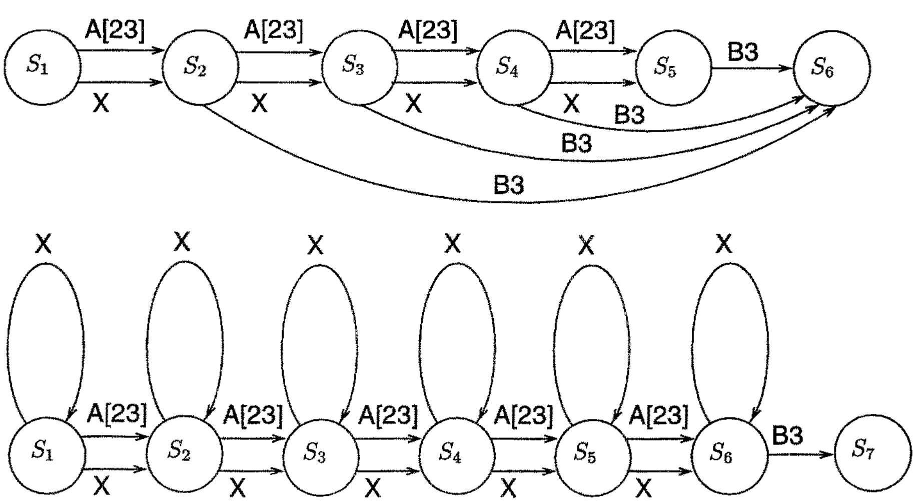


Figure 7.3: HMM structure (topology 1) yielding best boundary recognition rates in the phrase modeling experiments using the phrase models as defined in Table 7.1; top: model for Phr2–short, bottom: model for Phr2–long. X is used as an abbreviation for A[01]+B[01].

In all cases we give the recognition rates for the parameter setting, which yielded the best results.

The recognition rates for accents as well as for boundaries are given in Table 7.2; in addition, the phrase recognition accuracy (RA) is shown for completeness, although with respect to the use in an ASU system we are only interested in the recognition of accented syllables or boundaries and not in the recognition of specific types of phrases. The recognition rates are given in percent; they were evaluated word–wise without taking into account the end of utterances in the case of the boundaries. In the case of accents, the evaluation was conducted syllable–wise.

The NN used here is a preliminary version of the one described in Section 6.4, yielding a recognition rate of $RR=$ 81.2% for the three boundary classes and $RR=$ 86.1% for the two accent classes (row "NN alone"); it had 142 input features determined over an interval of ± 4 syllables except for the durational features which at that time were only computed for the actual syllable; the features are listed in Table A.4 in the appendix.

For comparison the HMM was trained to model directly the prosodic features. Only a smaller number of features as used for the NN could be used for the HMM; it consisted of a subset of 53 features. Different topologies were investigated;

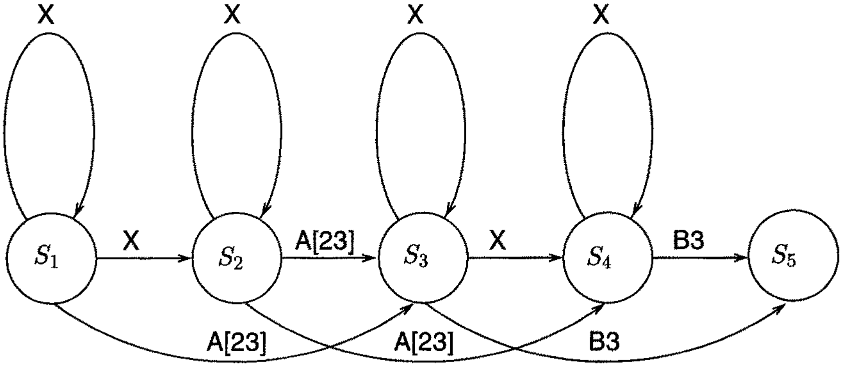


Figure 7.4: Alternative HMM structure (topology 2) using the phrase models as defined in Table 7.1 without distinguishing short and long phrases. X is used as an abbreviation for A[01]+B[01]

best results were achieved with a simpler HMM topology than the ones shown in Figures 7.3 to 7.5. The recognition rates (column "HMM alone" in Table 7.2) are even worse than using only the NN.

Then we investigated the NN/HMM hybrid, where the NN transforms the features (row "NN–Feat/HMM") using different HMM topologies. Best recognition rates for boundary detection were achieved with HMMs as shown in Figure 7.3, cf. row "topology 1" in the table. In the top of the figure the HMM for Phr2–short and at the bottom the one for Phr2–long is depicted. Transitions for accentuation (A[23]) and for A[01]+B[01] were put in parallel so that one to five accented syllables per phrase are allowed. The HMMs for the other types of phrases have the same topology, except that B3 is replaced by B2, A[23]+B3, or A[23]+B2, respectively; furthermore, in the latter two cases all the edges in Figure 7.3 labeld by A[23] are deleted. The boundary recognition rate is three percent points better than using only the NN. However, the accent recognition rate is two percent points worse than with the NN.

Better accent recognition rates were achieved with HMMs where short and long phrases got the same model. For the phrases Phr2–short and Phr2–long the topology is depicted in Figure 7.4. The main difference to "topology 1" is that in accordance with the theory only one accented syllable per phrase is allowed. Recognition rates are given in the row "topology 2" in Table 7.2. Compared with the NN the recognition rates for both boundaries and accents increased by about one percent point.

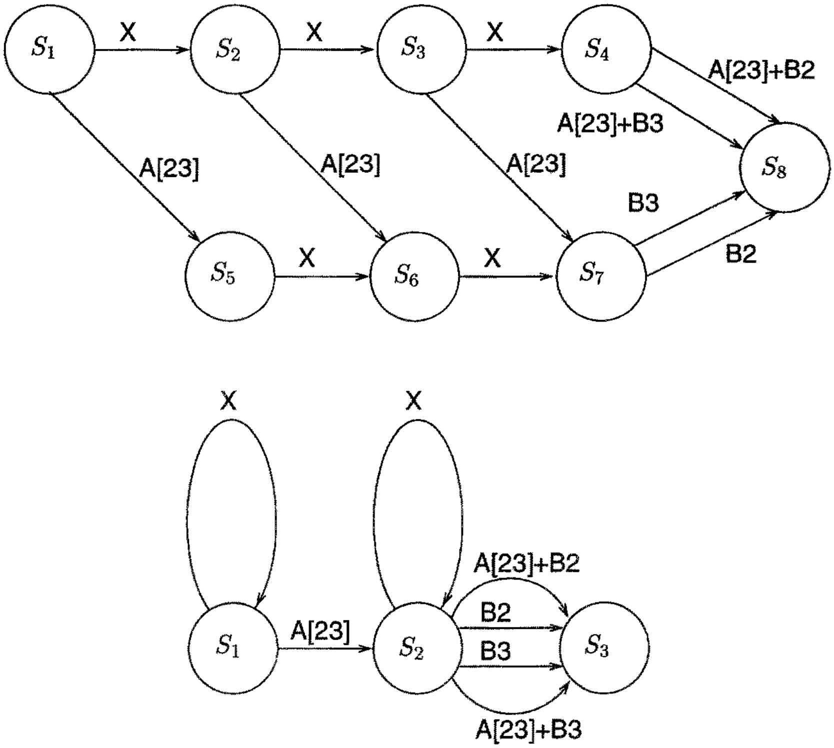


Figure 7.5: Examples for HMM structures (topology 3) which yielded good recognition rates for accents as well as for boundaries in the phrase modeling experiments. Different models were not distinguished by the phrase classes as defined in Table 7.1 but by the number of syllables per phrase. X is used as an abbreviation for A[01]+B[01]

Comparable recognition rates for accents as well as for boundaries were achieved with "topology 3" where a different type of phrase classes was defined. Phrases were only distinguished by the number of syllables no longer by their structure. At the top of Figure 7.5 the HMM for phrases consisting of four syllables is given. For phrases of up to nine syllables a similar topology was used, none of them contained loops. For all phrases of more than nine syllables the HMM at the bottom of the figure was used. With these HMMs the recognition rates for boundaries and accents both were improved by 1.4 percent points with respect to the NN alone.

We also experimented with the type of NN/HMM hybrid where the NN is used as HMM observation probability estimator (row "NN–Prob/HMM"), cf. Section 2.7.2. Best results were achieved with "topology 1" as shown in Figure 7.3.

				RA	*RR*	
phrase model	topol.	tying	# dens.	phrases	B3/B2/B[01]	A[01]/A[23]
NN alone	–	–	–	–	81.2	86.1
HMM alone	0	no	1	46.6	77.6	74.1
NN–Feat/HMM	1	no	1	50.4	84.3	84.7
	2	no	2	63.7	83.1	86.5
	3	yes	2	38.0	82.6	87.5
NN–Prob/HMM	1	–	–	53.0	85.1	89.4

Table 7.2: Recognition rates (in percent) for accents (A[01] versus A[23]) and the three boundary classes B[01], B2, and B3 on ERBA with NN/HMM hybrids. For completeness, also the phrase accuracy (RA) is given.

It can be seen in Table 7.2 that with respect to the NN (row "NN alone") the boundary recognition rate could be increased by 3.9 percent points, which is an error reduction of 21%; the accent recognition rate was improved by 3.3 percent points which corresponds to 24% percent fewer errors.

The experiments with "topology 2" were repeated using the VERBMOBIL sub–corpora BS_TRAIN and BS_TEST. The same types of phrase structures as for the experiments on ERBA were used. For this, we combined the B9 with the B0 labels yielding the class B[09], because B9 does not mark the boundary of a regular phrase. In contrast to ERBA, we now deal with word–wise instead of syllable–wise computed feature vectors, because already in Section 6.4.3 on VERBMOBIL a word–wise recognition was conducted. The recognition rates are given in Table 7.3. Two NNs are used this time. The NN for boundary classification is the same as described in Section 6.4.3 for the discrimination of B0/B2/B3/B9; For the NN–Prob/HMM hybrid system, where the NN estimates the observation probabilities, the probabilities for B0 and B9 computed by the NN were added to yield the HMM observation probability for the combined class B[09]. The NN for accent classification is as well the one from Section 6.4.3. Both NN/HMM hybrid systems considerably reduce the error rate of the boundary recognition (by up to 45%), whereas the accent recognition rate decreases. The NN–Prob/HMM hybrid system yields slightly better results than the other hybrid.

The good results obtained with the HMM, where the observation probabilities are estimated by the NN, encouraged us in the approaches we developed afterwards and which are described in the remaining chapters to assume that NNs compute probabilities. These NN/HMM hybrids do not take information about the spoken or recognized words into account. This can be an advantage if no such information is

				RA	*RR*	
phrase model	topology	tying	# dens.	phrases	B3/B2/B[09]	A/UA
NN alone	–	–	–	–	70.5	82.8
NN–Feat/HMM	2	no	2	26.0	82.3	78.7
NN–Prob/HMM	2	–	–	30.2	83.9	78.5

Table 7.3: Recognition rates (in percent) for accents (A versus UA) and the three boundary classes B3, B2, and B[09] on VERBMOBIL with NN/HMM hybrids. For completeness, also the phrase accuracy (RA) is given.

available. However, in the context of the VERBMOBIL system, the prosody module "knows" about the recognized words and should therefore utilize this information, because it might improve the recognition rates. To cope with this we developed the classification approach described in the following section.

7.2 Polygram Classifier and Combination with NN

In Section 6.4 acoustic–prosodic accent or boundary models on the basis of NNs have been described. In Section 7.1 this has been extended to phrase models using an NN/HMM hybrid. As has been outlined in Section 4.2, there is a close relationship between prosodic and syntactic structures. This means that prosodic models should take into account language information which can be derived from the spoken or recognized word chain.

In ASR n-grams are widely used as language models within word recognizers to estimate the a priori probabilities $P(v)$ of word sequences v which are then combined with the likelihoods computed by the acoustic model so that a posteriori probabilities according to the Bayes rule can be computed, cf. equation (2.2). We adapted the n-gram approach for the task of prosodic attribute classification given a word chain. Specifically, the polygram models as described in Section 2.4 are used. Our goal is to classify a word or syllable as being accented or as succeeded by a boundary. Here, we consider some context, but we do not build entire phrase models as in the previous section. Acoustic context is already taken into account in the NN. The NNs which we described in Section 6.4 were used in this section. We restrict ourselves in the following to clause boundaries, however, the approach can be applied to any type of prosodic attribute. The polygrams model the probabilities of a boundary given the left and right word context.

In [Kom94a] we already showed that polygram models can be used for the classification of boundaries. Meanwhile, n-gram models have been also used for other classification tasks as described in [ST95b, Mas95c, War95].

In this section we describe an n-gram approach for the classification of prosodic events given a partial word chain; it can also be used for the scoring of word graphs, cf. Section 8.1. We briefly described in [Kom95b] a preliminary version of the approach. The B labels were used for model training in [Kom95b]. The M labels were not available at that time; however, these turned out to yield superior results as we will show below.

The Polygram Classifier

In the following, we will first theoretically describe our approach assuming that we want to classify prosodic attributes in spoken word chains. Second, we will give an example. The method was developed such that it can be applied to word graphs as well, as will be shown in Section 8.1. Since the model is based on sequences of words and prosodic classes, we will in the following also refer to it as the *language model* (LM).

Let v_i be a word out of a vocabulary V, where i denotes the position in the utterance; $u_i \in U$ denotes a prosodic symbol associated with word v_i, where $U = U_1, \ldots, U_L$ defines an alphabet of prosodic symbols. These can for example be $U = \{\mathsf{B[01]}, \mathsf{B2}, \mathsf{B3}\}$, $U = \{\mathsf{A[01]}, \mathsf{A[23]}\}$, $U = \{\mathsf{M3}, \mathsf{M0}\}$ or a combination $\mathsf{U} = \{\mathsf{B[01]} + \mathsf{A[01]}, \mathsf{B[01]} + \mathsf{A[23]}, \ldots, \mathsf{B3} + \mathsf{A[23]}\}$ depending on the specific classification task. For example, $u_i = \mathsf{M3}$ means that the i^{th} word in an utterance is succeeded by an M3 boundary, and $u_i = \mathsf{A23}$ means that the i^{th} word is accented.

Ideally, in speech recognition we would like to model the following a priori probability

$$P(v_1 u_1 v_2 u_2 \ldots v_m u_m) \tag{7.1}$$

which is the probability for strings, where words and prosodic labels alternate; m is the number of words in the utterance. In the following, we are not interested directly in these string probabilities, but we want to model the a priori probabilities for prosodic classes given a symbol string as in term 7.1, that is, we have to estimate the probabilities

$$\begin{aligned} & P(u_i = U_l | v_1, v_2, \ldots v_m, u_1, u_2, \ldots, u_{i-1}, u_{i+1}, u_{i+2}, \ldots, u_m) \\ = \; & \frac{P(v_1 u_1 v_2 u_2 \ldots v_i U_l v_{i+1} u_{i+1} \ldots v_m u_m)}{\sum\limits_{s=1}^{L} P(v_1 u_1 v_2 u_2 \ldots v_i U_s v_{i+1} u_{i+1} \ldots v_m u_m)} \, . \end{aligned} \tag{7.2}$$

In the following our task is to recognize all the prosodic classes $u_1, \dots, u_m$ while the word sequence $v_1, \dots, v_m$ is given either by the transliteration of an utterance or by the word recognizer. If in this task the probabilities of equation (7.2) are used, a search procedure like A* is needed, because when the probability for u_i is computed the classes $u_j, j \neq i$ are unknown. Since our approach has to be applicable in the prosodic scoring of word graphs, we consider it as too inefficient to perform such a search. Therefore, we approximate

$$\begin{aligned} & P(u_i = U_l | v_1, \dots, v_m, u_1, \dots, u_{i-1}, u_{i+1}, \dots, u_m) \\ & \quad \approx \quad P(u_i = U_l | v_1, \dots, v_m) \\ & \quad = \quad \frac{P(v_1 \dots v_i U_l v_{i+1} \dots v_m)}{\sum\limits_{s=1}^{L} P(v_1 \dots v_i U_s v_{i+1} \dots v_m)} \ . \end{aligned} \tag{7.3}$$

Furthermore, the symbol string probabilities in equation (7.3) are estimated by n-grams as defined by equation (2.53):

$$\begin{aligned} & P(v_1 \dots v_i U_l v_{i+1} \dots v_m) \\ & \quad \approx \quad P(v_1) \prod_{j=2}^{i} P(v_j | v_{j-n+1} \dots v_{j-1}) \cdot \\ & \qquad \cdot P(U_l | v_{i-n+2} \dots v_i) P(v_{i+1} | v_{i-n+3} \dots v_i U_l) \cdot \dots \\ & \qquad \dots \cdot P(v_{i+n-1} | U_l v_{i+1} \dots v_{i+n-2}) \cdot \\ & \qquad \cdot \prod_{j=i+n}^{m} P(v_j | v_{j-n+1} \dots v_{j-1}) \ . \end{aligned} \tag{7.4}$$

Applying the approximation (7.4) to equation (7.3) yields

$$\begin{aligned} & P(u_i = U_l | v_1, \dots, v_m) \\ & \quad \approx \quad \frac{P(U_l | v_{i-n+2} \dots v_i) P(v_{i+1} | v_{i-n+3} \dots v_i U_l) \cdot \dots}{\sum\limits_{s=1}^{L} \{ P(U_s | v_{i-n+2} \dots v_i) P(v_{i+1} | v_{i-n+3} \dots v_i U_s) \cdot \dots} \\ & \qquad \frac{\dots \cdot P(v_{i+n-1} | U_l v_{i+1} \dots v_{i+n-2})}{\dots \cdot P(v_{i+n-1} | U_s v_{i+1} \dots v_{i+n-2}) \}} \ . \end{aligned} \tag{7.5}$$

In practice the estimation of the probabilities of equation (7.5) is done by an

$	also	MU	dienstags		paßt								
	also		dienstags	M0	paßt		es						
			dienstags		paßt	M0	es		Ihnen				
					paßt		es	M0	Ihnen		ja		
							es		Ihnen	MU	ja		$
									Ihnen		ja	M3	$

Table 7.4: Symbol chains used for the training of a prosodic polygram classifier.

n-gram modeling of symbol strings

$$v_{i-n+2} \dots v_i U_l v_{i+1} \dots v_{i+n-1} \tag{7.6}$$

as described in Section 2.4. In the experiments presented below we use polygrams. Recall that polygrams are a mixture of n-grams where n can be arbitrarily large. However, in practice, we restrict n to a certain value to limit the model size. In the following n denotes the maximum n used for the n-grams contained in the polygram model.

We use the probabilities for prosodic classes defined by equation (7.5) for their classification. Given a word chain $v_1 \dots v_i \dots v_m$, the optimal prosodic class u_i^* is determined by maximizing the probability of equation (7.5):

$$u_i^* = \operatorname*{argmax}_{1 \leq l \leq L} P(u_i = U_l | v_1, \dots, v_m) \; . \tag{7.7}$$

Let us consider an example to illustrate this approach. Assume we are given the following turn in which the references of the prosodic–syntactic M labels have been inserted:

$ also MU dienstags M0 paßt M0 es M0 Ihnen MU ja M3 $ (7.8)
($ *so* MU *Tuesdays* M0 *suits* M0 *it* M0 *you* MU *isn't–it* M3 $)
so Tuesdays is ok for you isn't it

The "$" symbol is used as a special turn begin and turn end marker. For training, as many copies of this symbol chain are created as many as there are M labels or word boundaries. In each copy only one label is kept, all others are deleted. Furthermore, only $n-1$ words around each M label are used; further context contains no information with respect If the polygrams are again limited to n-grams with $n \leq 3$ the symbol chains depicted in Table 7.4 are derived from the turn of example (7.8).

These chains are used for the polygram training. If larger contexts are to be considered, that is, the upper bound for n is greater than 3, similar but larger symbol chains are used.

When such polygram models have been trained, they can be used for classification according to equation (7.7), that is, given a word chain each word boundary is classified at a time by inserting each of the prosodic class symbols and computing the probabilities for the resulting symbol chain. Let us still consider the utterance from example (7.8) and again assume the polygram models are limited to n-grams with $n \leq 3$. Assume furthermore, we want to classify the word boundary after dienstags, whether there is a M3, MU or M0 boundary. Then we have to compute the three probabilities

$$P(also\ dienstags\ \mathsf{M3}\ passt\ es) \tag{7.9}$$

$$P(also\ dienstags\ \mathsf{MU}\ passt\ es) \tag{7.10}$$

$$P(also\ dienstags\ \mathsf{M0}\ passt\ es) \tag{7.11}$$

and decide for the class with the maximum probability. In this case of trigrams no further context of the symbol chain is needed. The approximation of these probabilities by trigrams yields in the case of (7.9) to

$$\begin{aligned} & P(also\ dienstags\ \mathsf{M3}\ passt\ es\ Ihnen\ ja) \approx \\ & \qquad P(\mathsf{M3} \mid also\ dienstags) \\ & \quad \cdot\ P(passt \mid dienstags\ \mathsf{M3}) \\ & \quad \cdot\ P(es \mid \mathsf{M3}\ passt) \end{aligned} \tag{7.12}$$

Combination of Language and Speech Model

So far we described the language model, which will only be part of our final prosodic model. The polygrams provide a priori probabilities for prosodic classes given a left and right context of words. The acoustic or speech model is built by an NN as described in Section 6.4. As discussed in Section 6.4.4 the NN is considered to estimate likelihoods $P({}^i\mathrm{c}|U_l)$. Both models are combined via Bayes rule. However, since both models are suboptimal, it was found useful to introduce a weight ξ which balances the influence of these models. So, the combined NN/LM classifier is based on the a posteriori probabilities

$$
\begin{aligned}
P_{v_i}(U_l) &= P(U_l|{}^i\mathbf{c}, v_{i-n+2}, \dots, v_{i+n-1}) \\
&= \frac{P({}^i\mathbf{c}|U_l)P^{\xi}(v_{i-n+2}\dots v_i U_l v_{i+1}\dots v_{i+n-1})}{\sum\limits_{s=1}^{L} P({}^i\mathbf{c}|U_s)P^{\xi}(v_{i-n+2}\dots v_i U_s v_{i+1}\dots v_{i+n-1})} .
\end{aligned}
\tag{7.13}
$$

Note that the equality only holds if $\xi = 1$. The notation $P_{v_i}(U_l)$ is used as an abbreviation henceforth; it denotes the probability of the prosodic class symbol U_l being associated with the i–th word of an utterance; similarly we will denote with $P_{W_i}(U_l)$ the probability of class U_l associated with the word hypothesis W_i. Recall that the computation of ${}^i\boldsymbol{c}$ is also based on $\pm j$ context words: $v_{i-j+1}\dots v_i\dots v_{i+j}$. We heuristically determined the value for ξ on the test data used in the boundary recognition experiments of this section; $\xi = 5.0$ was then used for all experiments described in this section. For the scoring of word graphs, which is described in Section 8.1, we achieved best results with $\xi = 3.0$. These test data, on which ξ was determined, eventually have to be considered as cross–validation data with respect to the experiments described in Sections 8.2 and 8.3 where the probabilities computed according to equation (7.13) are used in other ASU modules. The classifier as defined by (7.13) can either be used to classify the word v_i in an utterance to one out of a set U of prosodic classes by the decision rule

$$
u_i^* = \underset{1\leq l\leq L}{\operatorname{argmax}} P_{v_i}(U_l) \tag{7.14}
$$

or the probabilities $P_{v_i}(U_l)$ can directly be used by other components of an ASU system. In this context it is important to have an approach like this one, which does not conduct hard decisions but which provides probabilities for different hypotheses.

Experiments and Results

We carried out different experiments on ERBA as well as on VERBMOBIL to test the approach we developed above. The combined NN/LM classifier for clause boundaries eventually is used in Chapter 8 for the integration with the syntactic analysis. In all experiments we used category–based polygrams. For ERBA only the names of train stations, months, days of the weeks, and numbers were represented by a category, respectively. In the case of VERBMOBIL we experimented with a set of 574 categories designed heuristically by hand. In this category system, 365 word types occur more than 20 times in the M_TRAIN sub–corpus. These words and the prosodic classes each build a category on its own. The less frequent

	RR	RR(B01)	RR(B2)	RR(B3)
LM_2	97.7	98.4	95.9	93.9
LM_4	99.3	99.6	98.4	99.4

Table 7.5: Recognition rates in percent for B01 vs. B2 vs. B3 on ERBA.

words were grouped into categories depending on their intuitive syntactic/semantic function. A default category collects all the word types which could not be assigned to one of the categories in the training corpus[1].

We also investigated category sets of different sizes which were automatically determined by the approach described in [Och95, ST95b]. With this approach a partition of the vocabulary into a predefined number of categories is achieved so that the test set perplexity of a bigram model is minimized. As optimization method among others simulated annealing is used. In preliminary experiments we found that there is no significant difference between the manually created category system and the automatically determined ones as long as the number of categories is between 100 and 600. If considerably more than 600 categories are used, with the currently available training data no robust polygrams can be trained anymore. The automatically created category systems have the disadvantage that the categories have no "meaning"; for example, numbers are distributed over different categories. If these category systems have to be expanded with respect to words not included in the training vocabulary, these words can only be put in a default category. The manually designed category system has the advantage that it can be better expanded; for example, numbers not contained in the training corpus are assigned to the category which already contains other numbers. Therefore, in the following we will only use the manually created category system.

Results for the distinction of three boundaries with a pure polygram classifier on ERBA are given in Table 7.5. In the tables LM_j denotes a polygram where the n-grams are limited to $n \leq j$. It can be noticed that almost no errors occur, which is due to the low perplexity of the corpus, cf. Table 5.1. Therefore, the ERBA corpus is not very interesting for further investigation concerning this topic.

All of the following experiments are performed on VERBMOBIL. The sub–corpus BS_TEST was used for testing in all cases. The polygrams for the distinction of the M labels were trained on M_TRAIN, the ones for S recognition were trained on S_TRAIN, and the polygrams which model the B labels were trained on BS_TRAIN. We found for the available training data and for a category system of a few hundred categories that the use of n-grams with $n > 3$ did not improve the

[1] We wish to thank Florian Gallwitz who built the major part of this category system.

reference	% recognized		
	M3	MU	M0
M3	77	0	23
MU	8	52	40
M0	2	0	98

Table 7.6: Confusion table showing the results achieved with a polygram classifier trained and tested on M3/MU/M0 labels.

classifier	RR (CRR) evaluated on	
	S3+/S3–	M3/M0
M3/M0–LM_3	95 (87)	95 (86)
S3–/S3+–LM_3	92 (86)	92 (83)

Table 7.7: Comparison of polygrams trained on M labels versus the one trained on the S labels.

recognition rate. All recognition rates given below do not consider the end of turns because their classification as clause boundary is trivial. Non–verbals including pauses were discarded from the word chain prior to polygram classification. Preliminary experiments showed that this yields slightly better results. A reason for this is the low correlation between pauses and clause boundaries, cf. Section 5.2.8.

We started with experiments on the three classes M3, MU and M0. The recognition result is given in Table 7.6 as a confusion table. Both M0 and M3 are recognized very well. The undefined or ambiguous boundaries MU were badly recognized. This is not surprising, because MU by definition refers to the word boundaries where it cannot be decided from the word chain if there is a boundary or not. In other words, if the MUs were reliably recognized with the polygrams, our M labeling system would be wrong. Altogether the results meet our expectations and show once more, as in Section 6.4, that the M labeling scheme is sound and has been consistently applied.

In Section 5.2.4 we stated that the S labels reflect syntactic structures very precisely. Therefore, in general, one might expect that stochastic language modeling using these labels can be done more reliably than using the rather coarse M labels. In Table 7.7 results for polygrams trained on M3 versus M0 are compared with the recognition rates of polygrams trained on S3+ versus S3–. The ambiguous boundaries MU and S3? have not been considered neither during training nor during testing. In the table, recognition rates (RR) are in percent for different polygram classifiers (first column) distinguishing between clause boundary and no–boundary. The average of the class–wise recognition rates (CRR) is given in parentheses. At the top of the table the reference labels used for the evaluation in the respective column are shown. It turns out that the M3/M0–LM is much better than the S3–/S3+–LM even when tested on S3–/S3+. This is due to the much larger amount of available training data.

After these investigations we intended to find the best suited classifier for

classifier		*RR* (*CRR*) evaluated on		
no.	name	B3/B[029]	M3/M0	MB3/MB0
1	B3/B[029]–NN	88.3 (86.8)	86.4 (82.1)	85.4 (81.6)
2	M3/M0–NN	81.5 (85.3)	86.0 (86.7)	82.2 (83.3)
3	M3/M0–LM_3	91.7 (84.2)	94.7 (85.5)	92.1 (84.1)
4	M3/M0–NN & M3/M0–LM_3	92.3 (87.8)	95.2 (88.6)	92.8 (87.2)
5	B3/B[029]–NN & M3/M0–LM_3	93.7 (89.3)	95.5 (89.0)	93.9 (88.0)
6	B3/B[029]–NN & B3/B[029]–LM_3	92.7 (84.5)	92.3 (81.9)	91.6 (81.1)

Table 7.8: Recognition rates (*RR*) in percent for different classifiers (first column) distinguishing between clause boundary and no–boundary. Classifiers no. 1 and 5 are further used in the parsing experiments described in Section 8.3. With respect to this the result achieved using MB3/MB0 as reference is most important.

clause boundaries to be used for the integration with the syntactic analysis, cf. Section 8.3. In Table 7.8 we compare the recognition rates of two NNs alone, the polygrams trained on M3 versus M0 and a combination of both. The NN results were copied from Table 6.5 (page 202) for convenience. At the left–hand side of the table the number and the name of the classifier is given. At the top of the table the reference labels which were used for the evaluation of the classifiers are shown. Recall that the MB3/MB0 labels are a combination of M and B labels such that M3/M0 has priority and B3/B0 is used at MU boundaries. We were mainly interested in these boundaries; they reflect what we eventually intend to recognize in an ASU system because M3 and M0 is close to the needs of a syntax module and, furthermore, at the MU boundaries the acoustic–prosodic classifier has to decide whether there is a clause boundary or not. In the table the recognition rates for B3/B[029] and M3/M0 are given for completeness. Note that the latter do not take into account word boundaries labeled with MU.

In the case of the NNs we are mainly interested in the class–wise recognition rate *CRR* because the NN estimates likelihoods which do not take into account a priori probabilities. However, for the combined model the total recognition rate *RR* is the figure we want to optimize. When looking at *CRR* we already found in Section 6.4 that the M3/M0–NN is better in distinguishing MB3/MB0 than the B3/B[029]–NN. Therefore we first combined the M3/M0–NN with the polygrams to the "M3/M0–NN & M3/M0–LM_3" classifier (no. 4 in Table 7.8). As can be seen, in all columns the recognition rates are better than with the NNs alone (classifiers 1 and 2) or the polygrams alone (classifier 3); this is even the case for the B reference labels.

Then we investigated the combination of the M3/M0–LM_3 and the B3/B[029]–NN; the latter alone shows a lower *CRR* on the MB3/MB0 labels. However, the combined classifier (no. 5) reduces the error rate RR on these labels from 7.2% to 6.1%, that is, a reduction by 15%, with respect to the "M3/M0–NN & M3/M0–LM$_3$" classifier. Even the CRR is better for this classifier. This means that the B3/B[029]–NN, so to speak, adds more information to the M3/M0–LM$_3$ than the M3/M0–NN.

In the last row of the table the result for an NN/LM classifier is given where the polygrams were trained on the B labels. When comparing classifiers 5 and 6 it turns out that the M3/M0–LM is even better in recognizing B3/B[029] than the B3/B[029]–LM. The latter result could especially not be expected because the B labels are perceptual labels. This is due to the much larger amount of available training data for the M3/M0–LM which results in a better classifier regardless the systematic differences between the M and the B labels, cf. Section 5.2.7.

From the classification results we can draw the following conclusions:

- The M labels are best suited for boundary language modeling because of the large amount of training data.
- The M labels are consistent.
- The M labels meet prosodic constraints.
- Speakers use prosody in a consistent way.

If speakers would not use prosody in a consistent way, one could not obtain good results with the B3/B[029]–LM, and, furthermore, it would not be possible to predict B3/B[029] reliably with the M3/M0–LM. The best NN (no. 1 in Table 7.8) and the best combined classifier (no. 5 from Table 7.8) will be further used in this thesis, cf. Section 8.3.

7.3 An Integrated Speech and Language Model using MSCTs

In the approach described in the previous section we tried to model the a posteriori probability $p(\Omega_\kappa|\boldsymbol{c})$ by applying the Bayes rule, cf. equation (2.2), that is, the a priori probabilities p_κ (the language model) and the likelihoods $p(\boldsymbol{c}|\Omega_\kappa)$ (the speech model) are estimated separately. As discussed in Section 2.1 this approach is optimal with respect to the error probability provided p_κ and $p(\boldsymbol{c}|\Omega_\kappa)$ can be estimated in an optimal way. However, the approach undertaken in the previous section is based on a couple of assumptions which are not fulfilled in practice, for example, the polygrams take only into account a local context. As discussed in

Section 6.4.4, the NN only computes approximations of the probabilities $p(\boldsymbol{c}|\Omega_\kappa)$. As a consequence the entire approach is suboptimal, hence, it might by useful to develop a model which directly estimates $p(\Omega_\kappa|\boldsymbol{c})$. With respect to this, the MSCTs as described in Section 2.8 are an ideal approach for training an *integrated model of speech and language* and thereby estimating the class a posteriori probabilities. Furthermore in contrast to NNs and polygrams, they can take into account arbitrarily large context in an efficient manner in both the acoustic and the language information.

In the following we will describe experiments which we summarized already in [Nöt96a], cf. also [Fis94, Geb95]. We restricted these experiments to the recognition of B3 boundaries on VERBMOBIL. It will turn out that the integration of acoustic and language information improves the recognition rate over separate models. However, altogether, probably due to the lack of sufficient training data, the MSCT approach shows somewhat lower recognition rates than the combination of polygrams and NN on the same data, cf. Section 7.2.

The Approach

First, we tried to directly classify the features which were described in Section 6.4.2 with MSCTs; this yielded very poor results due to the limited training data. Therefore, we used three NNs for the processing of the acoustic–prosodic features. Two NNs were trained on the data described above to distinguish between B3/B[029] and accented (henceforth A) versus unaccented words, respectively. Recall that the position of accents can indicate the phrase structure of a turn in addition to the prosodic boundary markers, cf. Section 4.1.2. The third NN is the one described in Section 6.3 for the classification of sentence mood. Each of the words in the corpus was attached with the class probabilities computed by these NNs. As for a discussion about how and what probabilities are computed by the NNs cf. Section 6.4.4. For the classification of a particular word with the NN for the sentence mood, we computed the features as described in Section 6.3 using the simplifying assumption that this word is at the end of an utterance.

These probabilities and a few additional acoustic–prosodic features were used as continuous *features* in the multi–level information to be modeled by MSCTs, specifically, the following continuous input features were used:

- $P(\boldsymbol{c}|\mathsf{B3})$: the probability for a B3 boundary computed by the NN.
- $P(\boldsymbol{c}|\mathsf{A})$: the probability for a word being accented computed by an NN.
- MOOD: denotes the three probabilities for the intonation contour being rising, falling or continuation–rise.

- FEAT: denotes the following five acoustic–prosodic features: the intrinsically normalized duration of the final syllable and syllable nucleus in the word, the F0 regression over two different windows, the mean F0 over the final syllable in the word.

The features FEAT are computed from the time–alignment of the spoken word chain as described in Section 6.4.2. Note that they are a subset of the features which are the input to the B3–NN.

Furthermore, the multi–level information contained the following categorical or discrete features:

- The word identity itself: the corpus consists of about 1200 word types including non–verbals.
- 150 word categories: each word uniquely corresponds to one category. The categories were automatically determined so as to optimize the bigram test set perplexity using simulated annealing [Och95, ST95b].

These multi–level features are computed for each word in the corpus. In the case of the non–verbals default values are taken for the continuous features.

Results

We will first describe the results of the different experiments described in [Nöt96a]. These were obtained using preliminary training and testing subsets of the VERBMOBIL corpus: the training corpus consisted of 22 dialogs (592 turns, 32 different speakers, 71 min of speech, 9336 words) and 3 dialogs were used for testing (48 turns, 4 different speakers, 237 sec of speech, 520 words). The testing data consist of 74 B3 and 398 B[029] boundaries not counting the end of turns. Finally, we will give the results for the best experiment repeated on the BS_TRAIN and BS_TEST sub–corpora used for other B3/B[029] classification experiments described in this thesis.

Different MSCTs or SCTs were trained using the algorithm described in Section 2.8. The results of eight experiments are depicted in Table 7.9. The first column in the table specifies the experiment number to which it is referred to in the following; the second and third column specify the input features used (except for experiment 8); the fourth column shows the total recognition rate, whereas the fifth column gives the recognition rate of B3 alone. The recognition rates do not take into account the turn final boundaries, which to classify is a trivial task. For the first two experiments "traditional" SCTs were used. Experiments 3–7 were conducted using MSCTs which integrate the acoustic information computed by an

	MSCT input		results	
Exp. no.	entrance level	additional features	RR	RR(B3)
1	words	—	86.6	22.9
2	category	—	87.1	29.4
3	category	words	87.5	32.4
4	—	$P(\boldsymbol{c} \mid \text{B3})$	87.9	36.5
5	category	words, $P(\boldsymbol{c} \mid \text{B3})$	87.7	44.6
6	category	words, $P(\boldsymbol{c} \mid \text{B3})$, $P(\boldsymbol{c} \mid \text{A})$ MOOD, FEAT	90.3	44.5
7	words	$P(\boldsymbol{c} \mid \text{B3})$, $P(\boldsymbol{c} \mid \text{A})$, MOOD, FEAT	89.6	54.1
8	NN & SCT(category)		88.6	40.5
7	repeated using BS_TRAIN, BS_TEST		91.4	55.2

Table 7.9: Results for the different MSCTs for B3/B[029] recognition.

NN with the word information. In addition the result of a classifier is given, which externally combines the probabilities computed by the SCT and the NN (experiment 8).

We used either the *word* or the *category* level as the entrance level. Note that in the first case the *category* level is of no further use because it provides no additional information. So far no explicit syntactic or semantic information, which could, for example, be computed by a parts–of–speech tagger, was used.

From experiment 1 to 7 the recognition rate of B3 improves with increasing experiment number. In most cases also the total recognition rate increases. In the following we will mention the B3 recognition rates only: In the first experiment only the words were used, yielding a recognition rate of 22.9%. This could be improved to 29.4% by using the categories instead of the words (experiment 2). So far only the "traditional" SCT approach with regular expressions as questions is used. When combining the *category* level and the *word* level with the *category* level being the entrance level, the recognition rate further improves to 32.4% (experiment 3). So far only discrete features have been used. In experiment 4 the probability $P(\boldsymbol{c} \mid \text{B3})$ was used as the only feature; since the entrance level is not used only numerical questions about the probability $P(\boldsymbol{c} \mid \text{B3})$ attached to the word to be classified is used. Thus, the SCT more or less learns the a priori probability $P(\boldsymbol{c} \mid \text{B3})$. The recognition rate (36.5%) of this classifier is better than the one of the purely categorical ones. Combining the different information sources, that is, *words*, *categories*, and $P(\boldsymbol{c} \mid \text{B3})$, in a single MSCT the recognition rate increases to 44.6% (experiment 5). Using the probability $P(c|\mathsf{A})$ and the MOOD probabilities as features in addition does not change the B3 recognition rate, but it improves the total recognition rate (experiment 6). When keeping these continuous features but switching to the words as entrance level, the different knowledge sources obviously are integrated in a more effective manner by the MSCT, which improves the recognition rate further to 54.1%.

Since the experiments for [Nöt96a] were conducted the training and testing data for prosodic experiments conducted in the VERBMOBIL project have been changed to BS_TRAIN and BS_TEST, cf. Section 5.2.1. These were used in the B3/B[029] classification experiments presented in this chapter and in the previous chapter. In order to compare the performance of the MSCTs with the other classifiers we repeated experiment 7 using BS_TRAIN and BS_TEST. We achieved slightly better results, cf. Table 7.9.

In experiment 8 we combined at each of the words v_i of the utterances the probabilities $P({}^ic|\mathsf{B3})$ and $P({}^ic|\mathsf{B[029]})$ computed by the B3–NN and the a priori probabilities $P(v_1 \dots v_i\mathsf{B3}v_{i+1} \dots v_m)$ and $P(v_1 \dots v_i\mathsf{B0}v_{i+1} \dots v_m)$ estimated by the SCT of experiment 3, via Bayes rule yielding a recognition rate of 40.5%. The SCT in this case is a pure language model, estimating probabilities for word or category sequences where one boundary symbol has been inserted. Note that this is already worse than the result of experiment 5 with which it directly compares, because in experiment 5 the same knowledge sources were used as in the combination of SCT and B3/B[029]–NN. However, the MSCT allows for the integration of even more information, which eventually (experiment 7) yields better results than the pure multiplication of probably bad probability estimates. The main conclusion we can draw from these results is that the integration of different knowledge sources including categorical and continuous features improves the recognition rate. However, as already indicated, the recognition rates are still somewhat lower than the ones we achieved with the combination of the B3/B[029]–NN and polygrams, cf. Table 7.8. We believe that the main reason is the small amount of training data, which especially does not allow the MSCT to make use of a broad context within the questions in the nodes.

During the experiments, we observed that adding more input features sometimes can reduce the recognition rate of the MSCTs a lot. This occurs because the tree–growing algorithm does not find the tree structure which is globally optimal for the training data, rather, the algorithm employs a greedy question selection criterion which chooses for each current leaf node a locally optimal question. This can cause a globally suboptimal question to be asked early, since it might split the training data best at that time. In an extreme case, the subtrees of this node could be identical, which causes optimization problems having only sparse training data.

Of course with "unlimited" training data this problem does not exist.

In [Wan92] traditional classification trees (CTs) were also used for prosodic boundary classification. This approach is suboptimal compared to ours because at each word boundary only a fixed sized feature vector is input to the classifier. Therefore, no arbitrarily large context can be taken into account. Furthermore, no automatically determined acoustic–prosodic information was used so that these CTs are a pure language model. With this approach a recognition rate for boundary versus no–boundary classification of 82% was achieved on a balanced subset of the ATIS spontaneous speech corpus.

7.4 Summary

NNs were used in Chapter 6 for the acoustic–prosodic modeling. The feature vector they receive as input takes some context into account. However, since the different prosodic attributes influence each other it should be beneficial to develop models for entire prosodic phrases. HMMs have been proven very useful to model time sequences of feature vectors in speech recognition. Therefore we used HMMs for the modeling of prosodic phrases. A prosodic phrase contains one and only one B2 or B3 boundary, which is at the right end of the phrase. A phrase structure is given by a sequence of accent and boundary labels. These phrase structures are clustered for the entire training corpus to retrieve prototypical phrase classes. Each of these phrase classes was modeled by one HMM. The transitions in the HMMs are associated with prosodic labels on the syllable level. During recognition a Viterbi search determines the optimal transition and thereby the optimal label sequence. Directly modeling acoustic–prosodic features with HMMs was not successful. Therefore, we used an NN/HMM hybrid, where the NN performs a feature transformation and the HMM models the phrase structure. In fact the NNs described in Chapter 6 are used, which have one output node per prosodic accent or boundary class. We experimented on ERBA with many different and unconventional HMM topologies. A slight improvement with respect to the NN alone could be achieved. However, when we used the NN for computing the HMM observation probabilities, the recognition rate increased by about 4 percent points for the three boundary classes and by 3 percent points for the two accent classes. These experiments were repeated on VERBMOBIL corpora yielding an improvement of the boundary recognition rate by 13 percent point, while the accent recognition rate decreased by 4 percent points.

The main drawback of this approach is that it does not take into account any information about the wording. We used polygrams for this task and combined them with NNs to a classifier for prosodic boundaries. With this approach no entire phrases are modeled, but we take more context information into account than with the NN alone. In this task polygrams model the probabilities for clause boundaries given a left and right context of words. For the training appropriate partial symbol chains consisting of words and one reference boundary symbol are used. During test a particular word boundary has to be classified. We simply insert all the prosodic classes (one at a time), including a symbol for no–boundary, between the two words and calculate the probabilities with the polygram model. With this approach and using the prosodic–syntactic M labels a recognition rate of about 95% could be achieved on VERBMOBIL data.

Both the language and the acoustic model used in this approach are suboptimal due to the underlying assumptions. Therefore, it might be useful to develop models which integrate language and acoustic information. MSCTs were considered to be useful for this task. Each word of the word chain was attached with categorical information, few acoustic prosodic features, and probabilities for the word being accented, succeeded by a boundary, and for three different sentence mood classes. We could show that an MSCT employing all this information has better recognition rates for prosodic clause boundary detection than an MSCT only trained on words and categorical information. The latter MSCT constitutes a pure language model. When we combined the MSCT probabilities externally with the NN probabilities the results were worse than for the integrated MSCT model. This shows that MSCTs are suitable for developing integrated speech and language models. However, the MSCT approach showed lower recognition rates than a polygram classifier trained and tested on the same data. We believe the reason is that MSCTs need a large amount of training data.

Currently, we consider the combined NN/polygram classifier to be the best model to be integrated with other ASU modules as will be described in the following chapter.

Chapter 8

Integration of the Prosodic Attributes in ASU

In Sections 6.3 and 6.4.3 we described several acoustic–prosodic classifiers, and in Section 7.2 a combined NN/polygram classification method was developed. This chapter will show how these classifiers were integrated in the ASU systems EVAR and especially VERBMOBIL Chapter 3. Note that the methods developed are general in the sense that they do not depend on the specific system characteristics, for example, they do not depend on the grammars used in the parsers and not on the particular semantic formalism. We investigated the use of prosody on almost all levels of current ASU systems, namely word recognition, parsing, semantic analysis, and dialog. The results presented in this chapter are also summarized in [Nie97].

The emphasis of the work leading to this thesis was on the speed–up and disambiguation of the parsing of word graphs, cf. Sections 8.1 and 8.3. This is achieved by guiding the search using prosodic clause boundary scores. Our approach differs from others in that we first compute prosodic information on the basis of word graphs which is then directly used in the parsing process. In contrast, the approaches described in [Ost93b] as well as in [Hun95a] first parse word chains; in a second step, they prosodically verify alternative parses. Therefore, we had to develop an algorithm for the prosodic scoring of word graphs, which will be described in the first section. The succeeding sections describe the use of the prosodic information by different linguistic modules.

The algorithms we developed were evaluated within off–line experiments on large spontaneous speech databases. This showed a drastic improvement especially of the syntactic analysis by the use of prosody. Furthermore, the algorithms were integrated in the EVAR and VERBMOBIL prototype systems and could therefore be tested within fully operational systems. Both prototype systems have success-

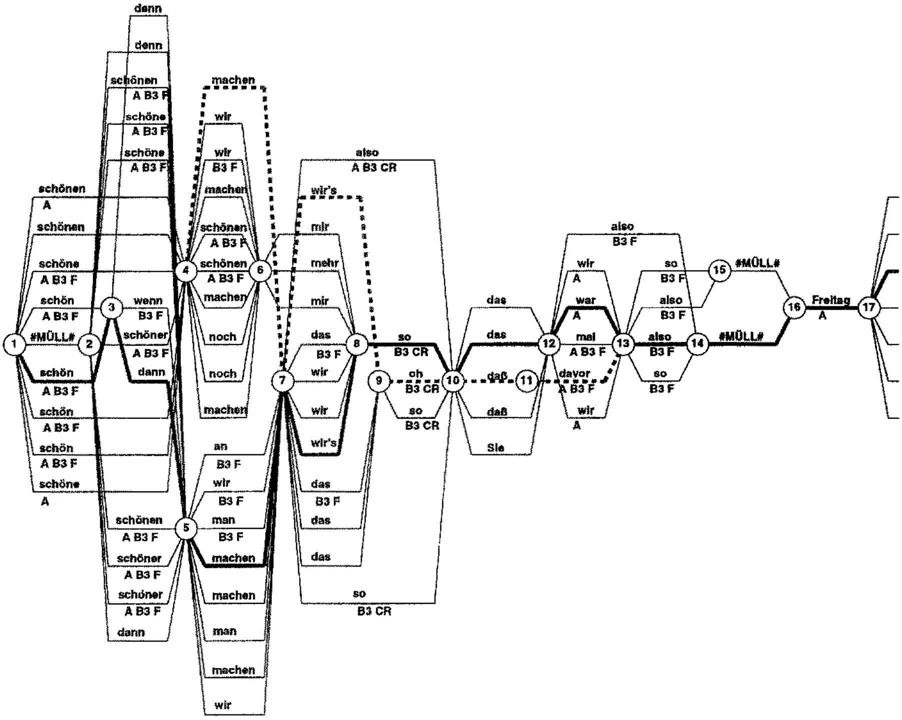


Figure 8.1: Part of a typical word graph, which was generated with the VERBMOBIL recognizer of Daimler Benz for a turn of the VERBMOBIL corpus. The word hypotheses were prosodically classified with the B3/B[029]-NN.

fully been demonstrated to the public at several events as press conferences, the Hannover CeBit fare and the Internationale Funkausstellung at Berlin. At these occasions also the usefulness of the prosody module could be made obvious to journalists [Rub96, Tho96, SPI97].

8.1 Prosodic Scoring of Word Hypotheses Graphs

A necessary prerequisite for the approaches described in the remainder of this chapter is a method for the prosodic scoring of word graphs. We already summarized the basic idea and preliminary results in [Kom95b]. The combined NN/polygram classifier as described in Section 7.2 is applied to each of the word hypotheses in the word graph. As a result the word hypotheses are annotated with probabilities for the different prosodic classes.

Our approach differs from others as [Nöt91a, Tay95, Str96] where syllable nuclei are classified which are detected in the speech signal independently from the word recognizer. In the case of prosodic boundary recognition, these approaches at best compute prosodic scores for the nodes in the word graph independently from the recognized words. Additionally, one has the problem of time aligning the detected syllable nuclei with the positions of the word graph nodes. In contrast to this, we believe it is important that each word hypothesis is individually scored because, as discussed in Section 6.4, the computation of the prosodic features and thereby the classification results depends very much on the particular word and on the time–alignment of its pronunciation.

As an example, consider the part of a word graph depicted in Figure 8.1[1]. The spoken word chain is given in example (8.1); the corresponding path in the word graph is indicated by bold lines. In the example the punctuation has been derived from the perceptual prosodic labels which were provided by Universität Braunschweig.

Schön. Dann machen wir's so. Das war also Freitag (8.1)
(Fine. Then do we–it this–way. That was so Friday)
Fine. Then let us do it this way. So that was Friday

The word graph in the figure has been prosodically scored using the algorithm described below. The scores are omitted in the figure. Instead we labeled each word hypothesis W_i with B3 if its probability $P_{W_i}(\mathrm{B3})$ was greater than 0.5. Furthermore, at each B3–edge the result of the intonational sentence mood classifier is shown, which can be fall F, rise (R), or continuation–rise CR. Similarly, each edge in the figure received an A label, if $P_{W_i}(\mathrm{A}) \geq 0.5$. One can see that different edges ending at the same node sometimes are classified differently. Even two edges between the same pair of nodes and corresponding to the same word may be classified differently as, for example, the hypotheses corresponding to the word das between nodes 7 and 9. This occurs when the word recognizer hypothesizes the same word twice for time intervals, which only slightly differ, or it computes a different time–alignment of the pronunciation of the word. Recall that the logical word graph nodes do not represent a fixed point in time, but they rather correspond to a (small) time interval. To show, for comparison, the influence of the M3/M0-LM_3 classifier, cf. Section 7.2 where we depicted the word graph classified with the combined "B3/B[029]–NN & M3/M0–LM_3" in the appendix, Figure B.1. It can be seen that the classification result is "smoother". The edges in the word graphs labeled with #MÜLL# correspond to a garbage model.

[1] We wish to thank DFKI, Kaiserslautern, who provided us with the software for plotting word graphs.

The Algorithm

As mentioned before, we consider it to be important that no hard decisions are made by the prosody component of an ASU system. We compute probabilities for the different prosodic classes so that in principle during linguistic analysis each alternative can be explored. In the current implementation of the VERBMOBIL prosody module as used for the experiments in the remainder of this chapter, we compute probabilities for the word being

- accented or unaccented,
- succeeded by an MB3 or MB0 boundary, and for
- the F0 contour at the word being rise (R), fall (F) or continuation–rise (CR).

Based on the discussion in Section 6.4.4 we use the NNs or combined NN/polygram classifiers as described in Sections 6.3, 6.4 and 7.2 for the computation of these probabilities.

The application of the sentence mood NN to word hypotheses in a word graph is trivial, because the features used in this NN do not depend on the words but classify the F0 contour in a certain interval preceding a potential clause boundary. Since we do not know a priori where the clause boundaries are, we simply classify each node in a word graph and, for the sake of uniform data representation, attach the same probabilities to each of the word hypotheses ending at this node. The subsequent linguistic analysis then may draw on this information whenever it "believes" that this particular word boundary is a clause boundary. Since this classifier currently only uses certain voiced regions in the F0 contour each word hypothesis ending at the same word graph node is annotated with the same probabilities for the different sentence mood classes.

The main difficulty in the prosodic scoring of word graphs lies in the use of the classifier for accents and boundaries because these rely on features computed from the time–alignment of words being in the context of the word to be classified. An exact solution to this problem would mean that all alternative contexts which can be found in the word graph would have to be considered. For example, if at most two words to the right were considered, in the case of the word hypothesis oh between the nodes 9 and 10 the possible right contexts would be given by

- the two das and the words daß and Sie being between nodes 10 and 12, and
- the word daß between nodes 10 and 11,
- and all possible continuations of these words with the five hypotheses leaving node 12 and the one leaving node 11, respectively.

```
1. Input
Read the speech signal of an utterance or of an utterance increment.
Read the $L$ word hypotheses $W_i$ in the graph generated on the basis of this speech signal.
2. Preprocessing
Compute the signal energy for the entire speech signal, cf. Section A.1.
Compute the F0 contour for the entire speech signal, cf. Section 6.2.
FOR each word hypothesis $W_i$
    Time–align the corresponding standard pronunciation within the time interval specified by the word hypothesis, cf. Section 6.4.1.
    Determine the speaking rate $\tau_i$ given this time–alignment, cf. Section A.2.
Determine the average speaking rate as $\frac{1}{L}\sum_{i=1}^{L} C_i\tau_i$.
Perform a Viterbi forward and backward search using the acoustic and $n$-gram scores. This results for each word hypothesis in pointers to the optimal predecessor and successor, cf. Section 2.5.
3. Prosodic Scoring of the Word Graph
FOR each word hypothesis $W_i$
    Compute acoustic–prosodic features based on the time–alignment of the optimal predecessors and successors $W_{i-n}, \dots, W_i, \dots, W_{i+k}$, with $n, k$ being sufficiently large, cf. Section 6.4.2.
    Based on these features estimate acoustic–prosodic scores using NNs, cf. Section 6.4.3.
    Determine the polygram score based on the appropriate number of predecessors and successors, cf. Section 7.2.
    Combine NN and polygram scores and normalize the scores to probabilities, cf. Section 7.2.
    Annotate the word hypothesis with these probabilities, that is, insert them in the additional information string $J$, cf. Section 3.2. Boundary, accent, and sentence mood probabilities are inserted in separate fields of $J$.
4. Output
Write this prosodically scored word graph
```

Figure 8.2: The algorithm for prosodic scoring of word graphs.

In total these would be $4 \cdot 5 + 1 \cdot 1 = 21$ alternative right contexts. Similarly, one obtains 36 alternative contexts to the left. A combination of all left and right contexts would result in a total of $21 \cdot 36 = 756$ alternatives. To be exact, one would have to consider all of these, which means to compute features based on all these alternatives and to estimate the probability $P(c|u_i)$ with the NN based on each of the different feature vectors and furthermore one would have to evaluate the a priori probabilities $P(v_{i-1}v_iu_iv_{i+1}v_{i+2})$, with v_i =oh in this example, based on all the alternative word chains.

However, this approach is computationally too expensive, so that we only compute one score for each of the prosodic classes and for each word hypothesis. Therefore, the optimal left and right context of the word hypothesis to be scored has to be determined. Assume the hypothesis of the word oh between the nodes nine and ten is to be scored, and assume furthermore that the NN and the polygram classifiers need at most ± 2 context words, then a possible context of words for the classification could be the one indicated by the bold dashed line in Figure 8.1. In the previous version of our algorithm we simply considered the word hypotheses in the context of the word hypothesis to be classified, which have the highest acoustic likelihood as determined by the word recognizer [Kom95b]. Meanwhile, as a preprocessing step, we conduct a full Viterbi search on the word graph using the acoustic scores of the HMMs as well as the scores computed by a bigram language model. This search is applied in a forward pass to obtain for each word hypothesis the optimal predecessor and it is applied in a backward pass, starting at the final word graph node, to determine the optimal successor of each word hypothesis. Note that with respect to the spoken word chain these contexts are erroneous.

The full algorithm for the prosodic scoring of word hypotheses graphs is given in Figure 8.2; it includes the preprocessing and classification steps described in Chapter 6 and Section 7.2. The time–alignment is done by the word recognizer described in Section 3.1. In the current version, the word models of the best available word recognizer for the application domain are used, which are based on context–dependent phone models.

Currently, we compute the global speaking rate on the basis of all word hypotheses, which is the weighted sum of the speaking rates of the individual word hypotheses. The weight $C_i' = C(v_h, t_b, t_e)/(t_e - t_b)$ is the acoustic score of the word hypothesis normalized to the length of the corresponding time interval; cf. Section 3.2. A preliminary evaluation showed that it does not make a big difference whether this global speaking rate or a locally computed one is used.

The incremental processing of utterances, cf. Figure 8.2, has been realized because in the VERBMOBIL context utterances of up to one minute occur. So far we have not made any systematic evaluation of the effect on the recognition rates with respect to the increment length. Furthermore, no overlapping treatment of increments, for example, for feature extraction or speaking rate determination, has been realized.

Evaluation

In the following we will describe a number of evaluations of these prosodically scored word graphs. First, the word graphs are prosodically scored with the algorithm of Figure 8.2. Second, the best matching word chain is extracted from each of the word graphs; recall that these are the word chains with the greatest word accuracy. Non–verbals or pauses are not counted when the word accuracy is computed. Finally, the recognition rates of the prosodic classes are determined on these word chains.

If the accuracy of such a word chain extracted from the word graph for evaluation is not 100%, it is unclear how to interpret the prosodic classification result, because we compute probabilities for prosodic attributes given a certain sequence of words. The word information is used in the polygram as well as in the computation of the acoustic features. Therefore, a prosodic "misclassification" given a misrecognized word, or, moreover, a correct classification given a misrecognized word is difficult to interpret. Therefore, most evaluations are conducted on *correct* word graphs; recall that these are graphs which contain the spoken word chain. For comparison, we also computed the recognition rate on all word graphs for one classifier. This recognition rate does not take into account inserted words. However, a deleted word is counted as a prosodic classification error.

Recognition rates of different classifiers for clause boundaries are given in Table 8.1. Column "word graph" shows the results on word graphs. As mentioned above the prosodic scoring of the word graphs is based on erroneous information because of the selection of the context words; even if the hypothesis to be scored is correct, the context words might not correspond to the spoken words. Therefore, we are interested in a comparison of the recognition rates on word graphs with the recognition rates which are achieved when the spoken word chains are directly scored. In the table, the column "spoken word chain" gives these recognition rates.

The word graphs used in the evaluation presented in Table 8.1 were computed by the word recognizer from the Universität Karlsruhe for the VERBMOBIL sub–corpus LING_BIG[2]. The density of these word graphs is 12.4 on the average; this is about the density currently used in the VERBMOBIL system. 160 of these word graphs are correct.

The recognition rates in Table 8.1 are given for M3 versus M0. The results do not include the boundaries labeled with MU, because for a part of the LING_BIG corpus B labels are not available. Recall that these are syntactically ambiguous boundaries for which only on the basis of a perceptual prosodic analysis it can be decided whether there is a clause boundary or not, cf. Section 5.2.5. Therefore, we

[2] We wish to thank Thomas Kemp for providing us with these word graphs.

	classifier	*RR* (*CRR*) word graph	 spoken word chain
160 correct graphs	B3/B[029]–NN	77.5 (78.0)	89.3 (82.5)
	M3/M0–LM_2	90.6 (76.5)	91.0 (77.6)
	M3/M0–LM_3	91.9 (81.3)	93.5 (84.8)
	B3/B[029]–NN & M3/M0–LM_3	92.2 (86.6)	94.0 (90.0)
all graphs	B3/B[029]–NN & M3/M0–LM_3	91.5 (84.3)	93.7 (90.2)

Table 8.1: Recognition rates in percent for M3/M0 on word graphs and on the spoken word chains for the LING_BIG sub–corpus. The average of the class–wise recognition rates *CRR* is given in parentheses. Except for M3/M0–LM_2 the classifiers are the same as in Table 7.8, page 228. For the parsing experiments described in Section 8.3 we used the classifiers B3/B[029]–NN and “B3/B[029]–NN & M3/M0–LM_3”.

are not able to evaluate the word boundaries labeled with MU. Furthermore, the recognition rates do not take into account the end of turns, which can be classified in a trivial manner.

The second column in the table shows the type of the classifier. It can be seen that the recognition rates on the word graphs (column three) are drastically better for a polygram compared to the NN. Regardless of the erroneous context words, a polygram that contains also trigrams (LM_3) even reduces the error by 13.8% with respect to a pure bigram model (LM_2). The combination of NN and LM_3 only yields a slight improvement in *RR*, but a considerable improvement in *CRR*. The reason for this is that M3 are much better detected, whereas the recognition rate of the M0 does not change much. In general, the improvement by the NN seems not to be dramatic, but one should not forget that on the MU boundaries the acoustic–prosodic classification by the NN is the only way to detect clause boundaries; recall that the MU boundaries occur frequently: about 5% of all of the turn internal word boundaries are labeled with MU, or, more important, 23% of the boundaries not labeled with M0 are labeled with MU, cf. Table 5.14.

As expected, the recognition rates on the word graphs are smaller than the ones determined on the spoken word chains (column four). For the best classifier (“B3/B[029]–NN & M3/M0–LM_3”) the error rate decreases by 23% from 7.8% to 6.0% when spoken word chains instead of word graphs are scored; for the NN by itself the decrease is 52%. The recognition rate on all word graphs, including the erroneous ones, does not differ much from the recognition rate on the correct word graphs. This was at first a surprising result, however, it can be explained by the fact that incorrect words hypotheses are acoustically similar to the spoken word and they are likely with respect to the language model. Therefore, the prosodic classification is not disturbed too much.

hypotheses per spoken word	*RA* words	prosody module real–time factor	*RR* (*CRR*) M3/M0
spoken word chains	—	0.56	95.8 (91.9)
10.6	91.5	1.01	92.7 (86.3)
17.8	93.4	1.46	92.5 (86.1)
45.6	95.4	3.63	91.8 (81.7)
135.7	97.1	14.31	90.2 (81.7)

Table 8.2: Recognition rates in percent for M3/M0 and real–time factors on word graphs of different size and on the spoken word chain for the LING_SMALL set using classifier "B3/B[029]–NN & M3/M0–LM_3", cf. Table 8.1.

time–alignment	DP search	F0, energy	features	NN	polygram	total
0.58	0.18	0.21	0.16	0.01	0.24	1.64

Table 8.3: Real–time factors for the different parts of the computation of the prosody module. The numbers were determined on the word graphs of the LING_SMALL set containing 17.8 hypotheses per spoken word.

In a second series of experiments we evaluated the computation time and the recognition rates depending on the size of the word graphs. The experiments were conducted on an HP 735 work station. For the VERBMOBIL sub–corpus LING_SMALL we had word graphs of different sizes available, which were computed with the word recognizer of Daimler Benz[3]. The results are given in Table 8.2. The column *RA* gives the word accuracy of the word graph, that is, the best word accuracy of any path contained in the word graph. It can be seen that the recognition rates for M3 versus M0 decrease with the size of the word graph, which is caused by the increasing number of "wrong" word hypotheses selected as the context of the word hypothesis to be classified. Nevertheless, even with the very large word graphs of 135.7 hypotheses per spoken word the recognition rate is still considerably high. Note that the real–time factors are determined without the use of a polygram model in the Viterbi search for the optimal predecessors and successors of a word hypothesis. We found that even on the very large word graphs the use of the polygram does not significantly influence the recognition rate. This result was at first surprising, but it can be explained by the fact that already during the generation of word graphs an n-gram model has a very high impact on the structure of the graphs.

In Table 8.3 we analyzed the real–time factors separately for the components of the prosody module. For this we used the word graphs of 17.8 hypotheses per

[3] We wish to thank Pablo Fetter for providing us with these word graphs.

	RR (CRR)	
classifier	word graph	spoken word chain
A/UA–NN	82.5 (82.0)	82.8 (82.4)

Table 8.4: Recognition rates in percent for accented versus unaccented words determined on word graphs and on spoken word chains for the BS_TEST sub–corpus.

spoken word of the LING_SMALL corpus. This time we turned the Viterbi search on. Note that the computation of F0 and energy only (linearly) depends on the speaking time. The amount of time needed for the Viterbi search increases non–linearly with the number of word hypotheses in the graph. The computation time of all other components increases linearly with the number of word hypotheses in the graph.

We also evaluated the accent classification on word graphs for the BS_TEST sub–corpus. The word graphs were provided by Daimler Benz[4] and had a density of 12.5. The results for all of the word graphs, not only the correct ones, are given in Table 8.4. It can be seen that there is no significant difference in the performance between the accent scoring of word graphs and spoken word chains.

We can conclude that although on word graphs the determination of context words is erroneous, the algorithm we developed achieves recognition rates which decrease not very much compared to the direct classification of the spoken word chain. Furthermore, the real–time factors show that it can be efficiently applied. Therefore, this algorithm builds the core of the prosody module as integrated in the VERBMOBIL system, and the prosodically scored word graph builds the basis for the evaluations in the remainder of this chapter.

8.2 Word Recognition

As has been pointed out in Section 1.2.1 state–of–the–art word recognizers do not make use of prosodic information although prosody influences acoustic–phonetic properties of speech. In Section 4.3.1 a couple of studies have been listed which tried to use prosodic information either directly in the word recognizer or in a pre– or postprocessing phase. We also conducted a few preliminary experiments concerning the improvement of word recognition by prosodic information. However, with the currently available data no positive results could be obtained so that we did not spend more effort on this topic. Nevertheless, we will briefly summarize what has been investigated because the methods are directly based on the approaches

[4] We are grateful to Pablo Fetter for providing us with these word graphs.

described in Sections 6.4 and 7.2 and it is obvious to try this.

First, we assumed that wrong word hypotheses would be a bad basis for the computation of prosodic features so that in some cases an NN boundary or accent classifier either would not be able to decide clearly for one of the classes or that the mismatch between NN and polygram would be greater on wrong than on correct word hypotheses. By the statement 'the NN cannot clearly decide' we mean the NN would compute probabilities which are on the average closer to 0.5 as has been shown in Section 6.4.4. A mismatch between NN and polygram means that the NN determines a high probability for a certain class, which in turn has a low probability according to the polygram. The combination of both classifiers, cf. equation 7.13, would result in a probability which is closer to 0.5 than in the cases where the NN and the polygram decisions match. Therefore, we prosodically scored the correct word chains on the one hand and the first–best recognized word chains on the other hand. This was conducted on 328 word chains computed by Universität Karlsruhe for part of the VERBMOBIL CD 4, which by that time was not contained in the training corpus of the word recognizer. The word accuracy on these word chains was 67%. Unfortunately, we found that there is no significant difference in either of these types of prosodic probabilities between correct and recognized word chains. The reason for this might be that incorrect word hypotheses in the word graph usually are phonetically similar to the correct ones and/or have similar linguistic properties with respect to the n-gram language model.

Another series of experiments was motivated by the very good results we achieved by the use of an NN for classifying pitch period hypotheses as false or correct [Har95]. Inspired by this, we conducted the following experiments using the same word graphs as in Section 8.1 of the LING_BIG corpus. For this particular experiment we used the 235 word graphs of 11 dialogs for training, and the 80 word graphs of 3 dialogs for testing. We computed for each of the word hypotheses in the word graph acoustic–prosodic features as described in Sections 8.1 and 6.4.2. With the help of a Viterbi search the best fitting path in the word graph, that is, the one with the best word accuracy, was determined. All correct word hypotheses on this path were defined as to be correct, all other word hypotheses in the word graph were defined to be false. Then we trained an NN using prosodic features to discriminate between correct and false word hypotheses. We ran a large number of experiments using different feature sets computed over differently large contexts of the word to be classified; different NN topologies were tried. The best result was a recognition rate of 62% with an average of the class–wise recognition rates of 75%. This is far too low for a verification of word graphs. Therefore, this approach was not investigated any further.

8.3 Syntax

In this section we will show how prosodic boundary information is used in the syntactic processing of the VERBMOBIL system. It is based on the prosodically scored word graphs as described in Section 8.1. For the time being we concentrate on the use of prosodic–syntactic clause boundaries because their disambiguation was considered as most important with respect to the syntactic analysis. In the syntax module a prosodic–syntactic clause boundary will be represented by the PSCB symbol. During syntactic analysis a PSCB hypothesis will be built based on a boundary score computed by a classifier integrated in the prosody module. Classifiers can for example be a B3/B[029]–NN, an M3/M0–LM or a combination of both, cf. Section 6.4.3. A turn is segmented into several clauses or clause–like units by these PSCB symbols. In the following we will refer to this by the term *segmentation*.

In the VERBMOBIL system alternatively the syntax module of Siemens, München, or the one of IBM, Heidelberg can be used, cf. Section 3.4. Both systems make use of the PSCB symbols. The two different approaches will be explained in this section. The Siemens parser, cf. Section 8.3.2, operates directly on word graphs. This is achieved by integrating the parser in an A* left–to–right search for the optimal word chain. After each word hypothesis, both alternatives, a PSCB or no–PSCB following the word, are considered. The search is guided by a combination of scores including the prosodic scores. The IBM approach, cf. Section 8.3.3, is a two stage solution: first a search procedure determines the n-best word chains by also using a combination of scores which includes the prosodic boundary scores. These word chains contain the PSCB symbols. Each pair of chains differs either in at least one of the words or in the position of one PSCB symbol. The parser then operates on these word chains.

The most effective use of prosodic information in the VERBMOBIL system is the integration of prosodic clause boundary scores in the search of the Siemens parser as we will see in Section 8.3.2. Before we explain the details of this algorithm we discuss in general how this prosodic information can contribute to the parsing of word graphs.

8.3.1 General Considerations

In Section 4.2.3 we discussed the potential use of prosodic phrase boundaries in syntactic analysis. It was pointed out that two aspects are important: prosody can

- *disambiguate* the syntactic structure of an utterance, and it can

- enhance the *intelligibility* by grouping words which belong together.

Here we want to explain the impact of these general observations on the automatic syntactic analysis of word graphs. We believe that both aspects are relevant. In this context the enhancement of intelligibility corresponds to a *speed–up* of parsing. In the following we want to explain this using the two example word graphs depicted in Figures 8.3 and 8.5. These word graphs were generated with the word recognizers of Daimler Benz and Universität Karlsruhe, respectively, and then they were prosodically scored by our prosody module. Both parsers used in VERBMOBIL skip the edges in the word graph which are labeled with pause, non–verbals or garbage symbols. Recall that the edges of the word graphs in the figures are labeled with the most probable prosodic class; the no–boundary and no–accent class symbols have been omitted for clarity in the graphics. However, keep in mind that the prosody module in the VERBMOBIL system attaches probabilities for all of the classes to the edges in the word graph allowing all alternatives to be potentially considered.

Disambiguation

In particular, since we have to deal with spontaneous speech, the situation gets complicated. The syntax has to model elliptic utterances and free phrases, which are clause level units. Interruptions of syntactic constructions occur. Furthermore, the word order in German in general and even more in spontaneous speech is rather free. As a consequence many words or word groups can form a free phrase. Hence, without prosodic information the search branches after almost every word boundary considering both cases, a PSCB or no–PSCB being at this position. As a result the search space gets very large. This should become a bit clearer when looking at the word graph depicted in Figure 8.3. It corresponds to the following utterance:

<ähm> Mittwoch den sechsten geht nicht. <Äh> Montag der elfte? (8.2)
<ehm> Wednesday the sixth is not okay. <Eh> Monday the eleventh?

In the figure the best matching word chain is indicated by solid bold edges. Recall that "best matching" refers to the spoken word chain, cf. Section 3.2. In this example it differs from the spoken word chain only in the non–verbals, which has no influence on the subsequent analysis. Without prosodic boundary information, this word chain has many different readings, a few of which are shown in Figure 8.4, where the non–verbals have been omitted for clarity. Some of the readings given in the figure also differ in the meaning, which becomes clear when looking at the

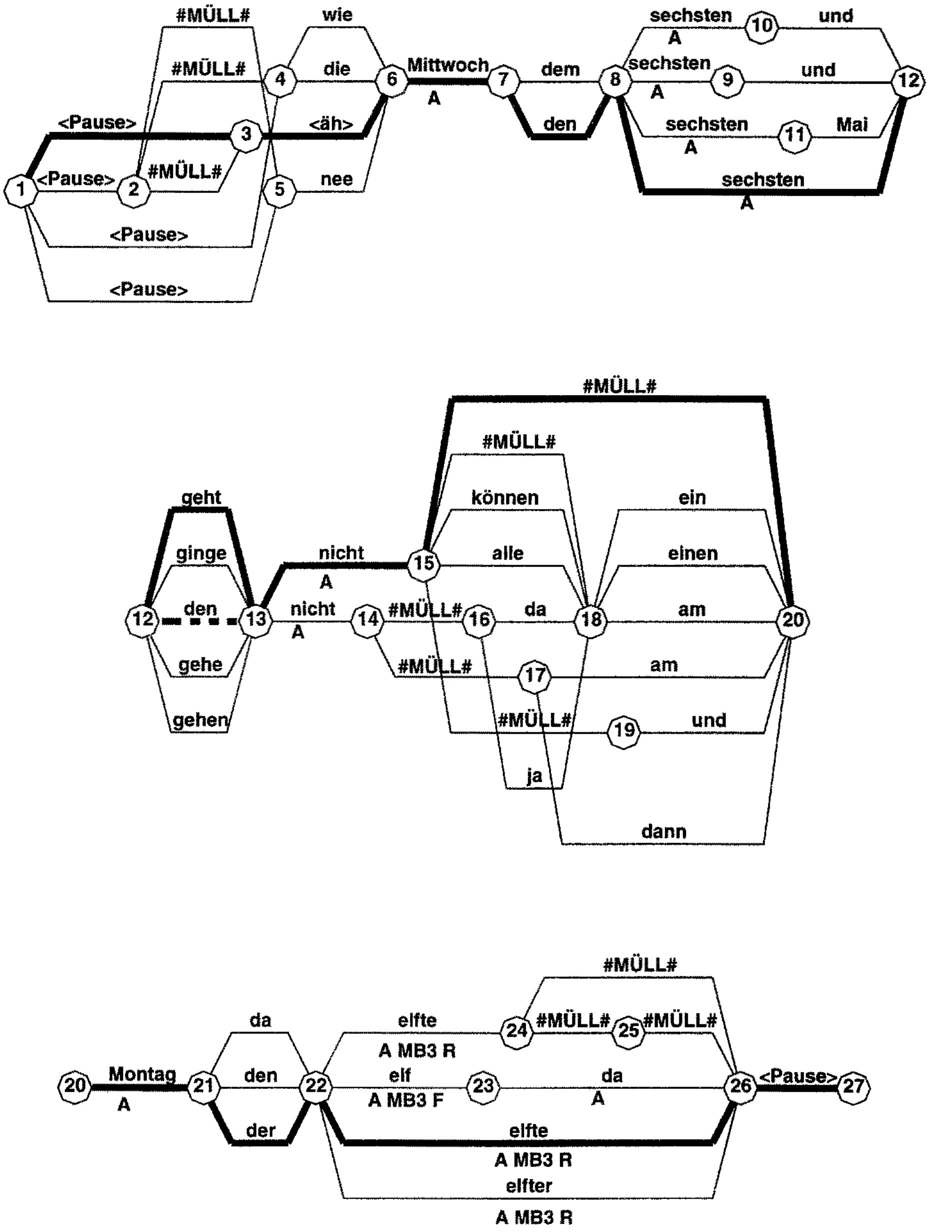


Figure 8.3: Part of a prosodically classified word graph. It was generated with the VERBMOBIL recognizer of Universität Karlsruhe for a turn of the VERBMOBIL corpus. For the prosodic scoring the “B3/B[029]–NN & M3/M0–LM$_3$” classifier was used; therefore, the edges are labeled with MB3.

Mittwoch den sechsten geht nicht. Montag der elfte?
Wednesday the sixth is not okay. Monday the eleventh?

Mittwoch den sechsten? Geht nicht! Montag der elfte?
Wednesday the sixth? That's not okay! Monday the eleventh?

Mittwoch? Den sechsten? Geht. Nicht! Montag. Der elfte?
Wednesday? The sixth? Is okay! Not! Monday. The eleventh?

Mittwoch den sechsten? Geht nicht Montag der elfte?
Wednesday the sixth? Wouldn't Monday the eleventh be okay?

Mittwoch den sechsten geht, nicht Montag der elfte.
Wednesday the sixth is okay, not Monday the eleventh.

Figure 8.4: Different readings of the same word chain contained in the word graph depicted in Figure 8.3. The differences are indicated by punctuation, which directly relates to the position of MB3 boundaries and the shape of the intonation contour.

translation. Not all of these readings are meaningful in the given context. However, since we deal with spontaneous speech, they have to be considered as being syntactically correct. Recall that the syntactic and the semantic modules currently work strictly bottom–up within the VERBMOBIL system. Therefore, meaningless segmentations of turns into clause–like units cannot be ruled out during syntactic analysis.

When probabilities for PSCB are computed by the prosody module, most of these readings become rather unlikely. In this example the classification result of the prosody module on the best fitting word chain corresponds to the following punctuation:

<äh> Mittwoch den sechsten geht nicht #MÜLL# Montag der elfte? (8.3)
<eh> Wednesday the sixth is not okay #GARBAGE# Wednesday the eleventh?

This is not a syntactically correct sentence because of the missing period (labels MB3 F) or question mark (labels MB3 R) after nicht. This should make it obvious that the prosody module may not perform any hard decisions. Therefore, we compute probabilities so that during syntactic analysis in principle all alternatives are explored. The prosodic scores guide the search by ranking the alternatives. In our example, the parser will reject the word chain immediately after having consumed the word Montag and it will draw the analysis to another alternative.

These principles are independent of the particular parser. In the case of the word graph parser, partial analyses containing no PSCB after nicht will be rejected and others, which include the hypothesis of a PSCB after the word nicht, will be further considered. We will explain this in more detail in Section 8.3.2. In the case of the parser operating on the n-best word chains, the analysis of the word chain of example (8.3) will fail, and the next word chain will be considered. However, there is a major drawback of the approach using the n-best word chains: for efficiency reasons n has to be limited ($n = 100$ in the VERBMOBIL system). Therefore, it is likely that the correct word chain with the correct segmentation is not among these alternatives. The word graph parser is more efficient and can explore a larger number of alternatives, because it has not to redo the same analysis on the same prefixes of similar word chains.

Note that above we only considered the best matching word chain. However, during the analysis this path in the word graph has to compete with alternatives. Some of the alternative paths may also correspond to analyzable word chains. The following example shows an alternative word chain contained in the word graph of Figure 8.3:

Mittwoch den sechsten? Den nicht. Montag der elfte? (8.4)
Wednesday the sixth? Not this one. Monday the eleventh?

This is a syntactically correct alternative, and it is even meaningful. It only differs by the word den (dashed bold edge) from the correct path. Which of the word chains in the graph and which segmentation eventually will be the result of the syntactic analysis depends on the different scores used during the search. As for the prosody we can say that example (8.4) is less likely than the correct word chain given in example (8.2), because (8.4) additionally requires a PSCB symbol after the word sechsten, however, as can be seen in Figure 8.3 a no–PSCB symbol is more likely after sechsten.

Speed–up of Parsing

The importance of prosody in comparing competing word chains contained in a word graph might become more obvious when looking at a different word graph, which is depicted in Figure 8.5. This word graph corresponds to the following

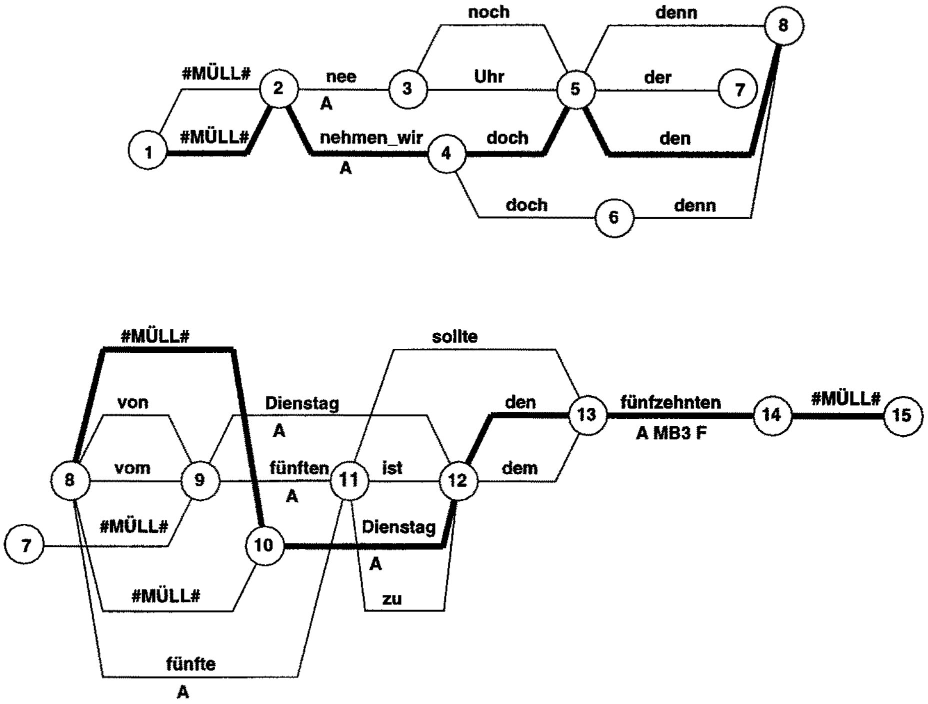


Figure 8.5: A prosodically classified word graph, which was generated with the VERBMOBIL recognizer of Daimler Benz for a turn of the VERBMOBIL corpus.

utterance:

n <Atmung> nehmen wir doch den +/fünf=/+ Dienstag den fünfzehnten <Atmung> (8.5)
(*n <breathing> take we still the +/five=/+ Tuesday the fifteenth <breathing>*)
why don't we take the +/fif=/+ Tuesday the fifteenth

The best fitting word chain is indicated by the bold edges in the figure. Note that the word chain is not correct in terms of word accuracy, because the word fünf (*five*) has been recognized as garbage, however, with respect to the subsequent syntactic analysis this is even better, because fünf in this case is an interruption of fünfzehn that has been corrected by the speaker; it should therefore not be interpreted or translated by the VERBMOBIL system. The word fragment n at the beginning of the turn has correctly been recognized as garbage.

Figure 8.6 shows a subset of partially correct word chains contained in the word graph. The garbage symbols have been omitted for clarity. The punctuation reflects expectations about the prosodic boundary and intonation contour classes.

Nehmen wir. Doch der Dienstag, dem
We take it. Nevertheless the Tuesday, on which

Nehmen wir doch, denn von Dienstag dem fünfzehnten
Nevertheless, we take it, because from Tuesday the fifteenth

Nehmen wir doch den. Von Dienstag dem fünfzehnten
Nevertheless, let us take that one. From Tuesday the fifteenth

Nehmen wir doch den vom fünften. Sollte
Nevertheless, let us take the one of the fifth. Should

Nehmen wir doch den vom fünften. Ist den fünfzehnten
Nevertheless, let us take the one of the fifth. Is the fifteenth

Nee. Noch der Dienstag
No. Still the Tuesday

Nee. Noch den vom fünften
No. Still the one of the fifth

Figure 8.6: Some of the syntactically correct partial word chains contained in the word graph of Figure 8.5.

None of these word chains can be part of a successful syntactic analysis, because there is either no syntactically correct continuation possible on the basis of the word graph or the end of the word graph is reached without a completion of the syntactic analysis being possible. All of these examples are garden path problems, because the solution/no–solution decision can only be made very late.

Without PSCB information all of these alternatives are likely to be considered by the parser. Of course eventually it will discard the wrong ones, but only after having wasted a lot of time on its analysis. When using probabilities for PSCB computed by the prosody module, all of these word chains will be analyzed with low priority, because they require a PSCB where none has been detected as can be seen when looking at the prosodic labels given in Figure 8.5. In most cases, prosody will contribute already very early during analysis to a bad score of these paths. By this a considerable speed–up is achieved. For the parser of the n-best word chains this means that prosodically unlikely word chains get a low rank. In the case of the word graph parser the partial analysis of the respective word chain is considered with low priority as soon as a boundary is encountered which received a low PSCB probability by the prosody module.

It should be clear now that the word graph in Figure 8.5 is an example where prosody does not contribute to disambiguation, but where it supports syntactic analysis in a way that a successful analysis is achieved in much less time. This corresponds to our hypothesis stated in Section 4.2.3 that prosody in human–human communication often "just" facilitates syntactic analysis by structuring the utterance. In a way this speed–up is also achieved by some kind of disambiguation, however, it does not disambiguate between two alternative readings of the same word chain, but it rather early rules out alternatives which anyway (but later) would be identified as syntactically incorrect.

After these considerations, we will describe exactly how the two parsers employ prosodic boundary information. Furthermore, results obtained on VERBMOBIL spontaneous speech data will be presented.

8.3.2 Parsing of Prosodically Scored Word Graphs

In this section the use of PSCB symbols and prosodic clause boundary scores in the word graph parser and the TUG grammar formalism developed by our colleagues from Siemens, München, is described[5]. This is the most important contribution prosodic information provides within the VERBMOBIL system. An overview of the system was already given in Section 3.4. The integration of a previous version of this parser in an A* search has been described in [Sch94]. The search algorithm has been extended in a joint effort so that prosodic boundary information can be used. The basic ideas and preliminary results have already been published in [Bak94]. The essential part concerning the use of prosody is the prosodic scoring of word graphs as described in Section 8.1. The extensions to the search algorithm and experiments will be described in the following.

Using Prosodic–Syntactic Boundaries in Grammar and Parser

A context–free grammar for spontaneous speech has to allow for a variety of possible input phrases following each other in a single utterance; the relevant part of the grammar is depicted in Table 8.5. The first rule defines that the input word sequence is split up into phrases. Among those phrases are normal sentences (rule 2),

[5] We want to thank especially Gabi Bakenecker, Hans–Ulrich Block, Stefanie Schachtl and Tobias Ruland for the close cooperation. They changed the grammar and modified the parser implementation so that PSCB symbols can be treated. They also conducted the actual parsing of the word graphs which we before had scored prosodically. The experiments described below were jointly planned and conducted.

rule 1:	S	→	PHRASE	S
rule 2:	PHRASE	→	SENTENCE	PSCB
rule 3:	PHRASE	→	TOPIC_ELLIPSIS	PSCB
rule 4:	PHRASE	→	ELLIPTIC_PHRASE	PSCB
rule 5:	PHRASE	→	EXCLAMATIVE	PSCB
rule 6:	PHRASE	→	EXCLAMATIVE	

Table 8.5: Part of the grammar which takes prosodic–syntactic clause boundaries (PSCB) into account. The first rule, for example, means that the start symbol S is replaced by the non–terminal symbol PHRASE and S. PHRASE represents any kind of clause–like phrase.

sentences with topic ellipsis (rule 3), elliptical phrases like prepositional or noun phrases, (rule 4) or exclamative phrases (rule 5 and rule 6). These phrases were classified as to whether they require an obligatory or an optional PSCB behind them. The grammar fragment in Table 8.5 defines that only exclamatives, for example, the word oh, may optionally be followed by a PSCB. This is achieved by the two rules 5 and 6. Rule 5 requires a PSCB symbol after the exclamative, whereas rule 6 does not contain the PSCB symbol. Sentences, where parts of the "Vorfeld"[6] are missing, are called topic ellipsis, for example, the sentence [das] finde ich gut (*I like [it]*) turns to a topic ellipsis, if das is omitted. An elliptic phrase is any group of words, which cannot be identified as being a part of a sentence, for example, tomorrow in Tomorrow? Monday would be better.

The analysis of word graphs is based on such a grammar, where the parser is integrated in an A* tree search, cf. Section 2.6. The structogram of the A* algorithm for this particular application is shown in Figure 8.7. The analysis is performed left–to–right starting at the first logical node in the word graph (l_1). A search space node n_i corresponds to a cycle–free path in the word graph starting at l_1 and, additionally, one and only one segmentation of the word chain underlying the path by PSCB symbols. In the following, we will denote as *symbol chains* word chains in which PSCB symbols have been inserted. A symbol chain associated with the search space node n_i is denoted as $chain(n_i)$. For example, the word graph in Figure 8.5 contains among others the following symbol chains, which begin at the logical word graph node number two:

nee noch	*no still*	(8.6)
nee Uhr	*no o'clock*	(8.7)
nee PSCB	*no* PSCB	(8.8)

[6] *Vorfeld* is the part of the matrix clause being before the finite verb.

```
Definitions:
$l_1, l_L$: the first/last logical node in the word graph.
$end\_node(n_i)$: returns the logical word graph node of the final word hypothesis contained in $n_i$.
$n_i$: A search space node; it corresponds to a cycle–free path in the word graph starting at $l_1$ and one and only one segmentation of the underlying word chain by PSCB symbols.
$chain(n_i)$: returns the word chain including the PSCB symbols which corresponds to $n_i$.

Initialization:
OPEN $= \{n_i|$ consisting of a single word hypothesis$\}$.
Compute $\widehat{\varphi}(n_i)$ for all $n_i \in$ OPEN.

WHILE OPEN $\neq \emptyset$
    Determine $n_i$ with minimal $\widehat{\varphi}(n_i)$
    OPEN $=$ OPEN$\setminus\{n_i\}$
    Analyze $chain(n_i)$ with the parser, i.e. extend each of the partial analyses associated with the predecessor of $n_i$.
    IF at least one successful partial analysis of $chain(n_i)$ exists
    THEN IF $end\_node(n_i) = l_L$
         THEN Output $chain(n_i)$ and all parse trees which are the result of the analysis of $chain(n_i)$.
              STOP
         ELSE Store all partial analyses of $chain(n_i)$ in the stack associated with $n_i$.
              Expand $n_i$:
              A successor $n_j$ is the continuation of $n_i$ by a word hypothesis, which starts at logical node $l_j = end\_node(n_i)$ OR the continuation by a PSCB hypothesis. $n_i$ can only be extended by a PSCB if the last symbol in $chain(n_i)$ is not PSCB.
              FOR each successor $n_j$ of $n_i$
                  Compute $\widehat{\varphi}(n_j)$
                  Set a pointer associated with $n_j$ to $n_i$.
                  OPEN $=$ OPEN $\cup\, n_j$.
```

Figure 8.7: Adaptation of the A* algorithm to word graph parsing.

Each of these chains represents a different node (or state) in the search space. With each node n_i a set of stacks is associated; a stack is a data structure representing one of possibly alternative partial analyses. This allows for efficient parsing, because as soon as a node n_i is expanded only these partial analyses have to be extended.

As can be seen in Figure 8.7, at each step of the search the best scored entry, that is, the one with minimal costs $\widehat{\varphi}(n_i)$, is removed from OPEN and analyzed by the parser. If the analysis fails, this hypothesis is discarded. If at least one partial analysis could successfully be extended, the corresponding word chain is expanded by the succeeding hypotheses in the word graph and additionally by the PSCB symbol, cf. example (8.8). All these extended word chains are scored and inserted in OPEN. To give an example, assume during the analysis of the word graph from Figure 8.5 a node n_i corresponding to the "chain" nee (*no*) is at some time of the analysis the best scored node in OPEN, cf. Figure 8.8. Because nee can be an elliptic sentence one partial analysis is successful. If the analysis of a node does not fail, the node is expanded. In our example, this means that three successor nodes of n_i are established, which correspond to the symbol chains shown in examples (8.6) to (8.7). After the expansion of a node, the successor nodes are scored and inserted into OPEN.

The overall score $\widehat{\varphi}(n_i)$ is a weighted sum of

- the acoustic scores of the word hypotheses as provided by the HMM word recognizer,
- the n-gram score for the word chain (since very recently a trigram is used),
- the probabilities $P_{W_r}(\text{PSCB})$ computed by the prosody module at each word hypothesis W_r according to equation (7.13), and
- appropriate remaining costs $\widehat{\chi}(n_i)$, which consider all these partial scores.

In the following we want to give a more formal definition of the costs $\widehat{\psi}(n_i)$. Assume that the node n_i corresponds to the path in the word hypotheses $W_1, \ldots, W_r$. Three cases have to be distinguished. In the first case, n_i is extended by a word hypothesis W_{r+1} yielding node n_j, which then represents $W_1, \ldots, W_r, W_{r+1}$. The symbol chain of n_j is a concatenation of $chain(n_i)$ and $v_{h(r+1)}$, where $v_{h(r+1)}$ is the word associated with the word hypothesis, cf. Section 3.2. If n_i corresponded to nee, then nee noch (8.6) would for example correspond to n_j. The costs of n_j are computed as

$$\begin{aligned}\widehat{\psi}(n_j) \;=\; & \widehat{\psi}(n_i) \\ & -log(p({}^{t_{b(r+1)}}\boldsymbol{c}, \ldots, {}^{t_{e(r+1)}}\boldsymbol{c} \,|\, \boldsymbol{\pi}_{h(r+1)}, \boldsymbol{A}_{h(r+1)}, \boldsymbol{B}_{h(r+1)})) \\ & -\xi_1 log(P(v_{h(r+1)} \,|\, v_{h(r-1)} v_{h(r)})) \\ & -\xi_2 log(1 - P_{W_r}(\text{PSCB})) \ . \qquad (8.9)\end{aligned}$$

In this case it is assumed that the last symbol in $chain(n_i)$ is not PSCB.

Second, n_i is extended by a PSCB hypothesis. This can only be done if the last symbol in $chain(n_i)$ is a word and not a PSCB. In this case the successor node n_l represents the same sequence of word hypotheses as n_i, and the symbol chain is a concatenation of $chain(n_i)$ and PSCB. The costs of n_l are computed as

$$\begin{aligned}\widehat{\psi}(n_l) &= \widehat{\psi}(n_i)\\ &\quad -\xi_2 log(P_{W_r}(\text{PSCB})) \ .\end{aligned} \tag{8.10}$$

In the third case, a node n_l corresponding to $W_1, \ldots, W_r$ with PSCB being the last symbol in $chain(n_l)$ is extended by a word hypothesis W_{r+1}. The resulting node is n_m. This time no prosodic costs are considered yielding the following cost function:

$$\begin{aligned}\widehat{\psi}(n_m) &= \widehat{\psi}(n_l)\\ &\quad -log(p({}^{t_{b(r+1)}}\boldsymbol{c}, \ldots, {}^{t_{e(r+1)}}\boldsymbol{c}|\boldsymbol{\pi}_{h(r+1)}, \boldsymbol{A}_{h(r+1)}, \boldsymbol{B}_{h(r+1)}))\\ &\quad -\xi_1 log(P(v_{h(r+1)}|v_{h(r-1)}v_{h(r)})) \ .\end{aligned} \tag{8.11}$$

It is important that not only in equation (8.10) but also in (8.9) prosodic scores are used, because then with (8.11) and (8.10) one achieves

$$\begin{aligned}\widehat{\psi}(n_m) &= \widehat{\psi}(n_i)\\ &\quad -log(p({}^{t_{b(r+1)}}\boldsymbol{c}, \ldots, {}^{t_{e(r+1)}}\boldsymbol{c}|\boldsymbol{\pi}_{h(r+1)}, \boldsymbol{A}_{h(r+1)}, \boldsymbol{B}_{h(r+1)}))\\ &\quad -\xi_1 log(P(v_{h(r+1)}|v_{h(r-1)}v_{h(r)}))\\ &\quad -\xi_2 log(P_{W_r}(\text{PSCB})) \ ,\end{aligned} \tag{8.12}$$

which is equivalent to (8.9). The only difference between nodes n_j and n_m lies in the segmentation of the same word sequence, which is reflected in the cost functions (8.9) and (8.12), which only differ in whether $log(1 - P_{W_r}(\text{PSCB}))$ or $log(P_{W_r}(\text{PSCB}))$ is used.

The probability $log(P_{W_r}(\text{PSCB}))$ in the equations above is computed by the prosody module using the algorithm of Section 8.1. Recall that different classifiers can be used for the computation of this probability. If, for example, a B3/B[029]–NN is used, then $log(P_{W_r}(\text{PSCB})) = log(P_{W_r}(\text{B3}))$. If the probabilities are computed by the combined "B3/B[029]–NN & M3/M0–LM$_3$" classifier, then $log(P_{W_r}(\text{PSCB}))$ is equal to $log(P_{W_r}(\text{MB3}))$.

The weights ξ_1 and ξ_2 normalize the variances of the trigram and prosodic scores, respectively. They were determined heuristically in the course of the experiments conducted by our colleagues at Siemens. The actual weights depend on the application domain and the type of prosodic classifier used in the prosody module.

The remaining costs $\hat{\chi}(n_i)$ are determined in the course of a backwards Viterbi search prior to the A* search. The Viterbi search does not integrate the parser, but it uses the same cost functions as defined above. With this, for each logical node in the word graph the least lower bound for the remaining costs can be determined in the case a bigram language model is used. In the experiments described below a trigram language model was used. This might in rare cases result in non–admissible remaining costs, however, overall better results could be achieved [Rul96].

An example for a sequence of analysis steps of the word graph depicted in Figure 8.5 is shown in Figure 8.8. The left–hand side shows four possible subsequent stages of OPEN during an analysis. The entries in OPEN are ranked by their score. The best scored hypothesis is at the top. These analysis steps shown in the figure do not correspond to an actual analysis. They are hypothetical but realistic. In the first step the node corresponding to the word nee (*no*) is popped from OPEN. The partial analysis by the parser is successful so that the node is expanded. The successor nodes corresponding to the chains nee Uhr, nee noch, and nee PSCB are scored and inserted into OPEN. The probability for PSCB at nee is close to zero. Therefore, nee PSCB gets a rather low rank and in turn the other two nodes get a high rank. In the next analysis step the node corresponding to nee noch is removed from OPEN and analyzed by the parser. Since this is not a grammatically correct beginning of a (spontaneous speech) sentence, this hypothesis is discarded. This is the crucial step of the analysis were prosody plays an important role. The hypothesis nee PSCB noch is syntactically correct and would be parsed successfully. However, due to the prosodic score nee PSCB got a bad rank and might, therefore, not or rarely be considered any further. This results in an overall speed–up of the analysis, since nee PSCB does not lie on a solution path. Note that also the word chain nee Uhr is not syntactically correct, whereas nee PSCB Uhr could be the beginning of a syntactically correct symbol chain at least in the context of spontaneous speech.

A drawback of our approach is the sequential bottom–up analysis of the prosody module and the parser. Recall that the probability $p(\text{PSCB}\,|W_r)$ is a combination of the acoustic–prosodic probability and the prosodic polygram probability. For the computation of both, word hypotheses in the context of W_r have to be chosen. These will in most cases differ from the actual word hypotheses in the context of W_r being in one node of the A* search. Therefore a better solution would be the integration of the prosodic processing within the A* search, because then the words in the left–context would be given in an optimal way by the path corresponding to a search space node. Apart from technical difficulties, for the time being it was considered as too inefficient to integrate the acoustic–prosodic processing in the A*

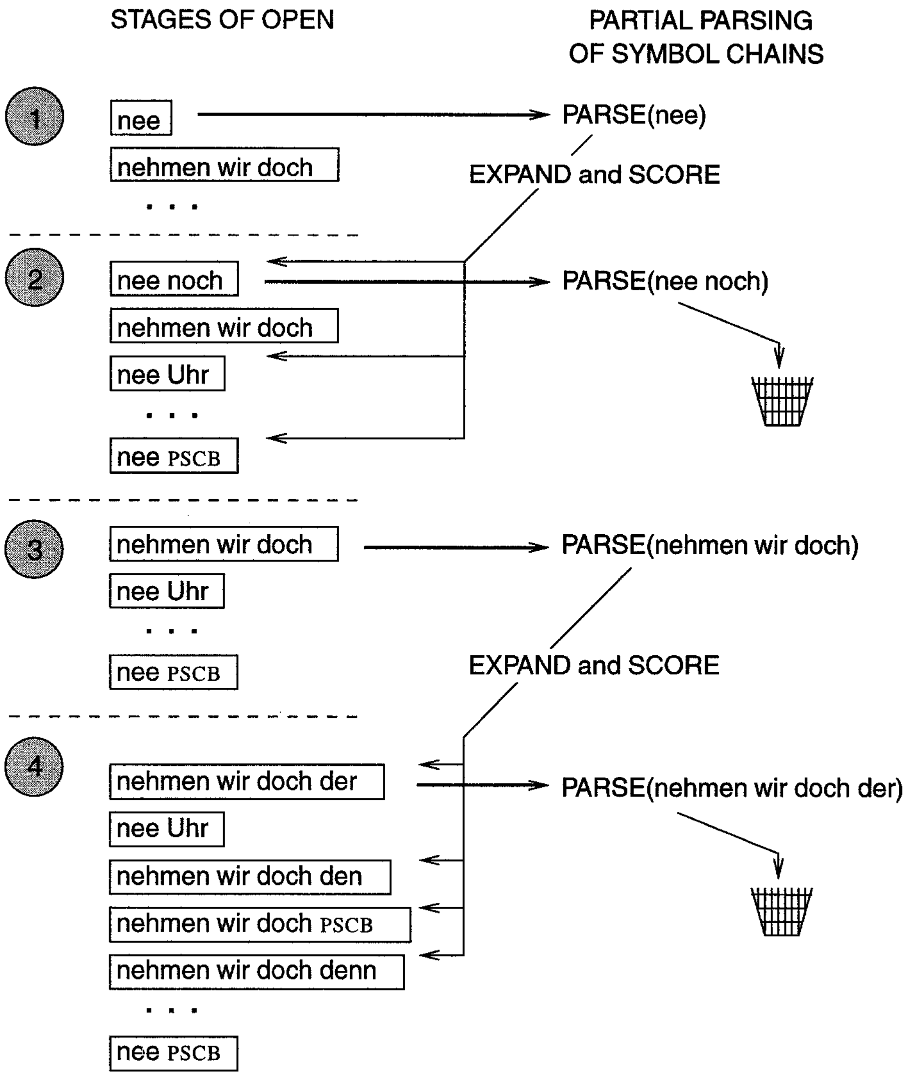


Figure 8.8: Example for subsequent analysis steps of the A* parser given the word graph depicted in Figure 8.5.

search, because that would mean that for each word hypothesis prosodic scores for different contexts would have to be computed. As for the prosodic polygram classifier, we found it important to consider n-grams beyond bigrams, cf. Section 8.1. However until very recently, the syntax module described above could only apply bigram models.

Experiments and Results

We will now report on a number of experiments we performed on ERBA as well as on VERBMOBIL using different sets of word graphs. At first we performed experiments on validation data which lead to the final version of the prosodic clause boundary classifier which was eventually integrated in the VERBMOBIL system. For this classifier we also give results on independent test data. The different results also show that the approach is independent from the word recognizer which computes the word graphs.

In the following, by *analyzed successfully* we mean that a path in the word graph was found within a time limit of 30 secs, the path had to lead from the first to the last node in the graph and the underlying symbol chain had to be completely analyzable by the parser. The latter means that at least one parse tree could be established for the path. The parsing times given below are partly greater than 30 secs. This is due to the fact that after the parser has found a path and at least one parse tree, for all of the different readings the variable bindings have to be resolved. This can take a considerable amount of time [Sch96a]. The *parse time* refers to the overall CPU time needed for search and parsing. The number of *syntactic readings* denotes the number of alternative parse trees on the word chain, which were obtained as a result of the search. In the experiments done *without prosody* at each word hypothesis the probability for both, PSCB and no–PSCB, is simply set to 0.5.

We have already published encouraging results on the ERBA corpus using this approach [Bak94]. Since the ERBA sentences are grammatically well–formed, the PSCBs were considered to be useful only in the case of sentences consisting of a main clause and an infinitive subordinate clause. Therefore, we conducted experiments mainly on the ERBA_INF sub–corpus. Word graphs for this corpus were provided by Daimler Benz, Ulm[7]. They consisted of about 10 word hypotheses per spoken word; 173 out of the 242 word graphs contained the spoken word chain. The word graphs were annotated with probabilities for prosodic boundaries as described in Section 8.1 using a preliminary version of the NN described in Section 6.4. This NN yielded a recognition rate of 70% for the three boundary classes B[01], B2, and B3. For the use in these experiments the probabilities for B[01] and B2 were summed up. The recognition rate of the resulting two class recognition problem is 82%. No polygrams were used at that time. Note that these experiments were conducted with an older version of grammar and parser in comparison to the experiments on VERBMOBIL data described below. Furthermore, in the A* search at that time only a bigram language model was used.

[7] We wish to thank Alfred Kaltenmeier and Peter Regel–Brietzmann for providing us with these word graphs.

	B3/B2/B[01]–NN	without prosody
# of successful analyses	144	145
average # of syntactic readings	2.7	9.0
average parse time (secs)	5.3	5.8

Table 8.6: Parsing results on 242 ERBA word graphs using the TUG system.

In Table 8.6 the parsing results on these 242 word graphs are summarized. Without the use of prosody the analysis yielded 3.3 times as many syntactic readings as with the use of PSCBs. The parse time remained constant on the average. With the use of prosodic boundary information one of the word graphs could not be parsed, which was successfully analyzed without the use of PSCBs. In the context of a full dialog system, in this case the user would have to be requested to repeat his utterance. However, the time needed for additionally analyzing this repetition is negligible compared to the overall speed–up of the system by the use of prosody. Apart from this we also parsed the spoken word chains of the remainder of the ERBA corpus. It turned out that on these sentences it made no difference in the number of readings and in the parsing time whether prosodic information was used or not.

Recently, we conducted experiments on VERBMOBIL corpora. If not stated otherwise the results will also appear in [Nöt96b]. In these experiments, the final versions[8] of the VERBMOBIL word recognizers, prosody module, and syntax module were used. Table 8.7 summarizes a few figures concerning the different word graph sets used in these experiments[9]. First, a large number of experiments was conducted using the LING_BIG sub–corpus to find the optimal configuration of prosody module and parser. Recall that part of this corpus was also used for the training of the B3/B[029]–NN. A final test was performed on an independent testing set using the LING_TEST sub–corpus. As for syntax and prosody this is entirely new data. However, the word recognizers were trained on all available VERBMOBIL data including LING_TEST.

For the experiments on LING_BIG Siemens decided to use only 274 of the 320 utterances. The remaining 46 utterances were not considered in the experiments because they contained only noise or isolated time–of–day expressions, greetings,

[8] The final versions of the different modules were integrated in the VERBMOBIL (phase 1) prototype system in July and August 1996.

[9] We wish to thank Thomas Kemp, Universität Karlsruhe who provided us with the KA–LING_BIG word graphs. The DB–LING_BIG were kindly generated by Pablo Fetter, Daimler Benz Ulm. Many thanks also to Andreas Klüter, DFKI Kaiserslautern, who generated the KA–LING_TEST word graphs with the VERBMOBIL prototype system using the word recognizer of Universität Karlsruhe.

acronym /		word graph characteristics			
sub–corpus	recognizer	hyp./word	total #	# correct	*RA*
KA–LING_BIG	Univ. Karlsruhe	12.2	274	128	91.3
DB–LING_BIG	Daimler Benz	8.4	274	108	91.4
KA–LING_TEST	Univ. Karlsruhe	9.3	594	117	73.3

Table 8.7: VERBMOBIL word graphs used for experiments with the TUG word graph parser. Note that the sub–corpus LING_BIG is used for validation purposes and is not completely independent from the training data, cf. Section 5.2.1. LING_TEST is independent test data used for a final evaluation.

or proper names. The KA–LING_BIG word graphs were also used for the evaluation presented in Section 8.1.

The classifiers used in the prosody module were initially optimized with respect to a minimal error rate. However, with respect to the syntactic analysis it could be useful to recognize more MB3 causing more false–alarms, or vice versa it might be better to reduce the number of false–alarms at the expense of a lower number of recognized MB3. This can be controlled by multiplying the clause boundary probability computed by the prosody module according to equation (7.13) with a heuristic weight ξ_3 and by then performing a normalization to yield probabilities:

$$P_{v_h}(\text{PSCB}) = \frac{\xi_3 P_{v_h}(\text{MB3})}{\xi_3 P_{v_h}(\text{MB3}) + (1-\xi_3) P_{v_h}(\text{MB0})} \quad . \tag{8.13}$$

In a first series of experiments, we heuristically optimized this weight. The main aspect was to find an optimal segmentation of the word chains determined in the A* search; the parsing time was not considered in these experiments. We used the “B3/B[029]–NN & M3/M0–LM$_3$” classifier in these experiments. The grammar writer at Siemens evaluated the parsing results for different values of ξ_3 manually by inspection of the parser output. The value $\xi_3 = 0.5$ is the default, because then Bayes classification is performed; cf. Section 2.1; this was used in all the evaluations presented in Sections 6.4.3 and 8.1. It turned out that $\xi_3 = 0.3$ is the best choice for the parser. The segmentation was judged to be better in 4% of the cases compared to $\xi_3 = 0.5$. None of the parser outputs was considered to be worse. This weight $\xi_3 = 0.3$ was used in all of the following experiments and in the prosody module integrated in the VERBMOBIL system.

With a preliminary version of the TUG system, and using the grammar rules shown in Table 8.5, 149 KA–LING_BIG word graphs could be analyzed successfully, cf. Table 8.8. In 28 of the 125 word graphs which were not analyzed the spoken word chain was contained, and vice versa, 17 word graphs which could be analyzed successfully did not contain the spoken word chain. One might argue that

grammar	# of successful analyses
(1) with optional PSCBs after exclamatives and obligatory PSCBs after every other phrase	149
(2) with obligatory PSCBs after every phrase	79

Table 8.8: Number of successfully parsed KA–LING_BIG word graphs using a preliminary version of the TUG system and the "B3/B[029]–NN & M3/M0–LM$_3$" classifier. The first grammar is the one from Table 8.5, for the second grammar rule 6 has been omitted.

	NN+LM	without prosody
# successful analyses	178	165
average syntactic readings	8.2	128.2
average parse time (secs)	4.9	38.4

Table 8.9: Parsing results for the KA–LING_BIG word graphs using the final version of the TUG system. Prosodic information clearly improves parse time and reduces the number of readings in the output of the parser.

the search space could be considerably reduced if a break were required after every input phrase including "exclamatives". The word ja, for example, is derived from the non–terminal symbol EXCLAMATIVE regardless its function. Recall that ja can be an elliptic utterance functioning as a confirmation and meaning yes, or it can be a discourse particle meaning well; cf. Section 4.2.5. Therefore, rule 6 in Table 8.5 has been omitted which results in obligatory PSCBs after every phrase including exclamatives. With such a kind of grammar only 79 of the word graphs could be analyzed. Therefore, in the following the grammar version shown in Table 8.5 is used.

Table 8.9 shows the parsing statistics for the final version of the TUG system using the "B3/B[029]–NN & M3/M0–LM$_3$" classifier. The number of syntactic readings and the parse time was in each case evaluated on the total number of successfully analyzed word graphs. Using the prosodic boundary information improves the number of successful analyses by 7.9%, and it reduces the average number of readings by 93.3% and the average parse time by 87.2%.

The number of readings can also be reduced by utilizing other knowledge sources in a postprocessing step. For example, the semantic analysis would reject a great number of readings. We investigated the use of a domain knowledge–base [Qua95][10]. This *domain model* is used to rank the different readings. The best ranked reading was then manually compared with a bank of ideal parse trees determined on the basis of the spoken word chains. The effect of the domain model

[10] We are grateful to Manfred Gehrke, Siemens München, who conducted this evaluation.

	#
different paths in the word graph	6
identical parse trees	83
non–identical trees	54
out of these	
better with prosody	33
better without prosody	14
unclear	7

Table 8.10: Domain knowledge was used to select the most plausible from alternative parse trees. Compared are the parse trees, which were obtained with the TUG parser on KA–LING_BIG with and without prosodic information [Geh96]. If prosodic information was used during parsing, a significant number of automatically computed parse trees better matches reference parse trees. The results also show that prosody improves not only the parse time or the number of parse trees but also the average quality of the parse trees.

will be explained considering the following example [Geh96]:

Zu der Zeit gehe ich zum Zahnarzt. (8.14)
(At that time go I to–the dentist.)
At that time I'll be at the dentists.

This is the correct reading which leads to an interpretation of zu der Zeit as a time expression and of zum Zahnarzt as the goal of a movement. However, syntactically possible would also be

Zu der Zeit gehe ich. Zum Zahnarzt. (8.15)
(To the time go I. To–the dentist.)
I go to the time. To the dentists.

Here, zu der Zeit is goal of the movement, after ich is a clause boundary, and zum Zahnarzt is an elliptic sentence. Of course this is no meaningful segmentation of the utterance, however, this case has to be considered by the syntactic analysis, because it would be plausible in a similar situation as the following example shows:

Zu der Bank gehe ich. Zum Wertpapierberater. (8.16)
(To the bank go I. To–the stocks–consultant.)
I go to the bank. To the stocks consultant.

Table 8.10 summarizes the results of the evaluation of parse trees with the domain model. We compared the syntactic analysis of prosodically scored KA–LING_BIG word graphs with the parse trees obtained without the use of prosodic information. The numbers given in the table were obtained on the 143 word graphs

		NN+LM	NN	without prosody
# successfully analyzed		176	165	174
# of readings	maximum	42	50	2592
	average	6.1	6.1	174.7
parse time (secs)	maximum	24.9	29	2967
	average	2.8	4.7	65.2

Table 8.11: Parsing results on DB–LING_BIG word graphs obtained with the final version of the TUG system. The clause boundary scores computed by the NN already improve the parsing significantly. Compared to this a further considerable improvement is achieved by the use of the combined NN/LM classifier. It can also be seen that prosodic clause boundary information improves also the parsing of word graphs from Daimler Benz.

which could be successfully analyzed with both versions. No comparison was possible for 6 word graphs, because in the two experiments different paths in the word graph were determined by the A* search. In 33 of the remaining 137 word graphs a better analysis could be found with prosodic information, in 14 cases the parse tree found without prosodic information was better. So in summary in 19 cases, that is, 14% of the 137 word graphs, the result of the combination of syntactic analysis and domain model is better with than without prosodic information. This again shows the importance of the prosodic information, but it also indicates that a great percentage of the numerous readings found in the word graph parsing without prosodic information can be discarded on the basis of the domain model. However note that the domain model can only be applied as a postprocessor to the parsing so that it at best can reduce the number of readings but it cannot speed–up the parsing itself. The results also show that prosody improves not only the parse time or the number of parse trees but also the average quality of the parse trees.

Next, we conducted experiments on the DB–LING_BIG word graphs. One goal was to evaluate if it makes a difference, which of the VERBMOBIL word recognizers was used for the generation of the word graphs, and furthermore we used these word graphs to evaluate the effect of the prosodic polygram classifier. Table 8.11 shows the number of successfully analyzed word graphs. The number of readings and the parse time shown in the table refer to the 154 word graphs which could be successfully analyzed within all of the three experiments. One can see that the results are comparable to those obtained on the KA–LING_BIG word graphs, cf. Table 8.9: Prosodic information reduces parse time and the number of readings considerably no matter if only the acoustic–prosodic B3/B[029]–NN or the combination "B3/B[029]–NN & M3/M0–LM_3" was used for prosodic scoring. Although in total more word graphs could be successfully analyzed without prosody compared to using the NN alone, this can only be achieved at the expense of an intoler-

		NN+LM	without prosody
# successfully analyzed		359	368
# of readings	maximum	108	9256
	average	5.6	137.7
parse time (secs)	maximum	31	5898
	average	3.1	38.6

Table 8.12: Parsing results on the 594 KA–LING_TEST word graphs using the final version of the TUG system. In contrast to the previous experiments, this data is independent in the sense that it has not been used before for the evaluation neither of the prosody module nor of the parser. Again prosodic information significantly improves the performance of the parser.

able high need in computing time. The results in Table 8.11 show also that the use of the prosodic polygram classifier considerably improves the parser performance: The number of analyzable word graphs increases by 6.7% while the average parse time decreases by 40.4%. In these experiments, for the "B3/B[029]–NN" classifier the weight ξ_3 was set such that the same number of clause boundaries was recognized as in the case of the "B3/B[029]–NN & M3/M0–LM_3" classifier.

Finally, we evaluated our system on the KA–LING_TEST word graphs again using the "B3/B[029]–NN & M3/M0–LM_3" classifier. Table 8.12 shows the results. Recall that this sub–corpus has not been used for training/development or testing of neither prosody nor parser. So in contrast to the previously described experiments this one is conducted on entirely new test data. Without the use of the PSCB probabilities 368 or 62% of the word graphs could be analyzed successfully. Thus in contrary to the results from Tables 8.9 and 8.11 2.5% more word graphs are successfully analyzed. However, again the PSCB probabilities computed by the prosody module drastically reduce the average number of readings (by 96%) and the average parse time (by 92%). Moreover, the parse time without the use of prosody is for many turns much higher than it would be tolerable in the VERBMOBIL system. Without the use of prosody the analysis of 18 word graphs takes more than a minute and the analysis of altogether 32 word graphs takes more than 30 secs. With prosody only 4 word graphs need a parse time of more than 30 secs.

Conclusion

We can conclude that we successfully employed prosodic clause boundary scores in the search of the TUG word graph parser. The use of these scores significantly speeds–up the search and reduces the number of readings, especially, on the VERBMOBIL spontaneous speech data. Both quantities improve by well over

90%. The number of successfully parsed word graphs is about the same with or without the use of prosodic information. However, in many cases the parse time is by far too large with respect to the use in an ASU system if no prosodic information is integrated in the search. Furthermore, the quality of the automatically computed parse trees is improved, that is, they better match reference parse trees when this prosodic information is used. It is worth mentioning that the real–time factor for the parser using prosodic information is about 0.5 on these word graphs using a Sparc 20 work station. The real–time factor for prosody module and parser together is about 1.5 whereas the real–time factor is 6.1 for the parser which does not use prosodic information. In the VERBMOBIL system the output of the parser is subsequently analyzed by the module for semantic construction. This module is able to analyze successfully 277 (77.2%) out of the 359 KA–LING_TEST word graphs which were parsed using the prosodic information, cf. Table 8.12. The results of this section are also summarized in [Kom97].

8.3.3 Constraining the Position of Head Traces

In this section we describe how prosodic boundaries are utilized in the VERBMOBIL syntax component developed by IBM, cf. Section 3.4. In this system, an HPSG is processed by a bottom–up chart parser that takes word chains as its input [Kis95]. Since in VERBMOBIL the interface between word recognition and syntactic analysis is the word graph, a preprocessor searches for alternative string hypotheses contained in the graph. These differ in the wording, in the place of empty elements (cf. below), and in the positions of segment boundaries. The individual segments are then parsed one after the other. Prosodic clause boundary information is used for constraining the positions of empty heads and for the segmentation of the string hypotheses. The probabilities for prosodic clause boundaries are taken from the annotations of the word graphs, which are computed with the algorithm described in Section 8.1. In the following we will summarize the basic approach; a detailed description will appear in [Bat96d, Bat96c][11].

The Approach

HPSG makes crucial use of so called *head traces* to analyze the verb–second phenomenon, that is, the fact that finite verbs appear in second position in main clauses

[11] The extensions of the HPSG and the parser were implemented by Stefan Geissler and Tibor Kiss. They also conducted the parsing experiments described in this section.

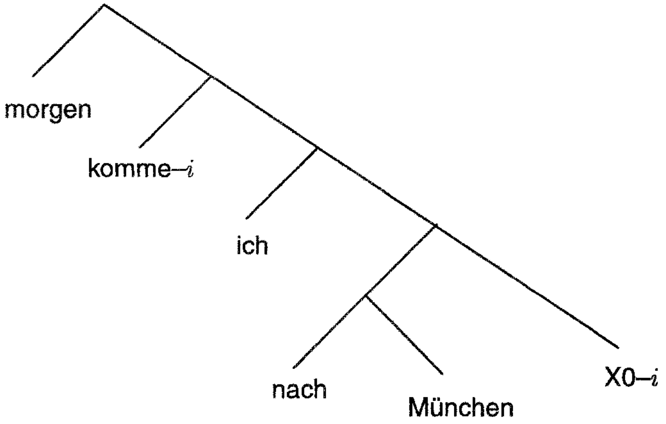


Figure 8.9: Syntax tree for Morgen komme ich nach München. The relationship between the verb and the empty element X0 is expressed by the index i.

but in final position in subordinate clauses, as exemplified in (8.17) and (8.18).

Morgen komme ich nach München. (8.17)
(Tomorrow go I to Munich)
Tomorrow, I'll go to Munich.

Ich dachte, daß ich morgen nach München komme. (8.18)
(I thought that I tomorrow to Munich come.)
I thought that I would come to Munich tomorrow.

It is assumed that the structural relationship between the verb and its arguments and modifiers is not affected by the position of the verb [Kis95]. As a consequence, the representation in the grammar is independent from the actual position of the verb. The relationship between the verb komme and its object nach München in (8.18) is preserved in (8.17), although the verb shows up in a different position. The apparent contradiction is resolved by assuming an empty element (X0) which serves as a substitute for the verb in second position. The empty element fills the position occupied by the finite verb in subordinate clauses, leading to the structure of main clauses exemplified in Figure 8.9. This approach allows to model main and subordinate clauses in the same way by the grammar.

Direct parsing of empty elements can become a tedious task, decreasing the efficiency of a system considerably, because empty elements can occur at many

	NN	without prosody
average parse–time	6.5	11.9
speed–up	46.0%	–

Table 8.13: Run–times (in secs) for parsing of word graphs with the HPSG system. The word graphs were obtained with the VERBMOBIL system.

positions in an utterance. This is especially the case in utterances containing multiple clauses and free phrases. Although the possible positions of the empty elements have been constrained by rules employed within the preprocessor, the search space still is very large. However, we found that the search space can be further constrained by the observation that empty verbal heads can only occur in the right periphery of a clause, and furthermore, at a phrase boundary. Often but not always empty verbal heads are at the end of a clause. So it seems to be obvious that prosodic information can be used to restrict the positions of empty heads.

For both, the prediction of empty head positions and the segmentation of utterances, the same prosodic clause boundary probabilities contained in the word graph and computed as described in Section 8.1 are used. The segmentation is based on whether the probability for PSCB is greater than 0.5. Empty elements are predicted if the probability for boundary exceeds a heuristically determined threshold. Here it is crucial that (almost) no correct positions are ruled out; false–alarms are less disturbing. A detailed off–line analysis is presented in [Bat96c]. It shows that the number of empty head positions can be significantly reduced with this method.

Experiments and Results

All experiments were conducted on word graphs with about 10 hypotheses per spoken word. In a first experiment 109 word graphs were used which were obtained from tests with the VERBMOBIL system with non–naive users. Table 8.13 compares the parse–time with and without the use of the prosodic clause boundary probabilities. These probabilities were computed with a preliminary version of the NN described in Section 6.4.3. No polygram information was used. Employing prosodic information reduces the run–time on these word graphs by about 46%.

In a second experiment, we were interested in the effect of employing an NN/polygram classifier. This time we used a subset of the word graphs used in Section 8.3.2, Table 8.8. They correspond to one dialog (21 turns) of the real VERBMOBIL spontaneous speech data. Table 8.14 compares the B3/B[029]–NN

	B3/B[029]–NN & M3/M0–LM$_3$	B3/B[029]–NN
# of units in turns	60	117
# of successfully parsed units	31	52
# average runtime per unit	17.1	5.1
# overall runtime	1025.8	595.9

Table 8.14: HPSG parsing results for a real spontaneous speech dialog with and without polygram boundary classification (M3/M0–LM$_3$).

described in Section 6.4.3 with the combined B3/B[029]–NN & M3/M0–LM$_3$ model described in Section 7.2. As can be seen in the table, not using the polygram information leads the processing module to hypothesize a larger number of segment boundaries; these are less meaningful in the context of a further processing in the full system than those that can be identified in the alternative setting. The seemingly unfavorable result that the parser performed much slower when using the polygram classification result has to be interpreted with respect to the fact that in the non–polygram setting the input was segmented into about twice as many units. On the average the units are twice as large as in the with–polygram setting so that it can be seen as a positive result that the overall run–time increased only by a factor of two. Furthermore, for a subsequent analysis of the units in the full VERBMOBIL system the larger units are much more useful. This was verified by a manual evaluation of the parsing results by the grammar writer at IBM. This is also indicated by the fact that in the no–polygram setting the average unit length is less than 3 words. With the polygram even a larger total portion of the input (52% of the units) could be parsed than without the polygram (44%). Note that without prosodic information it was not possible to parse this corpus at all due to memory limitations.

8.4 Semantics

In Sections 4.2.4 and 4.2.5 we already outlined the importance of accentuation with respect to focus disambiguation especially in the presence of discourse particles. With respect to the VERBMOBIL system the group working on the semantic module considered the interpretation of these particles as important. It was furthermore agreed that prosody can be of great use for this task [Bos95]. Let us consider the following examples, which are an extension of example (3.7):

der Montag geht auch bei mir (8.19)
der Montag geht auch bei mir
(the Monday suits also for me)
Monday as well is fine for me

der Montag geht auch bei mir (8.20)
Monday is fine for me as well

In these examples the word sequence is the same, however, it can be seen from the English translation that the meaning is different depending on the accentuation and thus on the focus position. In (8.19) the focus position is on Montag regardless if Montag itself or auch is accented. In fact in this situation normally a speaker will put the accent on auch. In example (8.20) the focus is on the word mir. With respect to this we analyzed 21 VERBMOBIL dialogs (589 turns); this is part of the BS_TRAIN sub–corpus and covers all VERBMOBIL turns for which at that time acoustic–prosodic accent labels were available. In these the particle auch occurs 53 times; 28 times it is accented and in 26 of these cases the focus is to the left of auch; 25 times auch is unaccented and the focus is actually to the right of auch in 21 of these cases. This analysis shows that in (8.19) the accentuation of Montag would be rather unlikely and consequently that the accentuation of auch really governs the focus and thereby the interpretation of the utterance. A similar behavior can be expected in the case of other particles as noch, nur, schon, immer, genau which, however, are less frequent in the corpus.

Based on these observations we made accent information available for the semantic module of the VERBMOBIL system. The information consists of a probability for a word hypothesis to be accented. It is computed with the algorithm for the prosodic scoring of word graphs described in Section 8.1 using the accent classifier of Section 6.4.3. The semantic module receives a parse tree, the underlying word chain and the scores for accentuation from the syntax module. Based on these, underspecified DRSs are created, cf. Section 3.4. These yield assertions, representing the direct meaning of a sentence, and presuppositions. For example (8.19) the assertion

$$[\, e\ i\ t \mid montag(t)\ gehen(e)\ theme(e,t)\ bei(e,i)\ t \neq t' \,] < i, speaker >$$

is constructed. The assertion for (8.20) is

$$[\, e\ i\ t \mid montag(t)\ gehen(e)\ theme(e,t)\ bei(e,i)\ i \neq i'\ t = t' \,]$$
$$< i, speaker >$$

Both examples have the same DRS as presupposition, however, the binding of the variables are different as indicated by the above constraints like $t = t'$:

# of clause–like segments	57
# correct accent patterns	49
# wrong accent patterns	0
# segments where accent disambiguated correctly	4
# segments where accent disambiguated in a wrong way	0

Table 8.15: Evaluation of the use of the accent information with respect to the semantic analysis.

$[\ e'\ i'\ t' \mid gehen(e')\ theme(e,t')\ bei(e',i')\ ]\ < i', speaker >$

In the first case, $t \neq t'$ expresses that the speaker talks about a date which was not previously mentioned; someone might, for example, have suggested *Tuesday* as a meeting date. It is left unspecified if *Monday* has been suggested by the same speaker or by another person. In the second example speaker i' previously suggested to meet on a *Monday* and a different speaker $i \neq i'$ confirms that for him the same *Monday* ($t = t'$) is fine as well.

Given just the word chain, both DRS are plausible and have to be hypothesized. If accent information is available the interpretation can be disambiguated by ruling out one of the DRS. If available, context information might also be used to disambiguate the interpretation, because the relationship $t = t'$ holds only in one of the DRS. However, prosodic information can presumably be utilized at much lower cost [Bos96a].

The semantic formalism has been extended in order to utilize the accent information computed by the prosody module. Preliminary evaluation was conducted on a set of 57 clause–like turn segments which were the result of a successful parse by the syntax module described in Section 8.3.2[12]. The analyzed segments were obtained in the course of tests with the VERBMOBIL system. The evaluation of the accent information is given in Table 8.15. The row "correct accent patterns" does not reflect if for each of the words it was correctly recognized, but this refers to the number of cases where the recognized accent pattern meets the requirements of the semantic analysis. For example in the sentence

Lassen sie uns doch noch einen Termin ausmachen (8.21)
(Let you us nevertheless still a date to fix)
Nevertheless, we still need to fix a meeting date

[12] The extension of the semantic formalism was carried out by Johan Bos, Universität Saarbrücken. He also kindly conducted the experiments with the semantic module, the results of which are presented here.

taken from the above corpus, a correct semantic interpretation requires that the word Termin is accented and the word noch is not accented; cf. examples (4.19, 4.20). The result of the accent classification of the other words is not relevant. It turns out that in 7% of the segments accent information is necessary for the disambiguation of the interpretation. In none of these cases the automatically computed accent information led to the wrong interpretation.

A further task of the semantic module is to determine the sentence mood. This information is not used for the semantic interpretation but it is passed on to the transfer module, which needs it for proper translation. The prosodic accent information is as well passed on to the transfer module.

8.5 Transfer

The most recent version of the transfer module of the VERBMOBIL prototype system uses accent and sentence mood information computed by the prosody module. In cases where the sentence mood cannot be determined from the word order or from particles indicating questions, it can only be determined by intonation. Consider the following examples where the verb, treffen (*meet*), is in initial position [Bos96a]:

Treffen wir uns dann beim Informationsbüro der IAA! (8.22)
So, let us meet at the IAA information office.

Treffen wir uns dann beim Informationsbüro der IAA? (8.23)
Do we meet then at the IAA information office?

Example (8.22) is a part of a turn taken from a VERBMOBIL corpus. The speaker marked the sentence by a falling intonation contour. Although the verb–first word order in German usually indicates a question in this case the intended sentence mood is an imperative. In the second example (8.23) it is assumed that the speaker marks the same word chain as a question by a rising pitch. It is as plausible as the first example. It can be seen that the translation of this word chain differs only depending on the pitch.

Currently in the VERBMOBIL system, the sentence mood classifier as described in Section 6.3 and applied to word graphs as explained in Section 8.1 is used within the prosody module. The class probabilities contained in the word graph are extracted by the syntax module and passed on to the semantic construction which then provides the transfer module with a binary decision, which is question versus no–question depending on whether the probability for rising pitch computed the

NN is greater than a heuristically determined threshold. Since the use of the sentence mood within the VERBMOBIL prototype became relevant only very recently, we were not able anymore to optimize the classifier for the VERBMOBIL domain.

In Section 4.2.5 it was stated that prosodic accent information is important for the proper translation of particles. This is closely related to the disambiguation of the semantic interpretation of particles presented in Section 8.4. In the context of the transfer module, the disambiguation takes place on the pragmatic level rather than on the semantic level. Since recently, for a few particles the prosodic accent information is used in the transfer module integrated in the VERBMOBIL prototype system [Rip96b]. In this context an important particle is nur:

ich habe <u>nur</u> eine Frage (8.24)
I only have a question

ich habe nur eine <u>Frage</u> (8.25)
I just have a question

The particle nur is translated as *only* if it is accented, cf. example (8.24), and it is translated as *just* if it is unaccented.

Results for the transfer module evaluating the use of intonational sentence mood or prosodic accent information are not available yet.

8.6 Dialog

8.6.1 Classification of Dialog Acts in VERBMOBIL

As already mentioned in Section 3.4 the VERBMOBIL system has to keep track of the dialog history, which means at least to recognize all dialog acts. In our approach this is done in two steps: first, a turn is segmented into dialog act units (DAU), and, second, these segments are classified into dialog act categories (DAC). The contribution of our research to this task is the segmentation of turns. For this the same methods are applied as for the segmentation of turns into clauses, cf. Section 7.2. In the following we will describe the approach; it will also appear in [Mas96]. The methods used for dialog act classification itself are described in detail in [Mas95c], where semantic classification trees and polygrams are compared; here we only will consider polygrams. The dialog act classification experiments described below are described in detail in [War96]. All experiments described below were performed using the VERBMOBIL sub–corpora D_TRAIN and D_TEST. Up to now we only worked on the spoken word chain.

θ	*accuracy*	*correct*	*deletions*	*insertions*
0.95	45.2	48.8	25.3	3.6
0.93	45.8	51.6	20.8	5.8
0.86	44.4	55.7	12.9	11.4
0.79	43.2	56.5	11.0	13.3
0.50	29.3	61.7	4.8	32.8

Table 8.16: Classification results for DACs based on automatically segmented VERBMOBIL turns.

Segmentation

For the segmentation of turns into DAUs NNs and polygrams were used. With NNs an average recognition rate of 83.6% for D3 vs. D0 was achieved. The polygram alone yielded a recognition rate of 90.7%, the best result (92.5%) could be achieved with a combination of NN and polygram. We also tried an NN trained on the perceptual–prosodic clause boundaries (B3/B[029]) for the classification of Ds; the recognition rate of the NN alone was slightly better (84.4%), however the combination with the LM yielded only a recognition rate of 91.3%. All these results do not take into account the end of turns which by default are labeled as D3; their classification is trivial.

Classification of Dialog Acts

For the classification of the 19 different DACs based on the hand–segmented data a recognition rate of 59.7% was achieved. For the automatically segmented turns the DAC classification results are shown in Table 8.16. The classification is conducted as follows: first, we compute for each word boundary the probabilities $P(\text{D3})$ and $P(\text{D0})$. Second we classify each boundary as D3 if $P(\text{D3}) > \theta$ and as D0 else. Third the word chain between each subsequent pair of D3 is extracted and classified with the polygram into one out of the 19 DACs.

For the evaluation it has to be taken into account that DAUs may be deleted or inserted. Therefore, we align for each turn the recognized sequence of DAC symbols with the reference. The alignment is performed with respect to the minimization of the Levenshtein distance. The percentage of *correct* classified DACs is given together with the percentage of *deleted* and *inserted* segments in Table 8.16. Furthermore, the recognition *accuracy* measures the combined classification and segmentation performance. Note that in this evaluation a DA is considered as classified correctly if it is mapped onto the same DA category in the reference no matter if the segment boundaries agree with the hand–segmented boundaries. In this context the most important numbers are the correctly classified DAs versus

	RR	CRR
"B3/B[029]–NN & D3/D0–LM$_3$"	92.8	81.5
"B3/B[029]–NN & M3/M0–LM$_3$"	92.4	84.5

Table 8.17: Recognition rates for D3/D0 using different polygram classifiers.

the insertions. In the table results for different thresholds θ are given. The smaller θ the smaller the number of deleted and the larger the number of inserted segments. The best accuracy achieved is 45.8%. The highest percentage of correctly classified DAs is 61.7%. This is even higher than the recognition rate of 59.7% The reason for this might be that on the one hand most of the deleted segments are misclassified on hand–segmented data and on the other hand the high number of insertions increases the likelihood for a correct class being among the recognized ones.

Integration in the VERBMOBIL system

The results above show that DAs can be reliably classified based on automatically detected segments. The approach has actually been integrated in the full VERBMOBIL system [Bos96e]. However, to save computational effort we did not want to run two boundary classifiers in parallel in the prosody module. Instead we use the "B3/B[029]–NN & M3/M0–LM$_3$" classifier as well for the D3/D0 classification. The suboptimal D3/D0 modeling by the M3/M0–LM$_3$ is compensated by the higher amount of available training data for the M labels. Table 8.17 compares the "B3/B[029]–NN & M3/M0–LM$_3$" classifier with the "B3/B[029]–NN & D3/D0–LM$_3$" classifier on the spoken word chains of the BS_TEST corpus. The D3/D0–LM$_3$ was this time trained on D_TRAIN and D_TEST so as to make use of the maximally available amount of data. It can be seen that the average recognition rate drops only from 92.8% to 92.4%. The same weight $\xi_3 = 0.3$ was chosen as in Section 8.3.2.

Results with the entire system especially on recognized word chains extracted from word graphs are not available yet.

8.6.2 Prosodic Dialog Control in EVAR

After we have described the use of prosodic information in the VERBMOBIL system we will turn to the speech recognition and dialog system EVAR. Recall that the domain of EVAR is train time table inquiry. We observed that in real human–human dialogs when the railway employee transmits the information, the customer very often interrupts. Many of these interruptions are just repetitions of the time of day given by the railway employee. The functional role of these interruptions is often determined by prosodic cues only. An important result of experiments where naive persons used the EVAR system is that it is hard to follow the train connection given via speech synthesis. In this case it is even more important than in human–human dialogs that the user has the opportunity to interact during the answer phase. Therefore, following the ideas of [Nöt91a], we extended the dialog module to allow the user to repeat the time of day and we added our prosody module to guide the continuation of the dialog by analyzing the intonation contour of this utterance. The prosody module in this task determines the sentence mood with the classifier described in Section 6.3. Our approach was already published in [Kom93b, Kom94b]; in the following we will give a summary.

Dialog Guiding Prosodic Signals

Since the goal of EVAR is to conduct dialogs over the telephone, the system answer is generated by speech synthesis. The system should allow for user interruptions and react adequately to them. In order to derive a formal scheme for this, we investigated a corpus of 107 "real–life" train time table inquiry dialogs recorded at different places, most of them conducted over the phone. In the following we will consider only the 92 dialogs concerned with train schedules; the rest had other topics such as fares. The most important question in the context of this section is how often and in which way during the answer phase the prosody of a user interruption alone controls the subsequent action of the railway employee. We presented a detailed analysis of this data also in [Bat92].

The customer utterances of these 92 dialogs contain 227 repetitions of the the time of arrival or departure given by the railway employee, that is, more than two repetitions per dialog on the average. In all but 3 cases, the repetition concerned the time–of–day the railway employee had just given before.

When repeating the time–of–day, the customer can have various intentions, that is, he wants to give the officer different kinds of information. We observed three different functional roles or *dialog acts* of the repetition of the time–of–day: *confirmation, question* and *feedback*, for examples cf. Table 8.18. A confirmation in this context is equivalent to a statement and marked by a falling pitch, cf. Section 4.2.2. A question, as pointed out in that section, is marked by a rising contour. In this particular context of isolated time–of–day expressions a continuation–rise functions as a feedback, with which the customer signals the officer something like *"I'm still listening", "I got the information"* and sometimes *"slow down, please!"* or *"just let me take down the information"*. This has been further discussed in [Kie93].

In our material, in 100 of the 227 repetitions of the customer the reaction of

confirmation:			
employee:	*You’ll arrive in Munich at 5 32.*		
customer:		*5 32.*	
question:			
employee:	*...you’ll leave Hamburg at 10 15...*		*...yes, 10 15, and you’ll reach...*
customer:		*10 15 ?*	
feedback:			
employee:	*...the next train leaves at 6 35...*		*...and arrives in Berlin at 8 15.*
customer:		*6 35 –*	

Table 8.18: Examples for customer interruptions of train time table information given by a railway employee and corresponding reactions.

the officer was governed by nothing but the intonation of the customer, because they did not contain other indicators like excuse me. The possible reactions are *confirmation* of the correctness, *correction* or *completion* of the time–of–day.

Just as in these human–human dialogs elliptic repetitions of parts of information can often be observed in simulations of human–machine dialogs as well (cf. [Kra90], [Hit89]). Therefore we intended to take into account in the dialog model of our system that the continuation of the dialog can be controlled by intonation. To simplify the problem for the beginning we restricted our model to user utterances where only the intonation and no grammatical indicators govern the system reaction. Further, only isolated time–of–day repetitions are considered which are the majority of the 100 cases mentioned above.

The Extended Dialog Module

We implemented the three different dialog acts confirmation, correction and completion, and we included them in the EVAR dialog model. The control module was extended so that the correct system reaction was performed depending on the intonation of the user utterance and taking into account if the time–of–day repetition by the user was complete or incomplete. Allowing the user to interrupt the train table information given by EVAR was considered to be a great technical problem. Furthermore, in tests with "naive" users of EVAR we found that they do not try to interrupt the system answer in the way observed in the human–human dialogs and Wizard of Oz experiments described above. This is expected to change if the system performance is improved considerably. Therefore, we restricted the implementation in a way that the user was only given the opportunity to repeat the last time–of–day uttered by the system. For this reason no results with the overall system can be given, however, it has been successfully presented to the public at several occasions, including the 1994 CRIM/FORWISS Workshop held in Munich.

8.7 Summary

For the integration of prosody in ASU, we developed a method for the prosodic scoring of word graphs; these word graphs are passed on to the linguistic modules for further processing. The main aspect is that prosodic information is already available during linguistic analysis so that especially during parsing the search space can be considerably reduced. It is important that no hard decisions are made by the prosody module but that it computes probabilities for the different classes. Furthermore, these probabilities should not only for accent classification but also in the case of boundary detection be computed for each of the word hypotheses in the graph rather than for the nodes because the feature computation and thereby the classification results depend on the time–alignment of the standard pronunciation of particular words. So we apply the NN/polygram classifier as described in Chapter 7 to each of the word hypotheses in a word graph. Both, acoustic–prosodic input features for the NN and the polygram classification, are (currently) based on ± 2 context words. Instead of considering all possible contexts and averaging the classifier scores over these contexts we determine the optimal predecessors and successors of each word hypothesis and perform the prosodic scoring based on these. Experiments on VERBMOBIL data showed that despite this suboptimal solution the recognition rate for prosodic–syntactic boundary versus no–boundary only drops from 94% to 92%. Furthermore, it can be shown that the NN/trigram classifier performs better than a NN/bigram classifier which in turn shows better results than the NN alone.

We used the same approach for the verification of word graphs based on the assumption that on wrong word hypotheses the decisions of NN and polygram would mismatch. This as well as the direct verification of word hypotheses on the basis of acoustic–prosodic features showed no positive result.

Our main research effort was spent on the improvement of parsing by prosodic boundary information. So far we restricted our studies to clause boundaries. Two approaches were explored. The first one uses a trace unification grammar and a Tomita parser from Siemens which operates on word graphs. This is achieved by applying the parser to partial word chains within an A* search. The so far analyzed partial word chains are on the OPEN list. At each step of the search the best partial word chain is removed from the OPEN list and parsed. If it is not analyzable by the parser it is not considered any further, otherwise it is extended according to the subsequent edges in the word graph. Additionally it is also extended by a special clause boundary symbol. All the extensions are scored and inserted in the OPEN list. The score includes trigram scores for the word chain, the acoustic word scores and the prosodic score for either a boundary or no–boundary succeeding the word. Furthermore, appropriate remaining costs for all these components of the scoring function are determined. With this approach we were able to speed–up the parsing of word graphs computed on VERBMOBIL spontaneous speech testing data by 92% to an average of 3.1 secs, that is a real–time factor of 0.5, with respect to the parsing experiments, where no prosodic information was used; the number of parse trees (readings) reduces by 96% to an average of 5.6. The word graphs contained about 9 word hypotheses per spoken word.

The other parser was developed by IBM on the basis of an HPSG formalism. This approach has the drawback that it does not directly parse word graphs. A preprocessor extracts the 100 best sentence hypotheses from the word graph. These are parsed one after the other until a parse has been found, if at all. In the preprocessor the prosodic information contained in the word graph is used for the segmentation of the word chain into clause–like units and it is used to restrict the possible positions of verb traces. So a pair of sentence hypotheses differs either in the wording and/or in the position of the verb trace and/or in the position of clause boundaries. Both, the prediction of verb traces and of clause boundaries, restrict the search space of the parser considerably. If only the NN is used for these prediction tasks a speed up of already 46% is achieved. The use of an NN/polygram classifier improves the segmentation of the turns and increases the amount of analyzable input.

A corpus based analysis of particles showed that the position of the focus in an utterance often depends on whether the particle is accented or not. In these cases usually the focus position is ambiguous without accent information; either meaning or the presupposition of a sentence changes depending on the focus position. Therefore, the semantic analysis uses the accent probability as computed by the prosody module for each of the word hypotheses in the word graph. It reduces the number of ambiguous interpretations of a sentence. Accentuation is a relational prosodic attribute: a word is accented if it is more prominent than others in a phrase. We found that this is also reflected in the probabilities computed by our NN for accent classification.

In VERBMOBIL the dialog module has to keep track of the dialog acts. Therefore, these have to be detected within the turns. This is performed within two steps: the turn is first segmented into dialog act units which then are classified into dialog acts. Our contribution to this task is in the segmentation of dialog acts. We showed that the recognition rate for boundaries is 93% including false–alarms using the NN/polygram classifier. The dialog act recognition accuracy drops from 60% to 46% using the automatically recognized segment boundaries rather than the reference boundaries.

In the current implementation of the VERBMOBIL prototype the prosody module fulfills the task as described above. This prototype has been successfully demonstrated to the public at various occasions like the Hannover CeBit fair and the Internationale Funkausstellung in Berlin.

In the context of our train time table inquiry system EVAR we integrated prosodic sentence mood information for dialog act classification itself. We observed that in human–human dialogs conducted over the telephone the customer frequently repeats the information the railway employee gives. These repetitions are very often elliptic time–of–day expressions. The sentence mood of these repetitions determines the dialog act and thereby the continuation of the dialog. Depending on whether the user intention is a confirmation, a question or just a feedback, the railway employee either does not react especially, or he repeats his last utterance, or slows down. This scheme has been successfully integrated in the EVAR prototype system and was successfully demonstrated at several occasions including a press conference at the 1994 CRIM/FORWISS Workshop in München.

Chapter 9

Future Work

Although we already showed that the performance of linguistic modules of ASU systems can considerably be increased by prosodic information, we believe that this research as well as the the work of [Ost93b, Hun95a] is just a starting point for the use of prosody. Despite a further improvement of our algorithms, we believe that a lot of basic research in this field of speech understanding still has to be done. In this chapter we will sketch a few ideas. We will start by giving examples for aspects of prosody which we did not consider in this thesis, but for which the use in ASU is promising. We will conclude by arguing for a more integrated processing strategy, which is in contrast to the bottom–up approach realized in the present version of VERBMOBIL.

Boundaries between Clause and Subordinate Clause

We could show that the prosodic scoring of word graphs based on classifiers for the detection of prosodic–syntactic clause boundaries improves the parsing of word graphs a lot. So far we treated boundaries between two clauses in the same way as boundaries between clause and subordinate clause. In a preliminary informal evaluation of this approach in conjuction with the semantic construction it turned out that for the semantic interpretation it would be helpful if clause and subordinate clause were distinguished already within the syntactic analysis [Sch96a]. Although subordinate clauses are often initiated by subordinations the discrimination between these two types of boundaries is a highly ambiguous task at least if local decisions have to be made as in the A* based word graph parsing. Consider the following examples:

Mir wäre es dann etwas lieber. Wenn wir die ganze Sache auf Mai verschieben könnten, wäre mir aber mehr gedient. (9.1)
It would be a bit better. If we could move the whole thing to May, however that would be really great.

Mir wäre es dann etwas lieber, wenn wir die ganze Sache auf Mai verschieben könnten. (9.2)
I would prefer, if we could move the whole thing to May.

In both examples a clause boundary is after the word lieber and the subsequent words are the same in both cases. However, in (9.1) a new sentence starts after lieber, whereas in (9.2) the first clause is continued with a subordinate clause. This might be disambiguated by prosodic information. Several prosodic attributes might be important in this context. Example (9.2) corresponds to the utterance shown in Figure 4.4. In the example shown in this figure, the speaker marked the boundary between clause and subordinate clause by a continuation rise. However, we believe that this is not mandatory and not the only possibility. Apart from the intonation contour the length of pauses might also play an important role in this context. It can be expected that the boundary after lieber in example (9.1) is, in contrast to (9.2), likely to be marked by a distinct pause. Furthermore, other prosodic attributes like the energy contour could also be relevant for this task.

If these boundary types are to be distinguished by the prosody module, one needs labeled training data. In the context of the M labeling scheme presented in this thesis, this means that the M3S labels have to be subdivided (cf. Section 5.2.5). Furthermore in contrast to the prosodic scoring of word graphs described in Section 8.1, it would be important to integrate sentence mood and boundary classification.

Constituent Boundaries

So far the classification of prosodic constituent (B2) boundaries on VERBMOBIL data is not very robust, cf. Section 6.4.3. We believe that this is mainly caused by an insufficient amount of training data, because using the ERBA corpus prosodic–syntactic constituent boundaries can reliably be detected based on prosodic features. Furthermore, we indicated in Section 4.2.3 that the recognition of constituent boundaries can be important for the disambiguation of the PP–attachment. Therefore, it is worthwhile to try to develop also for spontaneous speech robust classifiers for prosodic–syntactic constituent boundaries. For this reliable reference labels are needed. In Section 5.2.5 the M2I were briefly introduced. These have to be manually assigned to word boundaries of transliterated speech. Then those M2I, which are close to a M3 boundary, should be automatically turned into M1I using rules similar to those we developed for the ERBA corpus. This approach should result in a large and fairly reliable amount of training data so that prosodic–syntactic constituent boundaries in spontaneous speech can be modeled well enough for the use in automatic syntactic analysis.

Disfluencies

So far we did not consider any disfluencies in our models. However, as has been shown in [Kie97] these can be detected in utterances by prosodic means. Furthermore, information about disfluencies, especially repetitions, repairs and restarts, is important for the linguistic analysis. The classification results presented in [Kie97] might be further improved by polygram classification as developed in this book for prosodic boundary detection. On the other hand, the boundary recognition itself might even be improved by taking disfluencies into account in the models.

Accents

The use of accent information in the VERBMOBIL project is just at the beginning. So far it is used for the disambiguation in the interpretation and translation of particles, cf. Sections 8.4 and 8.5. We believe that there are a lot of further possibilities for the use of accent information. We will compare just two examples:

dann nehmen wir doch den Dienstag (9.3)
(then take we after–all the Tuesday)
then let's take Tuesday after all

dann nehmen wir doch den Dienstag (9.4)
(so take we how–about–if the Tuesday)
so how about if we take Tuesday

The German sentences in these examples differ only in the position of the accent. The accent position alone changes the meaning of the sentences, which results in a different translation not only of the particle doch but of the entire sentence. In addition to the contribution to the meaning of these sentences, the accent position also controls the type of the dialog act: example (9.3) is an acceptance of a previously mentioned date, whereas in example (9.4) a new date is suggested.

In both examples a speaker might put an accent on doch as well as on Dienstag. Then the position of the primary accent is relevant for meaning and dialog act. This requires that the prosody module is able to distinguish between primary and secondary accents. So far we integrated a classifier which discriminates between accented and unaccented words. In Sections 4.1.2 and 5.2.2 we outlined that different levels of accentuation have to be identified by relational rather than by absolute measurements. Therefore, it seems not to be feasible to train classifiers which explicitely distinguish between primary and secondary accents. Instead one needs statistical models that compute a score which correlates with the strength of the accent.

With respect to this we conducted a preliminary evaluation of the A/UA–NN described in Section 6.4.3. Recall that A subsumes emphatic (EK), primary (PA), and secondary (NA) accents. The NN has one output node corresponding to A and one for UA. It was trained with the usual unity vectors as desired output, that is, the strength of the accents was not reflected in the desired outputs. We investigated 21 VERBMOBIL dialogs, most of them being part of the training corpus. Within each prosodic clause bounded by B3 we compared the accent scores computed by the NN of all the words labeled as accented. It turned out that 80% of the EKs got higher scores than any PA or NA in the clause, and 72% of the PAs got higher scores than any NA in the clause. This indicates that even such an NN trained only to recognize any kind of accent computes a score, which to some extent reflects the (relative) strength of the accent. It might therefore be useful for a primary/secondary accent distinction.

The recognition rates of the VERBMOBIL accent classifier integrated in our prosody module might be improved in general. The increase of the amount of training data would be the most important step towards this goal. This means that more data has to be labeled so that it can be used for the training of classifiers. Recall that the accent labels for the ERBA corpus have been placed automatically given the transliteration of the utterances and the boundary labels. We are convinced that it is possible to develop a similar set of relatively simple rules for spontaneous speech as for the VERBMOBIL corpus and to label accents on the basis of the transliterations and boundary labels, which in this case would be the M3 labels. This would make a large amount of labeled training data available, which should improve the classification results.

Integrated Models

Many other possibilities for the use of prosodic information have been indicated in Section 4.2. However in the long perspective, research should focus on more integrated models as has been the case in this thesis; cf. the discussion in [DM96a]. For example, in our approach for the use of prosodic information in the parsing of word graphs, the scores of three different knowledge sources are combined:

the acoustic, the prosodic, and the n-gram scores. Since the scores are assumed to be probabilities of independent events one should be able to simply multiply them. However, these scores are computed by components which have been trained independently and in a sub–optimal way. Since the independence assumption does only partly hold, we used heuristic weights to normalize the scores when they are combined.

With respect to this, further research should focus on the development of a unified acoustic model for speech recognition, which integrates phonetic and prosodic information, because both prosodic and phonetic attributes of speech influence each other, cf. for example [Cam95b, Sen96]. A first step towards this goal in the case of continuous speech recognition was presented in [Dum94a], where the HMMs included a statistical model for acoustic–phonetic and another one for acoustic–prosodic feature vectors. Although in this case the HMMs use both information sources, this is not really an integrated model. That would probably require the asynchronous extraction of features which capture longer time intervals than the usual 10 msec frames. We consider the variable frame rate analysis presented in [Fla92], auditory modeling [Sen88], and the stochastic segment models [Ost89, Bac96] as well as the segmental HMMs [Hol96] as appropriate approaches towards a more integrated prosodic–phonetic processing.

For the classification of prosodic attributes as well as for word recognition heuristic features are used at the current state–of–the–art. We believe that the performance could be considerably improved by a joint optimization of feature extraction and recognizer. First approaches towards this goal have already been presented: on the one hand there are NN/HMM hybrid approaches as the one presented in Section 2.7.1 or the mixture of experts approach introduced in [Jor94] and successfully used for speech recognition in [Zha95]. We ourselves used an NN for the detection of laryngealizations where the speech signal is directly used as input to the NN. In this way the NN can be viewed as implicitly performing and thus learning the feature extraction. We published preliminary results in [Nie94a] where a recognition rate of 76% at a false–alarm rate of 16% could be achieved using the same speech data (1,329 utterances) for training and testing.

Furthermore, a sequential prosodic and syntactic processing of an utterance is not satisfying because both are knowledge sources which depend mutually on each other. In the case of word graphs, the prosodic scores attached to a word hypothesis are especially based on few context words. In the subsequent syntactic analysis at each time a particular partial path in the word graph is considered. This path, however, might not contain the context words, which were the basis for the computation of the prosodic scores. Therefore, with respect to this particular path the score is based on a sub–optimal word sequence. This drawback could be overcome by an integrated prosodic–syntactic analysis. For efficiency reasons, this might be not realizable with the kind of parser as used in Section 8.3. We believe that simple stochastic models which represent the rough or flat structure of phrases should be developed for such an integrated prosodic–syntactic analysis. These might in the long run even replace the n-gram models used in word recognition.

In this thesis, we showed for a few tasks that the incorporation of prosodic information in ASU can improve the linguistic analysis dramatically. In view of all the possibilities for future work mentioned above, we hope that this book might initiate further research, for example, along the lines discussed in this section.

Chapter 10

Summary

Automatic speech understanding (ASU) is becoming increasingly important. In many fields such systems can be of great help, for example, dictation of letters or reports, information retrieval over the telephone, or speech–to–speech translation. Since the problem of speech understanding is very complex, current systems are decomposed into modules for feature extraction, word recognition, syntactic analysis, semantic interpretation, dialog control, and possibly transfer as well as speech generation and synthesis. In most of the current systems prosodic information, which means acoustic information about accentuation, sentence mood, and phrase structure, is not used. Various groups tried to integrate prosodic information into word recognition with moderate success. Recently, two groups had success with the use of prosodic information in the disambiguation of syntactic analysis. However, these experiments were performed on speech read by trained speakers and did not lead to the integration of a prosody module into fully operational prototype systems. The contribution of the research presented in this book is the use of prosodic information on several levels of linguistic analysis. Experimental evaluation was mainly performed on spontaneous speech. Eventually the research lead to a prosody module integrated into two prototype systems for speech understanding. To our knowledge, these systems are the first anywhere in the world successfully employing prosodic information in speech understanding and dialog control.

In order to use prosodic information in these systems we drew on well known statistical pattern analysis approaches and adapted them to our task. For classification of feature vectors we used neural networks (NNs). NNs can theoretically estimate any functional mapping and compute class a posteriori probabilities under certain conditions.

Hidden Markov models (HMMs), a special kind of finite automaton, are widely used for the modeling of sequences of feature vectors. These rely on a doubly embedded stochastic process: the state sequence generates a sequence of feature vectors, so called observations. The state sequences are controlled by state transition probabilities. Observations are assumed to be generated according to state specific observation densities. The speech recognition task is solved by a search for the optimal state sequence given a sequence of feature vectors.

General search techniques have been considered in this book. The Viterbi algorithm is a special dynamic programming technique. The cost function only takes the costs along the so far expanded paths into account. Viterbi explores the full search space and thus is guaranteed to find the optimal path. In contrast, the A* search explores only part of the search space. This is achieved by not only considering the costs for the expansion of paths but also by taking into account an estimate of the remaining costs for the expansion of a path to the goal. If the remaining costs are monotonously increasing and if they underestimate the real costs then the first path which reaches any goal of the search is the optimal path.

A priori probabilities of symbol sequences are often modeled by n-grams, which are sub–sequences of symbols of length n. The n-gram probabilities are estimated on a corpus by their relative frequency of occurrence. The probabilities of n-grams not seen during training have to be interpolated.

For the modeling of feature sequences NN/HMM hybrid systems and multi–level semantic classification trees (MSCTs) have been considered. NN/HMM hybrids combine the property of NNs to classify arbitrarily distributed feature vectors with the sequence modeling capability of HMMs. The NN is either used for feature transformation, which can be jointly optimized with the HMM parameters, or it computes the HMM observation probabilities. MSCTs have been proven useful for training of integrated speech and language models.

Our main goal was the integration of prosodic information into the two ASU systems, EVAR and VERBMOBIL. EVAR is a dialog system for train time table inquiries developed at the institute for pattern recognition of the Universität Erlangen. The linguistic processing is based on the semantic network system ERNEST. All the linguistic modules for syntactic, semantic, pragmatic and dialog processing are implemented within the same framework thereby allowing for a high interaction between the modules at early processing stages. This implies that the search alternates between top–down and bottom–up analysis. The control uses the A* algorithm and is based on problem–independent inference rules and a problem–dependent judgment vector, which ranks competing hypotheses.

VERBMOBIL is a system for speech–to–speech translation in the domain of appointment scheduling. It has been developed within a German research project. The linguistic processing is strictly bottom–up. The syntax searches for the optimal word chain in the word graph which meets grammatical constraints. The grammatical structure of an utterance is basis for the semantic interpretation. The meaning is encoded in discourse representation structures, which basically are a hierarchy of predicates. The transfer transforms these structures into a form corresponding to the English goal language. Based on this and a formal grammar English utterances are generated.

The term prosody comprises speech attributes which are not bound to phone segments. Basic prosodic attributes are loudness, pitch, voice quality, duration and pause. Variations of these over time constitute the compound prosodic attributes, which are intonation, accentuation, prosodic phrases, rhythm, and hesitation.

Intonation is the distinctive usage of pitch; general patterns like falling, rising, high or low pitch contribute to the meaning of the utterance. Intonation can be used to directly determine the sentence mood, or it can be one among other attributes that mark accentuation or phrase boundaries. The intonational sentence mood is important in the absence of syntactic markers as is the case for elliptic clauses.

Accentuation refers to syllables in an utterance, which are more prominent than others. This can be expressed by a change in duration, intonation or by an increase in loudness. The default accent position within a phrase is the last content word. This supports the structuring in prosodic phrases. Accentuation differing from the default in position or strength has various functions. The most important one is the marking of the focus, which is the new information in an utterance.

Utterances are segmented into phrases by the prosodic marking of boundaries. Prosodic boundaries usually are marked by intonation, phrase final lengthening, or pauses. There is a strong but no exact correspondence between prosodic and syntactic boundaries. These boundaries in the first place support the intelligibility of utterances, because they structure the utterance into meaningful segments. Furthermore, they can disambiguate between different readings of a sentence.

The form of the prosodic marking of certain events varies depending on speaker specific style and context. Therefore, we use statistical models, the parameters of which are trained on large amounts of speech data.

Our experiments were conducted using two speech corpora: Initial studies were performed using ERBA, a corpus of 10,000 different sentences from the domain of train time table inquiries read by 100 untrained speakers. The text corpus was generated by a grammar. For most of the experiments we used the German part of the VERBMOBIL spontaneous speech corpus. It was obtained from dialogs between two persons who got the task to schedule an appointment. The VERBMOBIL sub–corpora used in this book altogether consist of about 8,300 utterances.

The creation of reference labels for the training and evaluation of prosodic models is an important issue. Note that all the approaches presented in this book do not need time–aligned prosodic labels. It is sufficient that the reference labels are introduced at the correct position in the transliteration of the utterances. The most frequently used approach for the creation of prosodic labels is based on the ToBI system. This relies on the perceptual evaluation of the speech and on the visual inspection of the intonation contour. Such a labeling is very time consuming and it is not as consistently applicable by different labelers as one might expect: agreements are below 85% percent.

Therefore, we developed schemes for the creation of labels based on text corpora. The goal was to accomplish large amounts of labels within rather short time. The labels we created are not merely a prerequisite for the training of classifiers but they are a major mile–stone for the successful integration of prosodic information in syntactic analysis. We showed that classifiers trained on these labels perform much better than classifiers trained on the perceptual labels due to the much larger amount of available training data.

In the case of ERBA the task of label creation was relatively easy because the sentences are grammatically well–formed and because they were automatically generated. Therefore, after the corpus had been read we introduced labels for syntactic boundaries into the grammar and generated the corpus again. Afterwards, a postprocessor deleted some of these boundaries on the basis of prosodic regularities. Based on these boundaries accent labels were placed according to the principle that the rightmost content word in a phrase is accented by default. On a subset of the corpus we showed that there is an over 90% agreement between listener judgments and automatically created labels.

For the VERBMOBIL corpus we defined along the lines of ERBA a scheme for the labeling of prosodic–syntactic labels, called the M labels. Since within short time large amounts of data should be labeled, the scheme is based only on the transliterations and it aims at a rather coarse labeling. We defined ten classes of boundaries taking into account frequent spontaneous speech phenomena such as free phrases and knowledge about prosodic regularities, especially, concerning the boundary marking of such phrases. In the experiments presented in this book we used the three main classes M3 (clause boundary), MU (ambiguous clause boundary) and M0 (no clause boundary). Meanwhile over 7,200 VERBMOBIL utterances have been labeled. The advantage of these labels is not only that large amounts of training data can be provided but also that these labels better meet the requirements of a syntactic module.

Other VERBMOBIL groups developed and applied different boundary labeling schemes: The ToBI scheme was applied to 861 utterances. Several tone labels describing the form of the pitch contour, as well as functional boundary and accent labels are distinguished. We also conducted some experiments on these labels. Furthermore, dialog act boundaries were labeled for a large number of utterances and another group applied a sophisticated scheme for syntactic boundaries to a small VERBMOBIL corpus. In comparing all these labels with our M labels, we found a very high agreement for those boundaries where it could be expected by definition. We concluded that the M labels provide a scheme which allows for a consistent labeling.

Using prosodic information in ASU requires preprocessing, feature extraction and classification. For this book we used the classifiers developed by Andreas Kießling. Sentence mood classification is based on features determined from the intonation contour. Three classes were distinguished: fall (statement), rise (question), and continuation–rise (feedback). On a corpus of isolated read time–of–day expressions a recognition rate of 93% was achieved with an NN. The classifiers for phrase boundaries and accents are based on features determined from the time–alignment of the phoneme sequence underlying the spoken or recognized words. This has as the main advantage that no manually time aligned reference labels are needed and that phoneme intrinsic parameters can be used for normalization. For each word to be classified a feature set is computed for the word itself and a few words in the context. The features include phone durations, intonational and energy information, pause length, and certain lexical information. NNs are used as classifiers. On the ERBA corpus consisting of read speech, three boundary classes and two accent classes were distinguished; the recognition rate is 90% and 95%, respectively. The recognition rate on VERBMOBIL for two prosodic accent classes is 83%. On VERBMOBIL boundary recognition experiments were performed using two classes. The recognition rate was 87% no matter if the acoustic–prosodic ToBI labels or the prosodic–syntactic M labels were taken as a reference for training and testing. The boundary results do not take the ends of utterances into account.

These classifiers make only little use of context information. However, neighboring prosodic events influence each other. Therefore, we experimented with different approaches which are suitable to take context information into account. NN/HMM hybrids were developed for the modeling entire prosodic phrases. Compared to the NN alone, the recognition rate of boundaries could be improved by up to 13 percent points. A greater improvement could be achieved by using multi–level semantic classification trees as an integrated model of acoustic–prosodic and language information.

So far the best results were achieved with the combination of NN and n-grams. In this task the n-grams model the probabilities for clause boundaries given a left and a right context of words. During recognition we insert at the word boundary to be classified all the prosodic classes (one at a time), including a symbol for no–boundary, and calculate the probabilities with the n-gram model. With this approach and using the prosodic–syntactic M labels as reference a clause boundary recognition rate of about 95% could be achieved on the VERBMOBIL data. This reduces the error rate by 62% over the NN alone and by 16% over an NN/n-gram classifier where the n-gram has been trained on the ToBI labels. This again proves the success of our M labeling scheme.

Currently, we consider the combined NN/n-gram classifier to be the best model for the integration with other ASU modules. We solve the integration by prosodic scoring of word graphs, that is, by annotating each of the word hypotheses in the word graph with probabilities for the different prosodic classes. These word graphs are subsequently used by the syntactic analysis. We showed that on VERBMOBIL word graphs a clause boundary recognition rate of 92% can be achieved despite the suboptimal selection of the context words for feature extraction and n-gram scoring.

Our main research effort was spent on the improvement of the parsers integrated in the VERBMOBIL system by prosodic boundary information contained in the word graph. So far we restricted our studies to clause boundaries. Two alternative parsers were investigated. The first one uses a trace unification grammar and a Tomita parser from Siemens which operates on word graphs. This is achieved by applying the parser to partial word chains within an A* search. Goal of the search is to find the optimal word chain and providing the semantic analysis with the parse trees found for this word chain. Using the prosodic boundary probabilities a speed–up of the search by 92% and a reduction in the number of parse trees by 95% was achieved on real spontaneous speech data. The other parser was developed by IBM on the basis of an HPSG formalism. In this approach the n-best word chains extracted from the word graph are used. If prosodic boundary information is introduced in these word chains, the search space of the parser is restricted which results in a speed–up of 46% on speech obtained with the VERBMOBIL prototype; most of the real spontaneous speech data was not analyzable at all without the use of prosodic information.

A corpus–based analysis of particles showed that the position of the focus in an utterance often depends on whether the particle is accented or not. This also affects the interpretation of an utterance. To disambiguate the focus position the VERBMOBIL semantic module makes use of the accent information provided by the prosody module.

In VERBMOBIL, the dialog module has to keep track of the dialog acts. Therefore, these have to be detected within the utterances. Our contribution to this task is the segmentation of utterances into dialog acts. The NN/n-gram classifier achieves a recognition rate of 93%.

In the current implementation of the VERBMOBIL prototype the prosody module fulfills the tasks described above. This prototype has been successfully demonstrated to the public at various occasions including the Hannover CeBit fair and the International Funkaustellung in Berlin.

In the EVAR system we integrated a prosodic sentence mood classifier for the dialog act classification itself. We observed that in human–human dialogs conducted over the telephone the customer frequently repeats the information the officer gives. These repetitions are very often elliptic time–of–day expressions where the sentence mood can only be determined by prosodic means. The sentence mood of these repetitions determines the dialog act and thereby the continuation of the dialog. This scheme has been integrated in the EVAR prototype system and was successfully demonstrated at several occasions including a press conference at the 1994 CRIM/FORWISS Workshop in München.

There is, of course, room for many improvements of the presented models concerning, for example, feature sets and model structures. Apart from this, we believe that research effort should be directed towards a higher integration of prosodic processing with word recognition and with syntactic analysis, because these knowledge sources are highly interrelated. Nevertheless, this book showed that prosodic information can improve automatic speech understanding dramatically, and for the first time a prosody module has been integrated in a fully operational prototype system.

Bibliography

[Abb96] B. Abb. Personal communication, April 1996.

[AD92] M. Adda-Decker and G. Adda. Experiments on Stress–Dependent Phone Modeling for Continuous Speech Recognition. In *Proc. Int. Conf. on Acoustics, Speech and Signal Processing*, volume 1, pages 561–564, San Francisco, CA, 1992.

[Adr91] L.M.H. Adriaens. *Ein Modell deutscher Intonation. Eine experimentell–phonetische Untersuchung nach den perzeptiv relevanten Grundfrequenzänderungen in vorgelesenem Text.* PhD thesis, Technische Universität Eindhoven, 1991.

[Ahl96] U. Ahlrichs. Sprachgesteuerte Fovealisierung und Vergenz. Diploma Thesis, Lehrstuhl für Mustererkennung (Informatik 5), Universität Erlangen–Nürnberg, 1996.

[Ain88] W.A. Ainsworth. *Speech Recognition by Machine*, volume 12 of *IEE Computing Series*. Peter Peregrinus Ltd., London, 1988.

[Ale95] J. Alexandersson, E. Maier, and N. Reithinger. A Robust and Efficient Three–layered Dialogue Component for a Speech–to–speech Translation System. In *Proc. of the 7th Conference of the European Chapter of the ACL (EACL-95)*, pages 188–193, Dublin, 1995.

[Alk92a] P. Alku. An automatic method to estimate the time–based parameters of the glottal pulseform. In *Proc. Int. Conf. on Acoustics, Speech and Signal Processing*, volume 2, pages II–29–II–32, San Francisco, CA, 1992.

[Alk92b] P. Alku. Inverse Filtering of the Glottal Waveform using the Itakura–Saito Distortion Measure. In *Int. Conf. on Spoken Language Processing*, volume 2, pages 847–850, Banff, 1992.

[Alt87] H. Altmann. Zur Problematik der Konstitution von Satzmodi als Formtypen. In J. Meibauer, editor, *Satzmodus zwischen Grammatik und Pragmatik*, pages 22–56. Niemeyer, Tübingen, 1987.

[Alt93] H. Altmann. Satzmodus. In J. Jacobs, A. v. Stechow, W. Sternefeld, and T. Vennemann, editors, *Syntax – Ein Internationales Handbuch Zeitgenössischer Forschung – An International Handbook of Contemporary Research*, volume 1, pages 1006–1029. Walter de Gruyter, Berlin, 1993.

[Amt95] J.W. Amtrup. ICE INTARC Communication Environment Users Guide and Reference Manual, Version 1.3. Verbmobil Technisches Dokument 14, Universität Hamburg, Hamburg, 1995.

[Amt96] J.W. Amtrup and J. Benra. Communication in Large Distributed AI Systems for Natural Language Processing. In *Proc. of the Int. Conf. on Computational Linguistics*, Copenhagen, 1996.

[Ana95] A. Anastasakos, R. Schwartz, and H. Sun. Duration Modeling in Large Vocabulary Speech Recognition. In *Proc. Int. Conf. on Acoustics, Speech and Signal Processing*, volume 1, pages 628–631, Detroit, 1995.

[Asa95] A. Asadi, D. Lubensky, L. Madhavarao, J. Naik, V. Raman, and G. Vyotsky. Combining Speech Algorithms into a “Natural” Application of Speech Technology for Telephone Network Services. In *Proc. European Conf. on Speech Communication and Technology*, volume 1, pages 273–276, Madrid, Spain, 1995.

[Aul84] A.M. Aull. Lexical Stress and its Application to Large Vocabulary Speech Recognition. Master’s thesis, MIT, Cambridge, MA, 1984.

[Aus94] M. Aust, M. Oerder, F. Seide, and V. Steinbiss. Experience with the Philips Automatic Train Timetable Information System. In *2nd IEEE Workshop on Interactive Voice Technology for Telecommunications Applications*, pages 67–72, Kyoto, 1994.

[Aus95] H. Aust, M. Oerder, F. Seide, and V. Steinbiss. The Philips Automatic Train Timetable Information System. *Speech Communication*, 17:249–262, 1995.

[Bac96] M. Bacchiani, M. Ostendorf, Y. Sagisaka, and K. Paliwal. Design of a Speech Recognition System based on Acoustically Derived Segemntal Units. In *Proc. Int. Conf. on Acoustics, Speech and Signal Processing*, volume 1, pages 443–446, Atlanta, 1996.

[Bae95] A. Baekgaard, O. Bernsen, T. Brøndsted, P. Dalsgaard, H. Dybkjær, L. Dybkjær, J. Kristensen, L.B. Larsen, B. Lindberg, B. Maegaard, B. Music, L. Offersgaard, and C. Povlsen. The Danish Language Dialogue Project. In P. Dalsgaard, L.B. Larsen, L. Boves, and I. Thomsen, editors, *Proc. of the ESCA Tutorial and Research Workshop on Spoken Dialogue Systems*, pages 89–92, Vigsø, Denmark, 1995. ESCA.

[Bah89] L.R. Bahl, P.F. Brown, P.V. De Souza, and R.L. Mercer. A Tree–based Statistical Language Model for Natural Language Speech Recognition. *IEEE Trans. on Acoustics, Speech and Signal Processing*, 37(7):1001–1008, 1989.

[Bak94] G. Bakenecker, U. Block, A. Batliner, R. Kompe, E. Nöth, and P. Regel-Brietzmann. Improving Parsing by Incorporating ‘Prosodic Clause Boundaries’ into a Grammar. In *Int. Conf. on Spoken Language Processing*, volume 3, pages 1115–1118, Yokohama, 1994.

[Ban85] R. Bannert. Fokus, Kontrast und Phrasenintonation im Deutschen. *Zeitschrift für Dialektologie und Linguistik*, 52:289–305, 1985.

[Bar95] J. Barnett, P. Bamberg, M. Held, Juan Huerta, L. Manganaro, and A. Weiss. Comparative Performance in Large–vocabulary Isolated–word Recognition in Five European Languages. In *Proc. European Conf. on Speech Communication and Technology*, volume 1, pages 189–192, Madrid, Spain, 1995.

[Bat88] A. Batliner. Modus und Fokus als Dimensionen einer nonmetrischen multidimensionalen Skalierung. In H. Altmann, editor, *Intonationsforschungen*, pages 223–241. Niemeyer, Tübingen, 1988.

[Bat89] A. Batliner and E. Nöth. The Prediction of Focus. In *Proc. European Conf. on Speech Communication and Technology*, pages 210–213, Paris, 1989.

[Bat91] A. Batliner, W. Oppenrieder, E. Nöth, and G. Stallwitz. The Intonational Marking of Focal Structure: Wishful Thinking or Hard Fact? In *Proc. of the 12th Int. Congress of Phonetic Sciences*, volume 3, pages 278–281, Aix–en–Provence, 1991. Université de Provence.

[Bat92] A. Batliner, A. Kießling, R. Kompe, E. Nöth, and B. Raithel. Wann geht der Sonderzug nach Pankow? (Uhrzeitangaben und ihre prosodische Markierung in der Mensch–Mensch– und in der Mensch–Maschine–Kommunikation). In *Fortschritte der Akustik — Proc. DAGA '92*, volume B, pages 541–544, Berlin, 1992.

[Bat93a] A. Batliner, S. Burger, B. Johne, and A. Kießling. MÜSLI: A Classification Scheme For Laryngealizations. In D. House and P. Touati, editors, *Proc. of an ESCA Workshop on Prosody*, pages 176–179. Lund University, Department of Linguistics, Lund, 1993.

[Bat93b] A. Batliner, R. Kompe, A. Kießling, E. Nöth, H. Niemann, and U. Kilian. The Prosodic Marking of Accents and Phrase Boundaries: Expectations and Results. In *Proc. NATO ASI Conference "New Advances and Trends in Speech Recognition and Coding"*, volume 2, pages 89–92, Bubion, 1993.

[Bat93c] A. Batliner, C. Weiand, A. Kießling, and E. Nöth. Why Sentence Modality in Spontaneous Speech is more difficult to classify and why this Fact is not too bad for Prosody. In D. House and P. Touati, editors, *Proc. of an ESCA Workshop on Prosody*, pages 112–115. Lund University, Department of Linguistics, Lund, 1993.

[Bat94a] A. Batliner. Prosody, Focus, and Focal Structure: Some Remarks on Methodology. In P. Bosch and R. van der Sandt, editors, *Focus & Natural Language Processing, Volume 1: Intonation and Syntax*, pages 11–28. IBM Scientific Centre, Heidelberg, 1994.

[Bat94b] A. Batliner, S. Burger, and A. Kießling. Außergrammatische Phänomene in der Spontansprache: Gegenstandsbereich, Beschreibung, Merkmalinventar. Verbmobil Report 57, 1994.

[Bat95a] A. Batliner, R. Kompe, A. Kießling, E. Nöth, and H. Niemann. Can You Tell Apart Spontaneous and Read Speech if You Just Look at Prosody? In A.J. Rubio Ayuso and J.M. López Soler, editors, *Speech Recognition and Coding. New Advances and Trends*, volume 147 of *NATO ASI Series F*, pages 321–324. Springer, Berlin, 1995.

[Bat95b] A. Batliner, R. Kompe, A. Kießling, E. Nöth, H. Niemann, and U. Kilian. The Prosodic Marking of Phrase Boundaries: Expectations and Results. In A.J. Rubio Ayuso and J.M. López Soler, editors, *Speech Recognition and Coding. New Advances and Trends*, volume 147 of *NATO ASI Series F*, pages 325–328. Springer, Berlin, 1995.

[Bat96a] A. Batliner. Personal communication, March 1996.

[Bat96b] A. Batliner, A. Feldhaus, S. Geißler, A. Kießling, T. Kiss, R. Kompe, and E. Nöth. Integrating Syntactic and Prosodic Information for the Efficient Detection of Empty Categories. In *Proc. of the Int. Conf. on Computational Linguistics*, pages 71–76, Copenhagen, 1996.

[Bat96c] A. Batliner, A. Feldhaus, S. Geissler, A. Kießling, T. Kiss, R. Kompe, and E. Nöth. Integrating Syntactic and Prosodic Information for the Efficient Detection of Empty Categories. In *Proc. of the Int. Conf. on Computational Linguistics*, Copenhagen, 1996.

[Bat96d] A. Batliner, A. Feldhaus, S. Geißler, T. Kiss, R. Kompe, and E. Nöth. Prosody, Empty Categories and Parsing — A Success Story. In *Int. Conf. on Spoken Language Processing*, volume 2, pages 1169–1172, Philadelphia, 1996.

[Bea90] J. Bear and P.J. Price. Prosody, Syntax, and Parsing. In *Proceedings of the 28th Conference of the Association for Computational Lingustics*, pages 17–22, Banff, 1990.

[Bec80] M. Beckman. *The Structure of Intonational Meaning: Evidence from English.* Indiana University Linguistics Club, Bloomington, 1980.

[Bec86] M. Beckman. *Stress and Non–stress Accent.* Foris Publications, Dordrecht, 1986.

[Bec90] M.E. Beckman, M.G. Swora, J. Rauschenberg, and K. de Jong. Stress Shift, Stress Clash, and Polysyllabic Shortening in a Prosodically Annotated Discourse. In *Int. Conf. on Spoken Language Processing*, volume 1, pages 5–8, Kobe, 1990.

[Bec94] M.E. Beckman and G. Ayers. Guidelines for ToBI transcription. version 2., 1994.

[Bel57] R. Bellman. *Dynamic Programming*. Princeton University Press, 1957.

[Bel72] R. E. Bellman. *Dynamic Programming*. Princeton University Press, 1972.

[Ben91] Y. Bengio, R. De Mori, G. Flammia, and R. Kompe. Phonetically Motivated Acoustic Parameters for Continuous Speech Recognition using

Artificial Neural Networks. In *Proc. European Conf. on Speech Communication and Technology*, pages 551–555, Genova, 1991.

[Ben92a] Y. Bengio, R. De Mori, G. Flammia, and R. Kompe. Global optimization of a neural network – hidden markov model hybrid. *IEEE Trans. on Neural Networks*, 3(2):252–259, 1992.

[Ben92b] Y. Bengio, R. De Mori, G. Flammia, and R. Kompe. Phonetically Motivated Acoustic Parameters for Continuous Speech Recognition using Artificial Neural Networks. *Speech Communication*, 11(2–3):261–271, 1992.

[Ben94] Y. Bengio, P. Simard, and P. Frasconi. Learning Long–term Dependencies with Gradient Descent is Difficult. *IEEE Trans. on Neural Networks*, 5(2):157–166, 1994.

[Ben95] Y. Bengio. *Neural Networks for Speech and Sequence Recognition.* International Thomson Computer Press, London, 1995.

[Bez94] J.C. Bezdek. What is computational intelligence? In J.M. Zurada, R.J. Marks, and C.J. Robinson, editors, *Computational Intelligence Imitating Life*, pages 1–12. IEEE, New York, 1994.

[Bie66] M. Bierwisch. Regeln für die Intonation Deutscher Sätze. In *Studia Grammatica VII: Untersuchungen über Akzent und Intonation im Deutschen*, pages 99–201. Akademie–Verlag, Berlin, 1966.

[Bis92] K. Bishop. Modeling Sentential Stress in the Context of a Large Vocabulary Continuous Speech Recognizer. In *Int. Conf. on Spoken Language Processing*, volume 1, pages 437–440, Banff, 1992.

[Bla91] E. Blaauw. Phonetic Characteristics of Spontaneous and Read–aloud Speech. In *Proceedings of the ESCA Workshop. Phonetics and Phonology of Speaking Styles: Reduction and Elaboration in Speech Communication*, pages 12–1 – 12–5, Barcelona, 1991.

[Bla92] E. Black, F. Jelinek, J. Lafferty, R. Mercer, and S. Roukos. Decision Tree Models Applied to the Labeling of Text with Parts–of–Speech. In *Speech and Natural Language Workshop*, pages 117–121. Morgan Kaufmann, 1992.

[Blo92] H.U. Block and S. Schachtl. Trace & Unification Grammar. In *Proc. of the Int. Conf. on Computational Linguistics*, volume 1, pages 87–93, Nantes, 1992.

[Blo94] U. Block. Personal communication, Sep. 1994.

[Boc95] E. Bocchieri and G. Riccardi. State Typing of Triphone HMM's for the 1994 AT&T ARPA ATIS Recognizer. In *Proc. European Conf. on Speech Communication and Technology*, volume 3, pages 1499–1502, Madrid, Spain, 1995.

[Bos94a] J. Bos. Focusing Particles & Ellipsis Resolution. Verbmobil Report 61, Universität des Saarlandes, Saarbrücken, 1994.

[Bos94b] J. Bos, E. Mastenbroek, S. McGlashan, S. Millies, and M. Pinkal. A

Compositional DRS–based Formalism for NLP Applications. In *Int. Workshop on Computational Semantics*, pages 21–31, Tilburg, 1994.

[Bos94c] J. Bos, E. Mastenbroek, S. McGlashan, S. Millies, and M. Pinkal. The Verbmobil Semantic Formalism. Verbmobil Report 59, Universität des Saarlandes, Saarbrücken, 1994.

[Bos95] J. Bos, A. Batliner, and R. Kompe. On the Use of Prosody for Semantic Disambiguation in Verbmobil. Verbmobil Memo 82, 1995.

[Bos96a] J. Bos. Personal communication, July 1996.

[Bos96b] J. Bos. Predicate Logic Unplugged. In *Proc. of the 10th Amsterdam Colloquium*, Amsterdam, 1996. Univeristy of Amsterdam.

[Bos96c] J. Bos, M. Egg, and M. Schiehlen. Definition of the Abstract Semantic Classes for the Verbmobil Forschungsprototyp. Verbmobil Report (to appear), Universität des Saarlandes, Saarbrücken, 1996.

[Bos96d] J. Bos, B. Gambaeck, Ch. Lieske, Y. Mori, M. Pinkal, and K. Worm. Compositional Semantics in Verbmobil. In *Proc. of the Int. Conf. on Computational Linguistics*, Copenhagen, 1996.

[Bos96e] N. Bos. Personal communication, July 1996.

[Bou92] H. Bourlard, N. Morgan, C. Wooters, and S. Renals. CDNN: A Context–Dependent Neural Network for Continuous Speech Recognition. In *Proc. Int. Conf. on Acoustics, Speech and Signal Processing*, volume 2, pages 349–352, San Francisco, CA, 1992.

[Bou94a] H. Bourlard, Y. Konig, and N. Morgan. REMAP: Recursive Estimation and Maximization of A Posteriori Probabilities — Application to Transition–Based Continuous Speech Recognition. TR–94–064, ICSI, Berkeley, 1994.

[Bou94b] H. Bourlard and N. Morgan. *Connectionist Speech Recognition: A Hybrid Approach*. Kluwer Academic Publishers, Boston, MA, 1994.

[Bou95a] P. Bourdot, M. Krus, and R. Gherbi. Management of Non–Standard Devices for Multimodal User Interfaces under UNIX / X11. In H. Bunt, R.-J. Beun, and T. Borghuis, editors, *Proc. International Conference on Cooperative Multimodal Communication CMC*, Eindhoven, 1995. Samenwerkingsorgaan Brabantse Universiteiten.

[Bou95b] H. Bourlard, Y. Konig, and N. Morgan. REMAP: Recursive Estimation and Maximization of A Posteriori Probabilities in Continuous Speech Recognition. In *Proc. European Conf. on Speech Communication and Technology*, volume 3, pages 1663–1666, Madrid, Spain, 1995.

[Bre84] L. Breiman. *Classification and Regression Trees*. Wadsworth, Belmont CA, 1984.

[Bri87] A. Brietzmann. Stufenweise Syntaktische Analyse mit integrierter Bewertung für die kontinuierliche Spracherkenunng. Technical Report 9, Arbeitsberichte des IMMD der Universität Erlangen–Nürnberg, Erlangen, 1987.

[Bri90a] J. Bridle. Alphanets: A Recurrent 'Neural' Network Architecture with a Hidden Markov Model Interpretation. *Speech Communication*, 9(1):83–92, 1990.

[Bri90b] J.S. Bridle. Training Stochastic Model Recognition Algorithms as Networks can Lead to Maximum Mutial Information Estimation of Parameters. In D.S. Touretzky, editor, *Advances in Neural Information Processing Systems 2*, pages 211–217, San Mateo, 1990. Morgan Kaufmann.

[Bri94] A. Brietzmann, F. Class, U. Ehrlich, P. Heisterkamp, A. Kaltenmeier, K. Mecklenburg, P. Regel-Brietzmann, G. Hanrieder, and W. Hiltl. Robust Speech Understanding. In *Int. Conf. on Spoken Language Processing*, pages 967–970, Yokohama, Japan, 1994.

[Brø96] T. Brøndsted. Adapting a Prosody Classification Module from German to Danish – a Contrastive Analysis. In J.-P. Haton, editor, *Proc. of the Workshop on Multi–lingual Spontaneous Speech Recognition in Real Environments*, Nancy, 1996.

[Buß90] Hadumod Bußmann. *Lexikon der Sprachwissenschaft*. Alfred Kröner Verlag, Stuttgart, 2 edition, 1990.

[But95] M. Butt. Transfer I: Tense and Aspect. Verbmobil Report 55, Universität Tübingen, Tübingen, 1995.

[Cah92] J. Cahn. An Investigation into the Correlation of Cue Phrases, Unfilled Pauses and the Structuring of Spoken Discourse. In *Proc. IRCS Workshop on Prosody in Natural Speech*, pages 19–30, Pennsylvania, 1992. University of Pennsylvania.

[Cam90] Nick Campbell. Evidence for a Syllable–based Model of Speech Timing. In *Int. Conf. on Spoken Language Processing*, volume 1, pages 9–12, Kobe, 1990.

[Cam92] Nick Campbell. Prosodic Encoding of English Speech. In *Int. Conf. on Spoken Language Processing*, volume 1, pages 663–666, Banff, 1992.

[Cam94] Nick Campbell. Combining the Use of Duration and F0 in an Automatic Analysis of Dialogue Prosody. In *Int. Conf. on Spoken Language Processing*, volume 3, pages 1111–1114, Yokohama, 1994.

[Cam95a] Nick Campbell. From Read Speech to Real Speech. In *Proc. of the 13th Int. Congress of Phonetic Sciences*, volume 2, pages 20–27, Stockholm, 1995.

[Cam95b] Nick Campbell. Prosodic Influence on Segmental Quality. In *Proc. European Conf. on Speech Communication and Technology*, volume 2, pages 1011–1014, Madrid, 1995.

[Car86] D.W. Carroll. *Psychology of Language*. Brooks/Cole Publishing Company, Pacific Grove, 1986.

[Car94] F. Caroli, R. Nübel, B. Ripplinger, and J. Schütz. Transfer in Verbmobil. Verbmobil Report 11, IAI, Saarbrücken, 1994.

[Cha76] W. Chafe. *Givenness, Contrastiveness, Definitness, Subjects, Topics and Point of View*. Academic Press, New York, 1976.

[Cho81] N. Chomsky. *Lectures on Government and Binding*, volume 9 of *Studies in Generative Grammar*. Foris, Dordrecht, 1981.

[Cla92] F. Class, A. Kaltenmeier, P. Regel, and K. Trottler. Fast Speaker Adaptation Combined with Soft Vector Quanization in an HMM Speech Recognition System. In *Proc. Int. Conf. on Acoustics, Speech and Signal Processing*, volume 1, pages 461–464, San Francisco, CA, 1992.

[Cla93a] F. Class, A. Kaltenmeier, and P. Regel. Evaluation of an HMM Speech Recognizer with Various Continuous Speech Databases. In *Proc. European Conf. on Speech Communication and Technology*, volume 2, pages 803–806, Berlin, 1993.

[Cla93b] F. Class, A. Kaltenmeier, and P. Regel. Optimization of an HMM-based continuous Speech Recognizer. In *Proc. European Conf. on Speech Communication and Technology*, volume 3, pages 1587–1590, Berlin, 1993.

[Cop95] A. Copestake. Semantic Transfer in Verbmobil. Verbmobil Report 93, CSLI, Stanford, 1995.

[Cor93] A. Corazza, M. Federico, R. Gretter, and G. Lazzari. Design and Acquisition of a Task–Oriented Spontaneous–Speech Database. In V. Roberto, editor, *Intelligent Perceptual Systems, Lecture Notes in Artificial Intelligence*, pages 196–210, Heidelberg, 1993. Springer Verlag.

[Cry79] D. Crystal. Prosodic Development. In P. Fletcher and M. Garman, editors, *Language Acquisition. Studies in First Language Development*, pages 33–48. Cambridge University Press, Cambridge, MA, 1979.

[Cry82] T.H. Crystal and A.S. House. Segmental durations in connected–speech signal: Preliminary results. *Journal of the Acoustic Society of America*, 72:705–716, 1982.

[Cry90] T.H. Crystal and A.S. House. Articulation Rate and the Duration of Syllables and Stress Groups in Connected Speech. *Journal of the Acoustic Society of America*, 88(1):101–112, 1990.

[Cum90] K.E. Cummings and M.A. Clements. Analysis of Glottal Waveforms Across Stress Styles. In *Proc. Int. Conf. on Acoustics, Speech and Signal Processing*, volume 1, pages 369–372, Albuquerque, 1990.

[Cut83] A. Cutler and D.R. Ladd, editors. *Prosody: Models and Measurements*. Springer–Verlag, Berlin, 1983.

[Dal90] N.A. Daly and V.W. Zue. Acoustic, Perceptual, and Linguistic Analyses of Intonation Contours in Human/Maschine Dialogues. In *Int. Conf. on Spoken Language Processing*, pages 497–500, Kobe, 1990.

[Dal92] N. Daly and V. Zue. Statistical and Linguistic Analyses of F0 in Read and Spontaneous Speech. In *Int. Conf. on Spoken Language Process-*

ing, volume 1, pages 763–766, Banff, 1992.

[Dal94a] P. Dalsgaard and A. Baekgaard. Spoken Language Dialog Systems. In H. Niemann, R. De Mori, and G. Hanrieder, editors, *Progress and Prospects of Speech Research and Technology: Proc. of the CRIM / FORWISS Workshop*, PAI 1, pages 178–191. Infix, 1994.

[Dal94b] N.A. Daly. *Acoustic–Phonetic and Linguistic Analyses of Spontaneous Speech: Implications for Speech Understanding*. PhD thesis, MIT, 1994.

[Dem77] A.P. Dempster, N.M. Laird, and D.B. Rubin. Maximum Likelihood from Incomplete Data via the EM Algorithm. *J. Royal Statistical Society*, 39(1):1–38, 1977.

[Den92] J. Denzler. Transformation von Sprachsignalen in Laryngosignale mittels künstlicher neuronaler Netze. Diploma Thesis, Lehrstuhl für Mustererkennung (Informatik 5), Universität Erlangen–Nürnberg, 1992.

[Den93] J. Denzler, R. Kompe, A. Kießling, H. Niemann, and E. Nöth. Going Back to the Source: Inverse Filtering of the Speech Signal with ANNs. In *Proc. European Conf. on Speech Communication and Technology*, volume 1, pages 111–114, Berlin, 1993.

[Der86] A. Derouault and B. Merialdo. Natural Language Modelling for Phoneme–to–Text Transcription. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 8(6):742–749, 1986.

[Dig90] V. Digilakis, M. Ostendorf, and J.R. Rohlicek. Fast Search Algorithms for Connected Phone Recognition Using the Stochastic Segment Model. In *Speech and Natural Language Workshop*, pages 173–178. Morgan Kaufmann, Hidden Valley, Pennsylvania, 1990.

[DK95] N.A. Daly-Kelly. Linguistic and Acoustic Characteristics of Pause Intervals in Spontaneous Speech. In *Proc. European Conf. on Speech Communication and Technology*, volume 2, pages 1023–1026, Madrid, Spain, 1995.

[DM88] R. De Mori. Planning, Neural Networks and Markov Models for Automatic Speech Recognition. In *Proc. Int. Conf. on Pattern Recognition*, volume 1, pages 395–402, Rome, 1988.

[DM90] R. De Mori, R. Kuhn, G. Lazzari, R. Gretter, and L. Stringa. Modelling Operator–Robot Oral Dialogue for Application in Telerobotics. In *Proc. Int. Conf. on Pattern Recognition*, pages 246–248, 1990.

[DM93] R. De Mori and G. Flammia. Speaker–independent Consonant Classification in Continuous Speech with distinctive Features and Neural Networks. *Journal of the Acoustic Society of America*, 94(6):3091–3104, 1993.

[DM95] R. De Mori, R. Kuhn, M. Galler, and C. Snow. Speech Recognition and Understanding. In J. Liebowitz and D.S. Prerau, editors, *Worldwide Intelligent Systems*, pages 125–162. IOS Press, 1995.

[DM96a] R. De Mori. Comments on: "Towards Increasing Speech Recognition Error Rates" by H. Bourlard, H. Hermansky, and N. Morgan. *Speech Communication*, 18:234–235, 1996.

[DM96b] R. De Mori and M. Galler. The Use of Syllable Phonotactics for Word Hypothesization. In *Proc. Int. Conf. on Acoustics, Speech and Signal Processing*, volume 2, pages 877–880, Atlanta, 1996.

[Dor94] M. Dorna, K. Eberle, M. Emele, and C.J. Rupp. Semantik–orientierter rekursiver Transfer in HPSG am Beispiel des Referenzdialogs. Verbmobil Report 39, Universität Stuttgart, Stuttgart, 1994.

[dP93] J.R. de Pijper and A. Sanderman. Prosodic Cues to the Perception of Constituent Boundaries. In *Proc. European Conf. on Speech Communication and Technology*, volume 2, pages 1211–1214, Berlin, Germany, 1993.

[Dud73] R.O. Duda and P.E. Hart. *Pattern Classification and Scene Analysis*. John Wiley, New York, 1973.

[Dud94] M. Duda. Lexicon Access on Parallel Machines. Verbmobil Report 10, Humboldt–Universität zu Berlin, Berlin, 1994.

[Dug95] C. Dugast, X. Aubert, and R. Kneser. The Philips Large–vocabulary Recognition System for American English, French and German. In *Proc. European Conf. on Speech Communication and Technology*, volume 1, pages 197–200, Madrid, Spain, 1995.

[Dum93] P. Dumouchel and D. O'Shaughnessy. Prosody and Continuous Speech Recognition. In *Proc. European Conf. on Speech Communication and Technology*, volume 3, pages 2195–2198, Berlin, 1993.

[Dum94a] P. Dumouchel. Suprasegmental Features and Continuous Speech Recognition. In *Proc. Int. Conf. on Acoustics, Speech and Signal Processing*, volume 2, pages 177–180, Adelaide, 1994.

[Dum94b] Pierre Dumouchel. *La prosodie et la reconnaissance automatique de la parole*. PhD thesis, Université du Québec, 1994.

[Dum95] P. Dumouchel and D. O'Shaughnessy. Segmental Duration and HMM Modeling. In *Proc. European Conf. on Speech Communication and Technology*, volume 2, pages 803–806, Madrid, Spain, 1995.

[DV94] J.E. Diaz-Verdejo, J.C. Sgura-Luna, P. Garcia-Teodoro, and A.J. Rubio-Ayuso. SLHMM: A Continuous Speech Recognition System Based on Alphanet–HMM. In *Proc. Int. Conf. on Acoustics, Speech and Signal Processing*, volume 1, pages 213–216, Adelaide, 1994.

[Ebe96] K. Eberle. Disambiguation by Information Structure in DRT. In *Proc. of the Int. Conf. on Computational Linguistics*, Copenhagen, 1996.

[Eck93] W. Eckert, T. Kuhn, H. Niemann, S. Rieck, A. Scheuer, and E. G. Schukat-Talamazzini. A Spoken Dialogue System for German Intercity Train Timetable Inquiries. In *Proc. European Conf. on Speech Communication and Technology*, pages 1871–1874, Berlin, 1993.

[Eck96] W. Eckert. Gesprochener Mensch–Maschine–Dialog. Dissertation, Technische Fakultät der Universität Erlangen–Nürnberg, 1996.

[Ehr88] U. Ehrlich and H. Niemann. Using Semantic and Pragmatic Knowledge for the Interpretation of Syntactic Constituents. In H. Niemann, M. Lang, and G. Sagerer, editors, *Recent Advances in Speech Understanding and Dialog Systems*, volume 46 of *NATO ASI Series F*, pages 485–490. Springer–Verlag, Berlin, 1988.

[Ehr90] U. Ehrlich. *Bedeutungsanalyse in einem sprachverstehenden System unter Berücksichtigung pragmatischer Faktoren*, volume 22 of *Sprache und Information*. Niemeyer, Tübingen, 1990.

[Elm90] J.L. Elman. Finding Structure in Time. *Cognitive Science*, 14:179–211, 1990.

[Eng77] U. Engel. *Syntax der deutschen Gegenwartssprache*. Erich Schmidt Verlag, Berlin, 1977.

[Fah89] S.E. Fahlman. Faster–learning Variations on Back–propagation: An Empirical Study. In *Proc. of the 1988 Connectionist Summer School*, pages 38–51. Morgan Kaufmann, San Mateo, 1989.

[Fah90] S.E. Fahlman and C. Lebiere. The Cascade–correlation Learning Architecture. In D.S. Touretzky, editor, *Advances in Neural Information Processing Systems 2*, pages 524–532, San Mateo, 1990. Morgan Kaufmann.

[Fal90] F. Fallside, H. Lucke, T.P. Marsland, P.J. O'Shea, M.S.J. Owen, R.W. Prager, A.J. Robinson, and N.H. Russell. Continuous Speech Recognition for the TIMIT Database using Neural Networks. In *Proc. Int. Conf. on Acoustics, Speech and Signal Processing*, volume 1, pages 445–448, Albuquerque, 1990.

[Fel95] A. Feldhaus and T. Kiss. Kategoriale Etikettierung der Karlsruher Dialoge. Verbmobil Memo 94, 1995.

[Fér93] C. Féry. *German Intonational Patterns*. Niemeyer, Tübingen, 1993.

[Fer95] J. Ferreiros, R. de Córdoba, M.H. Savoji, and J.M. Pardo. Continuous Speech HMM Training System: Applications to Speech Recognition and Phonetic Label Alignment. In A.J. Rubio Ayuso and J.M. López Soler, editors, *Speech Recognition and Coding. New Advances and Trends*, volume 147 of *NATO ASI Series F*, pages 68–71. Springer, Berlin, 1995.

[Fil68] Ch. Fillmore. A case for case. In E. Bach and R. T. Harms, editors, *Universals in Linguistic Theory*, pages 1–88. Holt, Rinehart and Winston, New York, 1968.

[Fin95] W. Finkler and A. Kilger. Der englische Generator im Demonstrator. Verbmobil Memo 71, DFKI, Saarbrücken, 1995.

[Fis94] J. Fischer. Integrierte Erkennung von Phrasengrenzen und Phrasenakzenten mit Klassifikationsbäumen. Diploma Thesis, Lehrstuhl für Mustererkennung (Informatik 5), Universität Erlangen–Nürnberg, 1994.

[Fis95] K. Fischer. Lexikonaufbau: Diskurspartikel–Datenbank. Verbmobil Memo 85, 1995.

[Fla92] G. Flammia, P. Dalsgaard, O. Andersen, and B Lindberg. Segment based variable frame rate speech analysis and recognition using a spectral variation functio. In *Int. Conf. on Spoken Language Processing*, volume 2, pages 983–986, Banff, 1992.

[Fou89] A.J. Fourcin, G. Harland, W. Barry, and V. Hazan, editors. *Speech Input and Output Assessment.* Ellis Horwood, Chichester, 1989.

[Fra91] N.M. Fraser and G.N. Gilbert. Simulating Speech Systems. *Computer Speech & Language*, 5(1):81–99, 1991.

[Fuj90] H. Fujisaki, K. Hirose, and N. Takahashi. Manifestation of Linguistic and Para–linguistic Information in the Voice Fundamental Frequency Contours of Spoken Japanese. In *Int. Conf. on Spoken Language Processing*, volume 1, pages 485–488, Kobe, 1990.

[Gal96] F. Gallwitz, E.G. Schukat-Talamazzini, and H. Niemann. Integrating Large Context Language Models into a Real Time Word Recognizer. In N. Pavesic and H. Niemann, editors, *3rd Slovenian–German and 2nd SDRV Workshop.* Faculty of Electrical and Computer Engineering, University of Ljubljana, Ljubljana, 1996.

[Gau95] J.L. Gauvin, L. Lamel, and M. Adda-Decker. Developments in Continuous Speech Dictation using the ARPA WSJ Task. In *Proc. Int. Conf. on Acoustics, Speech and Signal Processing*, volume 1, pages 65–68, Detroit, 1995.

[Geb95] A. Gebhard. Entwicklung eines Systems für vielschichtige Klassifikationsbäume. Diploma Thesis, Lehrstuhl für Mustererkennung (Informatik 5), Universität Erlangen–Nürnberg, 1995.

[Geh96] M. Gehrke. Personal communication, June 1996.

[Gei94a] S. Geißler. Lexikalische Regeln in der IBM–Basisgrammatik. Verbmobil Report 20, IBM, Heidelberg, 1994.

[Gei94b] A. Geist, A. Beguelin, J. Dongarra, W. Jiang, R. Manchek, and V. Sunderam. Pvm3 User's Guide and Reference Manual. Technical Report ORNL/TM-12187, Oak Ridge National Laboratory, Oak Ridge, 1994.

[Gei95] S. Geissler. Personal communication, 1995.

[Gel91] S. Gelfand, C. Ravishankar, and E. Delp. An Iterative Growing and Pruning Algorithm for Classification Tree Design. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 13:302–320, 1991.

[Gie95] M. Giebner. Bestimmung der Anregungsart von Sprache mit einem HMM/NN–Hybridverfahren. Diploma Thesis, Lehrstuhl für Mustererkennung (Informatik 5), Universität Erlangen–Nürnberg, 1995.

[Gis90] H. Gish. A Probabilistic Approach to the Understanding and Training of Neural Network Classifiers. In *Proc. Int. Conf. on Acoustics, Speech and Signal Processing*, volume 3, pages 1361–1364, Albuquerque, 1990.

[Gra84] R.M. Gray. Vector Quantization. *IEEE ASSP Magazine*, 1(2):4–29, 1984.

[Gri96] M. Grice, M. Reyelt, R. Benzmüller, J. Mayer, and A. Batliner. Consistency in Transcription and Labelling of German Intonation with GToBI. In *Int. Conf. on Spoken Language Processing*, Philadelphia, 1996.

[Gue90] Y. Guedon and C. Cocozza-Thivent. Explicit State Occupancy Modelling by Hidden Semi–Markov Models: Application of Derin's Scheme. *Computer Speech & Language*, 4(2):167–192, 1990.

[Haa95a] J. Haas. Ein Modul zur (halb)automatischen Transliteration und erzwungenen Erkennung. Diploma Thesis, Lehrstuhl für Mustererkennung (Informatik 5), Universität Erlangen–Nürnberg, 1995.

[Haa95b] J. Haas, A. Kießling, E. Nöth, H. Niemann, and A. Batliner. Contrastive Accents – how to get them and what do they look like. In *Proc. of the 13th Int. Congress of Phonetic Sciences*, volume 4, pages 276–279, Stockholm, 1995.

[Haf90] P. Haffner and A. Waibel. Multi–state Time–delay Neural Networks for Continuous Speech Recognition. In D.S. Touretzky, editor, *Advances in Neural Information Processing Systems 2*, pages 579–588. Morgan Kaufmann, San Mateo, 1990.

[Haf93] B. Haftka. Topologische Felder und Versetzungsphänomene. In J. Jacobs, A. v. Stechow, W. Sternefeld, and T. Vennemann, editors, *Syntax – Ein Internationales Handbuch Zeitgenössischer Forschung – An International Handbook of Contemporary Research*, volume 1, pages 846–867. Walter de Gruyter, Berlin, 1993.

[Haf94] P. Haffner. A New Probabilistic Framework for Connectionist Time Alignment. In *Int. Conf. on Spoken Language Processing*, volume 1, pages 621–624, Yokohama, Japan, 1994.

[Han95] G. Hanrieder and G. Görz. Robust Parsing of Spoken Dialogue Using Contextual Knowledge and Recognition Probabilities. In P. Dalsgaard, L.B. Larsen, L. Boves, and I. Thomsen, editors, *Proc. of the ESCA Tutorial and Research Workshop on Spoken Dialogue Systems*, pages 57–60. ESCA, Vigsø, Denmark, 1995.

[Har94] St. Harbeck. Entwicklung eines robusten Systems zur periodensynchronen Analyse der Grundfrequenz von Sprachsignalen. Diploma Thesis, Lehrstuhl für Mustererkennung (Informatik 5), Universität Erlangen–Nürnberg, 1994.

[Har95] S. Harbeck, A. Kießling, R. Kompe, H. Niemann, and E. Nöth. Robust pitch period detection using dynamic programming with an ANN cost function. In *Proc. European Conf. on Speech Communication and Technology*, volume 2, pages 1337–1340, Madrid, 1995.

[Har96] S. Harbeck. Personal communication, April 1996.

[Hed90] P. Hedelin and D. Huber. Pitch Period Determination of Aperiodic Speech Signals. In *Proc. Int. Conf. on Acoustics, Speech and Signal Processing*, volume 1, pages 361–364, 1990.

[Hei69] G. Heike. *Suprasegmentale Analyse.* Elwert Verlag, Marburg, 1969.

[Hei95] J.E. Heine and K.L. Worm. Semantic Phenomena for German with Examples. Verbmobil Memo 86, Universität des Saarlandes, Saarbrücken, 1995.

[Hel85] H. Helfrich. *Satzmelodie und Sprachwahrnehmung.* Walter de Gruyter, Berlin, 1985.

[Hes83] W. Hess. *Pitch Determination of Speech Signals*, volume 3 of *Springer Series of Information Sciences.* Springer–Verlag, Berlin, 1983.

[Hes95] W. Hess, K.J. Kohler, and H.-G. Tillmann. The Phondat–Verbmobil Speech Corpus. In *Proc. European Conf. on Speech Communication and Technology*, volume 1, pages 863–866, Madrid, Spain, 1995.

[Hes96] W. Hess, A. Batliner, A. Kießling, R. Kompe, E. Nöth, A. Petzold, M. Reyelt, and V. Strom. Prosodic Modules for Speech Recognition and Understanding in Verbmobil. In Yoshinori Sagisaka, Nick Campell, and Norio Higuchi, editors, *Computing Prosody. Approaches to a Computational Analysis and Modelling of the Prosody of Spontaneous Speech*, pages 363–383. Springer–Verlag, New York, 1996.

[Hir93a] J. Hirschberg. Pitch Accent in Context: Predicting Intonational Prominence from Text. *Artificial Intelligence*, 19(63):305–340, 1993.

[Hir93b] J. Hirschberg. Studies of Intonation and Discourse. In D. House and P. Touati, editors, *Proc. of an ESCA Workshop on Prosody*, pages 90–95. Lund University, Department of Linguistics, Lund, 1993.

[Hir93c] J. Hirschberg and D. Litman. Empirical Studies on the Disambiguation of Cue Phrases. *Computational Linguistics*, 19(3):501–529, 1993.

[Hir93d] J. Hirschberg and C. Nakatani. A Speech–First Model for Repair Identification in Spoken Language Systems. In *Proc. European Conf. on Speech Communication and Technology*, volume 2, pages 1173–1176, Berlin, 1993.

[Hir94a] J. Hirschberg and B.J. Grosz. Intonation and Discourse Structure in Spontaneous and Read Direction–Giving. In *Proc. of the International Symposium on Prosody*, pages 103–109, Yokohama, 1994.

[Hir94b] D. Hirst. The Symbolic Coding of Fundamental Frequency Curves: From Acoustics to Phonology. In *Proc. of the International Symposium on Prosody*, pages 1–5, Yokohama, 1994.

[Hit89] L. Hitzenberger and H. Kritzenberger. Simulation Experiments and Prototyping of User Interfaces in a Multimedial Environment of an Information System. In *Proc. European Conf. on Speech Communication and Technology*, volume 2, pages 597–600, Paris, 1989.

[Hol96] W.J. Holmes and Russell M.J. Modeling Speech Variability with Segmental HMMs. In *Proc. Int. Conf. on Acoustics, Speech and Signal Processing*, volume 1, pages 447–450, Atlanta, 1996.

[Hou92] J. House and N. Youd. Evaluating the Prosody of Synthesized Utterances within a Dialogue System. In *Int. Conf. on Spoken Language Processing*, volume 2, pages 1175–1178, Banff, 1992.

[Hua90] X.D. Huang, Y. Ariki, and M.A. Jack. *Hidden Markov Models for Speech Recognition.* Information Technology Series. Edinburgh University Press, Edinburgh, 1990.

[Hua93] X.D. Huang, W. Hon, M. Hwang, and K.-F. Lee. A Comparative Study of Discrete, Semicontinuous and Continuous Hidden Markov Models. *Computer Speech & Language*, 7(4):359–368, 1993.

[Hub88] D. Huber. *Aspects of the Communicative Function of Voice in Text Intonation.* PhD thesis, Chalmers University, Göteborg/Lund, 1988.

[Hub90] D. Huber. Prosodic Transfer in Spoken Language Interpretation. In *Int. Conf. on Spoken Language Processing*, volume 1, pages 509–512, Kyoto, 1990.

[Hub92] D. Huber. Perception of Aperiodic Speech Signals. In *Int. Conf. on Spoken Language Processing*, volume 1, pages 503–506, Banff, 1992.

[Hun93] A. Hunt. Utilizing Prosody to Perform Syntactic Disambiguation. In *Proc. European Conf. on Speech Communication and Technology*, volume 2, pages 1339–1342, Berlin, 1993.

[Hun94a] A. Hunt. A Generalised Model for Utilising Prosodic Information in Continuous Speech Recognition. In *Proc. Int. Conf. on Acoustics, Speech and Signal Processing*, volume 2, pages 169–172, Adelaide, 1994.

[Hun94b] Andrew Hunt. A Prosodic Recognition Module based on Linear Discriminant Analysis. In *Int. Conf. on Spoken Language Processing*, volume 3, pages 1119–1122, Yokohama, 1994.

[Hun95a] A. Hunt. *Models of Prosody and Syntax and their Application to Automatic Speech Recognition.* PhD thesis, University of Sydney, 1995.

[Hun95b] A. Hunt. Syntactic Influence on Prosodic Phrasing in the Framework of the Link Grammar. In *Proc. European Conf. on Speech Communication and Technology*, volume 2, pages 997–1000, Madrid, Spain, 1995.

[Int63] International Phonetic Association. The principals of the ipa, 1963.

[Ipš95] I. Ipšić, F. Mihelič, E.G. Schukat-Talamazzini, and N. Pavešić. Generating Word Hypotheses in the Slovene Continuous Speech Recognition System. In Walter G. Kropatsch Franc Solina, editor, *Visual Modules. Proc. of 19. ÖAGM and 1. SDVR Workshop*, volume 81 of *Schriften-*

reihe der "Osterreichischen Computer Gesellschaft. R. Oldenbourg, Wien, München, 1995.

[Jac88] J. Jacobs. Fokus–Hintergrund–Gliederung und Grammatik. In H. Altmann, editor, *Intonationsforschungen*, pages 89–134. Niemeyer, Tübingen, 1988.

[Jek95] S. Jekat, A. Klein, E. Maier, I. Maleck, M. Mast, and J. Quantz. Dialogue Acts in Verbmobil. Verbmobil Report 65, 1995.

[Jel90] F. Jelinek. Self–organized Language Modeling for Speech Recognition. In A. Waibel and K.-F. Lee, editors, *Readings in Speech Recognition*, pages 450–506. Morgan Kaufmann Publishers Inc., San Mateo, California, 1990.

[Jen94] U. Jensen, R. K. Moore, P. Dalsgaard, and B. Lindberg. Modelling intonation contours at the phrase level using continuous density hidden markov models. *Computer Speech & Language*, 8(3):247–260, 1994.

[Jor86] M. Jordan. Attractor Dynamics and Parallelism in a Connectionist Sequential Machine. In *Proc. 1986 Cognitive Science Conference*, volume 21, pages 531–546. L. Erlbaum, 1986.

[Jor94] M.I. Jordan and R.A. Jacobs. Hierarchical Mixtures of Experts and the EM Algorithm. *Neural Computation*, 6(2):181–214, 1994.

[Jua92] B.H. Juang and S. Katagiri. Discriminative Learning for Minimum Error Classification. *IEEE Trans. on Signal Processing*, 40(12):3043–3054, 1992.

[Kah93] B. Kahles. Detektion von Laryngalisierungen mittels Neuronaler Netze im invers gefilterten Sprachsignal. Bachelor's Thesis, Lehrstuhl für Mustererkennung (Informatik 5), Universität Erlangen–Nürnberg, 1993.

[Kah94] B. Kahles. Integrierte Erkennung von Phrasengrenzen und Phrasenakzenten mit einem NN/HMM–Hybrid. Diploma Thesis, Lehrstuhl für Mustererkennung (Informatik 5), Universität Erlangen–Nürnberg, 1994.

[Kam93] H. Kamp and U. Reyle. *From Discourse to Logic and DRT; An Intorduction to Modeltheoretic Semantics of Natural Language.* Kluwer, Dordrecht, 1993.

[Ken91] P. Kenny, S. Parthasarathy, V.N. Gupta, M. Lenning, P. Mermelstein, and D. O'Shaughnessy. Energy, Duration and Markov Models. In *Proc. European Conf. on Speech Communication and Technology*, volume 2, pages 655–658, Genova, 1991.

[Ken94] P. Kenny, G. Boulianne, H. Garudadri, S. Trudelle, R. Hollan, M. Lennig, and D. O'Shaughnessy. Experiments in Continuous Speech Recognition Using Books on Tape. *Speech Communication*, 14(1):49–60, 1994.

[Kh95] Kyung-ho, Loken-Kim, Young-duk, Park, Suguru, and Mizunashi.

Verbal–gestural Behaviours in Multimodal Spoken Language Interpreting Telecommunications. In *Proc. European Conf. on Speech Communication and Technology*, volume 1, pages 281–284, Madrid, Spain, 1995.

[Kie90] A. Kießling. Optimierung des DPGF-Grundfrequenzverfahrens durch besondere Berücksichtigung irregulärer Signalbereiche. Diploma Thesis, Lehrstuhl für Mustererkennung (Informatik 5), Universität Erlangen–Nürnberg, 1990.

[Kie92] A. Kießling, R. Kompe, H. Niemann, E. Nöth, and A. Batliner. DP-Based Determination of *F0* contours from speech signals. In *Proc. Int. Conf. on Acoustics, Speech and Signal Processing*, volume 2, pages II–17–II–20, San Francisco, CA, 1992.

[Kie93] A. Kießling, R. Kompe, H. Niemann, E. Nöth, and A. Batliner. "Roger", "Sorry", "I'm still listening": Dialog guiding signals in information retrieval dialogs. In D. House and P. Touati, editors, *Proc. of an ESCA Workshop on Prosody*, pages 140–143. Lund University, Department of Linguistics, Lund, 1993.

[Kie94a] A. Kießling, R. Kompe, A. Batliner, H. Niemann, and E. Nöth. Automatic Labeling of Phrase Accents in German. In *Int. Conf. on Spoken Language Processing*, volume 1, pages 115–118, Yokohama, 1994.

[Kie94b] A. Kießling, R. Kompe, H. Niemann, E. Nöth, and A. Batliner. Detection of Phrase Boundaries and Accents. In H. Niemann, R. De Mori, and G. Hanrieder, editors, *Progress and Prospects of Speech Research and Technology: Proc. of the CRIM / FORWISS Workshop*, PAI 1, pages 266–269. Infix, 1994.

[Kie95] A. Kießling, R. Kompe, H. Niemann, E. Nöth, and A. Batliner. Voice Source State as a Source of Information in Speech Recognition: Detection of Laryngealizations. In A.J. Rubio Ayuso and J.M. López Soler, editors, *Speech Recognition and Coding. New Advances and Trends*, volume 147 of *NATO ASI Series F*, pages 329–332. Springer, Berlin, 1995.

[Kie96b] A. Kiessling. Personal communication, March 1996.

[Kie96c] A. Kießling, R. Kompe, A. Batliner, H. Niemann, and E. Nöth. Classification of Boundaries and Accents in Spontaneous Speech. In R. Kuhn, editor, *Proc. of the CRIM / FORWISS Workshop*, pages 104–113, Montreal, 1996.

[Kie97] A. Kießling. *Extraktion und Klassifikation prosodischer Merkmale in der automatischen Sprachverarbeitung.* Dissertation, Shaker Verlag, Aachen 1997.

[Kil93] U. Kilian. Personal communication, 1993.

[Kip66] P. Kiparsky. *Über den Deutschen Akzent.* Akademie–Verlag, Berlin, 1966.

[Kis95] T. Kiss. *Merkmale und Repraesentationen*. Westdeutscher Verlag, Opladen, 1995.

[Kla90] D.H. Klatt and L.C. Klatt. Analysis, Synthesis, and Perception of Voice Quality Variations among Female and Male Talkers. *Journal of the Acoustic Society of America*, 87(2):820–857, 1990.

[Kle93] Wolfgang Klein. Ellipse. In J. Jacobs, A. v. Stechow, W. Sternefeld, and T. Vennemann, editors, *Syntax – Ein Internationales Handbuch Zeitgenössischer Forschung – An International Handbook of Contemporary Research*, volume 1, pages 763–799. Walter de Gruyter, Berlin, 1993.

[Koh77] K.J. Kohler. *Einführung in die Phonetik des Deutschen*. Erich Schmidt Verlag, Berlin, 1977.

[Koh90] T. Kohonen. The Self–Organizing Map. *Proceedings of the IEEE*, 78(9):1464–1480, 1990.

[Koh94] K. Kohler, G. Lex, M. Pätzold, M. Scheffers, A. Simpson, and W. Thon. Handbuch zur Datenaufnahme und Transliteration in TP14 von Verbmobil, V3.0. Verbmobil Technisches–Dokument 11, Institut für Phonetik und digitale Sprachverarbeitung, Universität Kiel, Kiel, 1994.

[Koh95] K.J. Kohler. Articulatory Reduction in Different Speaking Styles. In *Proc. of the 13th Int. Congress of Phonetic Sciences*, volume 2, pages 12–19, Stockholm, 1995.

[Kom89] R. Kompe. Ein Mehrkanalverfahren zur Berechnung der Grundfrequenzkontur unter Einsatz der Dynamischen Programmierung. Diploma Thesis, Lehrstuhl für Mustererkennung (Informatik 5), Universität Erlangen–Nürnberg, 1989.

[Kom93a] R. Kompe, A. Batliner, A. Kießling, E. Nöth, and H. Niemann. Can You Tell Apart Spontaneous and Read Speech if You Just Look at Prosody? In *Proc. NATO ASI Conference "New Advances and Trends in Speech Recognition and Coding"*, volume 2, pages 101–104, Bubion, 1993.

[Kom93b] R. Kompe, A. Kießling, T. Kuhn, M. Mast, H. Niemann, E. Nöth, K. Ott, and A. Batliner. Prosody takes over: A prosodically guided dialog system. In *Proc. European Conf. on Speech Communication and Technology*, volume 3, pages 2003–2006, Berlin, 1993.

[Kom94a] R. Kompe, A. Batliner, A. Kießling, U. Kilian, H. Niemann, E. Nöth, and P. Regel-Brietzmann. Automatic Classification of Prosodically Marked Phrase Boundaries in German. In *Proc. Int. Conf. on Acoustics, Speech and Signal Processing*, volume 2, pages 173–176, Adelaide, 1994.

[Kom94b] R. Kompe, E. Nöth, A. Kießling, T. Kuhn, M. Mast, H. Niemann, K. Ott, and A. Batliner. Prosody takes over: Towards a prosodically guided dialog system. *Speech Communication*, 15(1–2):155–167, 1994.

[Kom95a] R. Kompe, W. Eckert, A. Kießling, T. Kuhn, M. Mast, H. Niemann, E. Nöth, E.G. Schukat-Talamazzini, and S. Rieck. Towards Domain–independent Understanding of Spontaneous Speech. In *Proc. European Congress on Intelligent Techniques and Soft Computing*, volume 3, pages 2315–2319, Aachen, 1995.

[Kom95b] R. Kompe, A. Kießling, H. Niemann, E. Nöth, E.G. Schukat-Talamazzini, A. Zottmann, and A. Batliner. Prosodic Scoring of Word Hypotheses Graphs. In *Proc. European Conf. on Speech Communication and Technology*, volume 2, pages 1333–1336, Madrid, 1995.

[Kom97] R. Kompe, A. Kießling, E. Nöth, H. Niemann, A. Batliner, S. Schachtl, T. Ruland, and U. Block. Improving Parsing of Spontaneous Speech with the Help of Prosodic Boundaries. In *Proc. Int. Conf. on Acoustics, Speech and Signal Processing*, München, 1997.

[Kra90] J. Krause, L. Hitzenberger, S. Krischker, H. Kritzenberger, B. Mielke, and C. Womser-Hador. Endbericht zum BMFT-Projekt "Sprachverstehende Systeme; Teilprojekt Simulation einer multimedialen Dialog–Benutzer Schnittstelle - DICOS". FG Linguistische Informationswissenschaft Universität Regensburg, 1990.

[Kuh90] R. Kuhn and R. De Mori. A Cache–Based Natural Language Model for Speech Recognition. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 12(6):570–583, 1990.

[Kuh92] T. Kuhn, H. Niemann, E.G. Schukat-Talamazzini, W. Eckert, and S. Rieck. Context–Dependent Modeling in a Two–Stage HMM Word Recognizer for Continuous Speech. In J. Vandewalle, R. Boite, M. Moonen, and A. Oosterlinck, editors, *Signal Processing VI: Theories and Applications*, volume 1, pages 439–442. Elsevier Science Publishers, Amsterdam, 1992.

[Kuh93] R. Kuhn. *Keyword Classification Trees for Speech Understanding Systems*. PhD thesis, School of Computer Science, McGill University, Montreal, 1993.

[Kuh94] T. Kuhn, H. Niemann, and E.G. Schukat-Talamazzini. Ergodic Hidden Markov Models and Polygrams for Language Modeling. In *Proc. Int. Conf. on Acoustics, Speech and Signal Processing*, volume 1, pages 357–360, Adelaide, Australia, 1994.

[Kuh95a] R. Kuhn and R. De Mori. The Application of Semantic Classification Trees to Natural Language Understanding. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 17:449–460, 1995.

[Kuh95b] T. Kuhn. *Die Erkennungsphase in einem Dialogsystem*, volume 80 of *Dissertationen zur Künstlichen Intelligenz*. Infix, St. Augustin, 1995.

[Kuh96] T. Kuhn, P. Fetter, A. Kaltenmeier, and P. Regel-Brietzmann. DP–

based Wordgraph Pruning. In *Proc. Int. Conf. on Acoustics, Speech and Signal Processing*, Atlanta, 1996.

[Kum92] F. Kummert. *Flexible Steuerung eines sprachverstehenden Systems mit homogener Wissensbasis*, volume 12 of *Dissertationen zur Künstlichen Intelligenz*. Infix, St. Augustin, 1992.

[Kun90] S. Kunzmann. *Die Worterkennung in einem Dialogsystem für kontinuierlich gesprochene Sprache*. Niemeyer, Tübingen, 1990.

[Lad82] P. Ladefoged. *A Course in Phonetics, Second Edition*. Hartcourt Brace Jovanovich, New York, 1982.

[Lad86] D.R. Ladd. Intonational Phrasing: The Case for Recursive Prosodic Structure. *Phonological Yearbook*, 3:311–340, 1986.

[Lad92] P. Ladefoged. Knowing Enough to Analyze Spoken Languages. In *Int. Conf. on Spoken Language Processing*, volume 1, pages 1–4, Banff, 1992.

[Lap87] A. Lapedes and R. Farber. Nonlinear Signal Processing Using Neural Networks: Prediction and System Modeling. LA–UR–87–2665, Los Alamos Laboratory, Los Alamos, 1987.

[Lav80] J. Laver. *The Phonetic Description of Voice Quality*. Cambridge University Press, Cambridge, MA, 1980.

[Laz94] G. Lazzari. Automatic Speech Recognition and Understanding at IRST. In H. Niemann, R. De Mori, and G. Hanrieder, editors, *Progress and Prospects of Speech Research and Technology: Proc. of the CRIM / FORWISS Workshop*, PAI 1, pages 149–157. Infix, 1994.

[Lea75] W.A. Lea, M.F. Medress, and T.E. Skinner. A prosodically guided speech understanding strategy. *IEEE Trans.*, ASSP-23:30–38, 1975.

[Lea80a] W. Lea. Prosodic Aids to Speech Recognition. In W. Lea, editor, *Trends in Speech Recognition*, pages 166–205. Prentice–Hall Inc., Englewood Cliffs, New Jersey, 1980.

[Lea80b] Wayne A. Lea, editor. *Trends in Speech Recognition*, Englewood Cliffs, NJ, 1980. Prentice Hall.

[Lee89] K.-F. Lee. *Automatic Speech Recognition: the Development of the SPHINX System*. Kluwer Academic Publishers, Boston, MA, 1989.

[Lee95] C.-H. Lee and J.-L. Gauvain. Adaptive Learning in Acoustic and Language Modeling. In A.J. Rubio Ayuso and J.M. López Soler, editors, *Speech Recognition and Coding. New Advances and Trends*, volume 147 of *NATO ASI Series F*, pages 14–31. Springer, Berlin, 1995.

[Leh59] I. Lehiste and G. Peterson. Vowel amplitude and phonemic stress in american english. *Journal of the Acoustic Society of America*, 31:428–435, 1959.

[Leh70] I. Lehiste. *Suprasegmentals*. MIT Press, Cambridge, MA, 1970.

[Leh95] M. Lehning. Statistical Methods for the Automatic Labelling of German Prosody. In *Proc. European Conf. on Speech Communication and*

Technology, volume 3, pages 2089–2092, Madrid, Spain, 1995.

[Len90] M. Lennig. Putting Speech Recognition to Work in the Telephone Network. *Computer*, 9(1):35–41, 1990.

[Leu88] H.C. Leung and V.W. Zue. Some Phonetic Recognition Experiments using Artificial Neural Nets of Neural Network Classifiers. In *Proc. Int. Conf. on Acoustics, Speech and Signal Processing*, volume 1, pages 422–425, New York, 1988.

[Leu91] H.C. Leung, I.L. Hetherington, and V.W. Zue. Speech Recognition Using Stochastic Explicit–Segment Modeling. In *Proc. European Conf. on Speech Communication and Technology*, volume 2, pages 931–934, Genova, 1991.

[Lev66] V.I. Levenshtein. Binary Codes Capable of Correcting Deletions Insertions and Reversals. *Cybernetics and Control Theory*, 10:707–710, 1966.

[Lev86] S.E. Levinson. Continuously Variable Duration Hidden Markov Models for Automatic Speech Recognition. *Computer Speech & Language*, 1(1):29–45, 1986.

[Lev89] S.E. Levinson, M.Y. Liberman, A. Ljolje, and L.G. Miller. Speaker Independent Phonetic Transcription of Fluent Speech for Large Vocabulary Speech Recognition. In *Int. Conf. on Acoustics, Speech and Signal Processing*, volume 2, pages 441–443, Glasgow, 1989.

[Lic92] R.J. Lickley and E.G. Bard. Processing Disfluent Speech: Recognising Disfluency Before Lexical Access. In *Int. Conf. on Spoken Language Processing*, volume 2, pages 935–938, Banff, 1992.

[Lin80] Y. Linde, A. Buzo, and R.M. Gray. An Algorithm for Vector Quantizer Design. *IEEE Trans. on Communications*, 28(1):84–95, 1980.

[Lip87] R.P. Lippmann. An Introduction to Computing with Neural Nets. *IEEE ASSP Magazine*, 4:4–22, 1987.

[Lip90] R.P. Lippmann. Review of neural networks for speech recognition. In A. Waibel and K.-F. Lee, editors, *Readings in Speech Recognition*, pages 374–392. Morgan Kaufmann Publishers Inc., San Mateo, California, 1990.

[Ljo94] A. Ljolje. High Accuracy Phone Recognition Using Context Clustering and Quasi–triphonic Models. *Computer Speech & Language*, 8(2):129–152, 1994.

[Ljo95] A. Ljolje, M. Riley, D. Hindle, and F. Pereira. The AT&T 60,000 Word Speech–To–Text System. In *Proc. of the ARPA Spoken Language Systems Technology Workshop*, pages 162–165, San Francisco, CA, 1995. Morgan Kaufman.

[Mag94] David M. Magerman. *Natural Language Parsing as Statistical Pattern Recognition*. PhD thesis, Stanford University, 1994.

[Man92] S.Y. Manuel, S. Shattuck-Hufnagel, M. Huffman, K.N. Stevens,

R. Carlson, and S. Hunnicutt. Studies of Vowel and Consonant Reduction. In *Int. Conf. on Spoken Language Processing*, volume 2, pages 943–946, Banff, 1992.

[Mar72] J.D. Markel. The SIFT Algorithm for Fundamental Frequency Estimation. *IEEE Trans. on Audio and Electroacoustics*, AU-20(5):367–377, 1972.

[Mas92] M. Mast, R. Kompe, F. Kummert, H. Niemann, and E. Nöth. The Dialog Module of the Speech Recognition and Dialog System EVAR. In *Int. Conf. on Spoken Language Processing*, volume 2, pages 1573–1576, Banff, 1992.

[Mas93] M. Mast. *Ein Dialogmodul für ein Spracherkennungs- und Dialogsystem*, volume 50 of *Dissertationen zur künstlichen Intelligenz*. Infix, St. Augustin, 1993.

[Mas94a] M. Mast, F. Kummert, U. Ehrlich, G. Fink, T. Kuhn, H. Niemann, and G. Sagerer. A Speech Understanding and Dialog System with a Homogeneous Linguistic Knowledge Base. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 16(2):179–194, 1994.

[Mas94b] E. Mastenbroek and S. McGlashan. The Verbmobil Syntax Semantics Interface — Version 1.2. Verbmobil Memo 41, IAI, Saarbrücken, 1994.

[Mas95a] M. Mast. Schlüsselwörter zur Detektion von Diskontinuitäten und Sprechhandlungen. Technical report, Lehrstuhl für Mustererkennung, Universität Erlangen–Nürnberg, 1995.

[Mas95b] M. Mast, E. Maier, and B. Schmitz. Criteria for the Segmentation of Spoken Input into Individual Utterances. Verbmobil Report 97, 1995.

[Mas95c] M. Mast, E. Nöth, H. Niemann, and E.G. Schukat-Talamazzini. Automatic Classification of Speech Acts with Semantic Classification Trees and Polygrams. In *International Joint Conference on Artificial Intelligence 95, Workshop "New Approaches to Learning for Natural Language Processing"*, pages 71–78, Montreal, 1995.

[Mas96] M. Mast, R. Kompe, St. Harbeck, A. Kießling, H. Niemann, and E. Nöth. Dialog Act Classification with the Help of Prosody. In *Int. Conf. on Spoken Language Processing*, volume 3, pages 1728–1731, Philadelphia, 1996.

[McA91] J. McAllister. The Perception of Lexically Stressed Syllables in Read and Spontaneous Speech. *Language and Speech*, 34(1):1–26, 1991.

[Mid60] D. Middelton. *An Introduction to Statistical Communication Theory*. McGraw Hill, New York, 1960.

[Moo94] R. Moore. Twenty Things we Still Don't Know about Speech. In H. Niemann, R. De Mori, and G. Hanrieder, editors, *Progress and Prospects of Speech Research and Technology: Proc. of the CRIM / FORWISS Workshop*, PAI 1, pages 9–17. Infix, 1994.

[Mor95] N. Morgan and H. Bourlard. Continuous Speech Recognition. *IEEE*

Signal Processing Magazine, 12(5):25–42, 1995.

[Mur93] H. Murveit, J. Butzberger, V. Digalakis, and M. Weintraub. Large–vocabulary Dictation using SRI's DECIPHER Speech Recognition System: Progressive Search Techniques. In *Proc. Int. Conf. on Acoustics, Speech and Signal Processing*, volume 2, pages 319–322, Minneapolis, MN, 1993.

[Nak92] S. Nakajima and J. Allen. Prosody as a cue for discourse structure. In *Int. Conf. on Spoken Language Processing*, pages 425–428, Banff, 1992.

[Nak93] C. Nakatani. Accenting on Pronouns and Proper Names in Spontaneous Narrative. In D. House and P. Touati, editors, *Proc. of an ESCA Workshop on Prosody*, pages 164–167. Lund University, Department of Linguistics, Lund, 1993.

[Ney93] H. Ney. Modeling and Search in Continuous Speech Recognition. In *Proc. European Conf. on Speech Communication and Technology*, volume 1, pages 491–498, Berlin, Germany, 1993.

[Ney94a] H. Ney, U. Essen, and R. Kneser. On Structuring Probabilistic Dependences on Stochastic Language Modelling. *Computer Speech & Language*, 8(1):1–38, 1994.

[Ney94b] H. Ney, V. Steinbiß, X. Aubert, and R. Haeb-Umbach. Progress in Large Vocabulary, Continuous Speech Recognition. In H. Niemann, R. De Mori, and G. Hanrieder, editors, *Progress and Prospects of Speech Research and Technology: Proc. of the CRIM / FORWISS Workshop*, PAI 1, pages 75–92. Infix, 1994.

[Nie83] H. Niemann. *Klassifikation von Mustern*. Springer–Verlag, Berlin, 1983.

[Nie85] H. Niemann, A. Brietzmann, R. Mühlfeld, P. Regel, and G. Schukat. The Speech Understanding and Dialog System EVAR. In R. De Mori and C.Y. Suen, editors, *New Systems and Architectures for Automatic Speech Recognition and Synthesis*, NATO ASI Series, pages 271–302. Springer–Verlag, 1985.

[Nie86] H. Niemann, A. Brietzmann, U. Ehrlich, and G. Sagerer. Representation of a Continuous Speech Understanding and Dialog System in a Homogeneous Semantic Net Architecture. In *Proc. Int. Conf. on Acoustics, Speech and Signal Processing*, pages 1581–1584, Tokyo, 1986.

[Nie88] H. Niemann, A. Brietzmann, U. Ehrlich, S. Posch, P. Regel, G. Sagerer, R. Salzbrunn, and G. Schukat-Talamazzini. A Knowledge Based Speech Understanding System. *Int. Journal of Pattern Recognition and Artificial Intelligence*, 2(2):321–350, 1988.

[Nie90a] H. Niemann. *Pattern Analysis and Understanding*, volume 4 of *Series in Information Sciences*. Springer–Verlag, Heidelberg, 1990.

[Nie90b] H. Niemann, G. Sagerer, S. Schröder, and F. Kummert. ERNEST: A Semantic Network System for Pattern Analysis. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 9:883–905, 1990.

[Nie92] H. Niemann, G. Sagerer, U. Ehrlich, E.G. Schukat-Talamazzini, and F. Kummert. The Interaction of Word Recognition and Linguistic Processing in Speech Understanding. In P. Laface and R. De Mori, editors, *Speech Recognition and Understanding. Recent Advances, Trends, and Applications*, NATO ASI Series F75, pages 425–453. Springer–Verlag, 1992.

[Nie93] H. Niemann, E. Nöth, E.G. Schukat-Talamazzini, A. Kießling, R. Kompe, T. Kuhn, K. Ott, and S. Rieck. Statistical Modeling of Segmental and Suprasegmental Information. In *Proc. NATO ASI Conference "New Advances and Trends in Speech Recognition and Coding"*, volume 1, pages 237–260, Bubion, 1993.

[Nie94a] H. Niemann, J. Denzler, B. Kahles, R. Kompe, A. Kießling, E. Nöth, and V. Strom. Pitch Determination Considering Laryngealization Effects In Spoken Dialogs. In *IEEE Int. Conf. on Neural Networks*, volume 7, pages 4457–4461, Orlando, 1994.

[Nie94b] H. Niemann, W. Eckert, A. Kießling, R. Kompe, Th. Kuhn, E. Nöth, M. Mast, S. Rieck, E.G. Schukat-Talamazzini, and A. Batliner. Prosodic Dialog Control in EVAR. In H. Niemann, R. De Mori, and G. Hanrieder, editors, *Progress and Prospects of Speech Research and Technology: Proc. of the CRIM / FORWISS Workshop*, PAI 1, pages 166–177. Infix, 1994.

[Nie94c] H. Niemann, E. Nöth, E.G. Schukat-Talamazzini, A. Kießling, R. Kompe, T. Kuhn, and S. Rieck. Phonetic and prosodic analysis of speech. In Bogomir Horvat and Zdravko Kačič, editors, *Modern Modes of Man–Machine Communication*, pages 12–1–12–12. University of Maribor, Maribor, Slovenia, 1994.

[Nie95] H. Niemann, E. Nöth, E.G. Schukat-Talamazzini, A. Kießling, R. Kompe, T. Kuhn, K. Ott, and S. Rieck. Statistical Modeling of Segmental and Suprasegmental Information. In A.J. Rubio Ayuso and J.M. López Soler, editors, *Speech Recognition and Coding. New Advances and Trends*, volume 147 of *NATO ASI Series F*, pages 192–209. Springer, Berlin, 1995.

[Nie97] H. Niemann, E.Nöth, A. Kießling, R. Kompe, and A. Batliner. Prosodic Processing and its Use in Verbmobil. In *Proc. Int. Conf. on Acoustics, Speech and Signal Processing*, München, 1997.

[Nil80] N.J. Nilsson. *Principles of Artificial Intelligence*. Morgan Kaufmann, Los Altos, 1980.

[Nol87] A. Noll and H. Ney. Training of Phoneme Models in a Sentence Recognition System. In *Proc. Int. Conf. on Acoustics, Speech and Signal*

Processing, volume 2, pages 1277–1280, Dallas, 1987.

[Nor91] Y. Normandin. *Hidden Markov Models, Maximum Mutual Information Training, and the Speech Recognition Problem.* PhD thesis, Department of Electrical Engineering, McGill University, Montreal, 1991.

[Nor94a] Y. Normandin, R. Kuhn, R. Cardin, R. Lacouture, and A. Lazarides. Recent Developments in Large Vocabulary Continuous Speech Recognition at CRIM. In H. Niemann, R. De Mori, and G. Hanrieder, editors, *Progress and Prospects of Speech Research and Technology: Proc. of the CRIM / FORWISS Workshop*, PAI 1, pages 121–130. Infix, 1994.

[Nor94b] Y. Normandin, R. Lacouture, and R. Cardin. MMIE Training for Large Vocabulary Continuous Speech Recognition. In *Int. Conf. on Spoken Language Processing*, volume 3, pages 1367–1370, Yokohama, Japan, 1994.

[Nor94c] G. Normann and A. Hildemann. Intuitive Elterliche Sprachförderung im 1. und 2. Lebensjahr und deren Relevanz für die Arbeit mit hörgeschädigten Säuglingen und Kleinkindern. In M. Gross, editor, *Aktuelle Phoniatrisch–Pädaudiologische Aspekte*, pages 63–64. R. Gross Verlag, Berlin, 1994.

[Nor95] G. Normann. Personal communication, June 1995.

[Nöt88a] E. Nöth and R. Kompe. Der Einsatz prosodischer Information im Spracherkennungssystem EVAR. In H. Bunke, O. Kübler, and P. Stucki, editors, *Mustererkennung 1988 (10. DAGM Symposium)*, volume 180 of *Informatik FB*, pages 2–9. Springer–Verlag, Berlin, 1988.

[Nöt88b] E. Nöth, H. Niemann, and S. Schmölz. Prosodic Features in German Speech: Stress Assignment by Man and Machine. In H. Niemann, M. Lang, and G. Sagerer, editors, *Recent Advances in Speech Understanding and Dialog Systems*, volume 46 of *NATO ASI Series F*, pages 101–106. Springer–Verlag, Berlin, 1988.

[Nöt89] E. Nöth and R. Kompe. Verbesserung der Worterkennung mit prosodischer Information. In *Fortschritte der Akustik — Proc. DAGA '89*, pages 343–346, Duisburg, 1989.

[Nöt91a] E. Nöth. *Prosodische Information in der automatischen Spracherkennung — Berechnung und Anwendung*. Niemeyer, Tübingen, 1991.

[Nöt91b] E. Nöth, A. Batliner, T. Kuhn, and G. Stallwitz. Intensity as a Predictor of Focal Accent. In *Proc. of the 12th Int. Congress of Phonetic Sciences*, volume 3, pages 230–233, Aix–en–Provence, 1991. Université de Provence.

[Nöt94a] E. Nöth. "Ja zur Not geht's auch am Samstag" oder — Wie die Prosodie der Linguistik die Satzzeichen verschaffen kann. In *Proc. 18. Deutsche Kahrestagung Künstliche Intelligenz*, volume KI–94 Workshops, pages 186–187, Saarbrücken, 1994.

[Nöt94b] E. Nöth and B. Plannerer. Schnittstellendefinition für den Worthypothesengraphen. Verbmobil Memo 2, 1994.

[Nöt96a] E. Nöth, R. De Mori, J. Fischer, A. Gebhard, S. Harbeck, R. Kompe, R. Kuhn, H. Niemann, and M. Mast. An Integrated Model of Acoustics and Language Using Semantic Classification Trees. In *Proc. Int. Conf. on Acoustics, Speech and Signal Processing*, volume 1, pages 419–422, Atlanta, 1996.

[Nöt96b] E. Nöth, R. Kompe, A. Kießling, H. Niemann, A. Batliner, S. Schachtl, T. Ruland, and U. Block. Prosodic Parsing of Spontaneous Speech. In J.-P. Haton, editor, *Proc. of the Workshop on Multi–lingual Spontaneous Speech Recognition in Real Environments*, Nancy, 1996. CRIN/CNRS–INRIA.

[Och95] F.J. Och. Maximum–Likelihood–Schätzung grammatischer Wortkategorien mit Verfahren der kombinatorischen Optimierung. Diploma Thesis, Lehrstuhl für Mustererkennung (Informatik 5), Universität Erlangen–Nürnberg, 1995.

[O'S83] D. O'Shaughnessy and J. Allen. Linguistic Modality Effects on Fundamental Frequency in Speech. *Journal of the Acoustic Society of America*, 74(4):1155–1171, 1983.

[O'S87] D. O'Shaughnessy. *Speech Communication.* Addison Wesley, Reading, MA, 1987.

[O'S92a] D. O'Shaughnessy. Analysis of False Starts in Spontaneous Speech. In *Int. Conf. on Spoken Language Processing*, volume 2, pages 931–934, Banff, 1992.

[O'S92b] D. O'Shaughnessy. Recognition of Hesitations in Spontaneous Speech. In *Proc. Int. Conf. on Acoustics, Speech and Signal Processing*, volume 1, pages 521–524, San Francisco, CA, 1992.

[O'S93] D. O'Shaughnessy. Locating Disfluencies in Spontaneous Speech: An Acoustic Analysis. In *Proc. European Conf. on Speech Communication and Technology*, volume 3, pages 2187–2190, Berlin, 1993.

[O'S95] D. O'Shaughnessy. Timing Patterns in Fluent and Disfluent Spontaneous Speech. In *Proc. Int. Conf. on Acoustics, Speech and Signal Processing*, volume 1, pages 600–603, Detroit, 1995.

[Ost89] M. Ostendorf and S. Roukos. A Stochastic Segment Model for Phoneme–based Continuous Speech Recognition. *IEEE Trans. on Acoustics, Speech and Signal Processing*, 37(12):1857–1869, 1989.

[Ost90] M. Ostendorf, P.J. Price, J. Bear, and C.W. Wightman. The Use of Relative Duration in Syntactic Disambiguation. In *Speech and Natural Language Workshop*, pages 26–31. Morgan Kaufmann, Hidden Valley, Pennsylvania, 1990.

[Ost93a] M. Ostendorf, P. Price, and S. Shattuck-Hufnagel. Combining Statistical and Linguistic Methods for Modeling Prosody. In D. House and

P. Touati, editors, *Proc. of an ESCA Workshop on Prosody*, pages 272–275. Lund University, Department of Linguistics, Lund, 1993.

[Ost93b] M. Ostendorf, C.W. Wightman, and N.M. Veilleux. Parse Scoring with Prosodic Information: an Analysis/Synthesis approach. *Computer Speech & Language*, 7(3):193–210, 1993.

[Ost94] M. Ostendorf and N.M. Veilleux. A Hierarchical Stochastic Model for Automatic Prediction of Prosodic Boundary Location. *Computational Linguistics*, 20(1):27–53, 1994.

[Ott93] K. Ott. Prosodisch basierte Dialogsteuerung in EVAR. Diploma Thesis, Lehrstuhl für Mustererkennung (Informatik 5), Universität Erlangen–Nürnberg, 1993.

[Pap81] M. Papoušek and H. Papoušek. Musical Elements in the Infant's Vocalization: Their Significance for Communication, Cognition and Creativity. In L.P. Lipsitt, editor, *Advances in Infancy Research*, volume 1, pages 163–224. Ablex, Norwood, 1981.

[Pat95] C. Pateras, G. Dudek, and R. De Mori. Understanding Referring Expressions in a Person–Machine Spoken Dialogue. In *Proc. Int. Conf. on Acoustics, Speech and Signal Processing*, volume 1, pages 197–200, Detroit, 1995.

[Pau72] E. Paulus and E. Zwicker. Programme zur automatischen Bestimmung der Lautheit aus Terzpegeln oder Frequenzgruppenpegeln. *Acustica*, 27(5):253–266, 1972.

[Pau90] D.B. Paul. Algorithms for an Optimal A* Search and Linearizing the Search in the Stack Decoder. In *Speech and Natural Language Workshop*, pages 200–204. Morgan Kaufmann, Hidden Valley, Pennsylvania, 1990.

[Pau92] D. Paul and J. Baker. The Design for the Wall Street Journal–based CSR Corpus. In *Proc. Speech and Natural Language Workshop*, pages 1–5, San Mateo, California, 1992. Morgan Kaufman.

[Pau94] Erwin Paulus and Michael Lehning. Die Evaluierung von Spracherkennungssystemen in Deutschland. In *Fortschritte der Akustik — Proc. DAGA '94*, volume A, pages 147–156, Dresden, 1994.

[Pea84] J. Pearl. Some Recent Results in Heuristic Search Theory. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 6(1):1–13, 1984.

[Pec91] J. Peckham. Speech Understanding and Dialogue over the Telephone: an Overview of Progress in the SUNDIAL Project. In *Proc. European Conf. on Speech Communication and Technology*, volume 3, pages 1469–1472, Genova, 1991.

[Pec94] J. Peckham and N.M. Fraser. Spoken Language Dialog over the Telephone. In H. Niemann, R. De Mori, and G. Hanrieder, editors, *Progress and Prospects of Speech Research and Technology: Proc. of the CRIM / FORWISS Workshop*, PAI 1, pages 192–203. Infix, 1994.

[Pic86] J. Picone, K.M. Goudie-Marshall, G.R. Doddington, and W. Fisher. Automatic Text Alignment for Speech System Evaluation. *IEEE Trans. on Acoustics, Speech and Signal Processing*, 34(4):780–784, 1986.

[Pie80] J. Pierrehumbert. *The Phonology and Phonetics of English Intonation*. PhD thesis, MIT, Cambridge, MA, 1980.

[Pie90] J. Pierrehumert and J. Hirschberg. The Meaning of Intonation Contours in the Interpretation of Discourse. In P.R. Cohen, J. Morgan, and M. Pollack, editors, *Plans and Intentions in Communication and Discourse*. MIT press, Cambridge, MA, 1990.

[Pie95] R. Pieraccini and E. Levin. A Learning Approach to Natural Language Understanding. In A.J. Rubio Ayuso and J.M. López Soler, editors, *Speech Recognition and Coding. New Advances and Trends*, volume 147 of *NATO ASI Series F*, pages 139–156. Springer, Berlin, 1995.

[Pol87] C. Pollard and I. Sag. *Information–based Syntax and Semantics, Vol. 1*, volume 13 of *CSLI Lecture Notes*. CSLI, Stanford, CA, 1987.

[Pol91] J. Polifroni, S. Seneff, and V.W. Zue. Collection of Spontaneous Speech for the ATIS Domain and Comparative Analyses of Data Collected at MIT and TI. In *Proc. Speech and Natural Language Workshop*, San Mateo, California, 1991. Morgan Kaufman.

[Pol94] L.C.W. Pols. Personal communication, September 1994.

[Por94] T. Portele, F. Höfer, and W. Hess. Structure and Representation of an Inventory for German Speech Synthesis. In *Int. Conf. on Spoken Language Processing*, pages 1759–1762, Yokohama, Japan, 1994.

[Pra86] R.W. Prager, T.D. Harrison, and F. Fallside. Boltzmann Machines for Speech Recognition. *Computer Speech & Language*, 1(1):3–27, 1986.

[Pri89] P.J. Price, M. Ostendorf, and Wightman C.W. Prosody and Parsing. In *Speech and Natural Language Workshop*, pages 5–11. Morgan Kaufmann, 1989.

[Pri90] P.J. Price, C.W. Wightman, M. Ostendorf, and J. Bear. The Use of Relative Duration in Syntactic Disambiguation. In *Int. Conf. on Spoken Language Processing*, volume 1, pages 13–18, Kobe, Japan, 1990.

[Pri91] P.J. Price, M. Ostendorf, S. Shattuck-Hufnagel, and C. Fong. The Use of Prosody in Syntactic Disambiguation. *Journal of the Acoustic Society of America*, 90:2956–2970, 1991.

[Pri92] P. Price and J. Hirschberg. Session 13: Prosody – Introduction. In *Speech and Natural Language Workshop*, pages 92–95. Morgan Kaufmann, 1992.

[Qua94] J.J. Quantz and B. Schmitz. Knowledge–based Disambiguation for Machine Translation. *Mind and Machines*, 4:39–57, 1994.

[Qua95] J.J. Quantz, M. Gehrke, and U. Küssner. Domain Modeling for Machine Translation. In *Proc. of 6th International Conference on Theoretical and Methodological Issues in Machine Translation*, Leuven,

1995. Katholieke Universiteit Leuven.

[Rab89] L.R. Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–285, 1989.

[Rab93] L. Rabiner and B.-H. Juang. *Fundamentals of Speech Recognition.* Prentice Hall, New Jersey, 1993.

[Rab95] L. Rabiner. Telecommunications Applications of Speech Processing. In A.J. Rubio Ayuso and J.M. López Soler, editors, *Speech Recognition and Coding. New Advances and Trends*, volume 147 of *NATO ASI Series F*, pages 283–300. Springer, Berlin, 1995.

[Rei94] W. Reichl, P. Caspary, and G. Ruske. A New Model–Discriminant Training Algorithm for Hybrid NN/HMM Systems. In *Proc. Int. Conf. on Acoustics, Speech and Signal Processing*, volume 1, pages 677–680, Adelaide, 1994.

[Rei95] N. Reithinger, E. Maier, and J. Alexandersson. Treatment of Incomplete Dialogues in a Speech–to–speech Translation System. In P. Dalsgaard, L.B. Larsen, L. Boves, and I. Thomsen, editors, *Proc. of the ESCA Tutorial and Research Workshop on Spoken Dialogue Systems*, pages 33–36. ESCA, Vigsø, Denmark, 1995.

[Rey93] M. Reyelt. Experimental Investigation on the Perceptual Consistency and the Automatic Recognition of Prosodic Units in Spoken German. In D. House and P. Touati, editors, *Proc. of an ESCA Workshop on Prosody*, pages 238–241. Lund University, Department of Linguistics, Lund, 1993.

[Rey94] M. Reyelt and A. Batliner. Ein Inventar prosodischer Etiketten für Verbmobil. Verbmobil Memo 33, 1994.

[Rey95a] M. Reyelt. Consistency of Prosodic Transcriptions Labelling Experiments with Trained and Untrained Transcribers. In *Proc. of the 13th Int. Congress of Phonetic Sciences*, volume 4, pages 212–215, Stockholm, 1995.

[Rey95b] M. Reyelt. Ein System zur prosodischen Etikettierung von Spontansprache. In R. Hoffmann and R. Ose, editors, *Elektronische Sprachsignalverarbeitung*, volume 12 of *Studientexte zur Sprachkommunikation*, pages 167–174. TU Dresden, Wolfenbüttel, 1995.

[Ric91] M.D. Richard and R.P. Lippmann. Neural Network Classifiers Estimate Bayesian a posteriori Probabilities. *Neural Computation*, 3(4):461–483, 1991.

[Rie93] S. Rieck. ERBA – A Domain Dependent Speech Data Base. *NESCA - The European Speech Communication Association Newsletter*, 1993.

[Rie94] S. Rieck. *Parametrisierung und Klassifikation gesprochener Sprache.* PhD thesis, Technische Fakultät der Universität Erlangen–Nürnberg, 1994.

[Rie95] S. Rieck. *Parametrisierung und Klassifikation gesprochener Sprache*, volume 10: Informatik/Kommunikationstechnik no. 353 of *Fortschritberichte*. VDI Verlag, Düsseldorf, 1995.

[Ril95] M. Riley, A. Ljolje, D. Hindle, and F. Pereira. The AT&T 60,000 Word Speech–to–text System. In *Proc. European Conf. on Speech Communication and Technology*, volume 1, pages 207–210, Madrid, Spain, 1995.

[Rip96a] B.D. Ripley. *Pattern Recognition and Neural Networks*. Cambridge University Press, Cambridge, 1996.

[Rip96b] B. Ripplinger. Personal communication, July 1996.

[Rip96c] B. Ripplinger and J. Alexandersson. Disambiguation and Translation of German Particles in Verbmobil, Verbmobil Memo 70, 1996.

[Rog95] I. Rogina and A. Waibel. The JANUS Speech Recognizer. In *Proc. of the ARPA Spoken Language Systems Technology Workshop*, pages 166–169, San Francisco, CA, 1995. Morgan Kaufman.

[Roj93] R. Rojas. *Theorie der neuronalen Netze*. Springer, Berlin, 1993.

[Ros74] M.J. Ross, H. Shaffer, A. Cohen, R. Freudberg, and H. Manley. Average magnitude difference function pitch extractor. *IEEE Trans. on Acoustics, Speech and Signal Processing*, ASSP-22(5):353–362, 1974.

[Ros92] K. Ross, M. Ostendorf, and S. Shattuck-Hufnagel. Factors Affecting Pitch Accent Placement. In *Int. Conf. on Spoken Language Processing*, volume 1, pages 365–368, Banff, 1992.

[Rou87] S. Roucos and M.O. Dunham. A Stochastic Segment Model for Phoneme–Based Continuous Speech Recognition. In *Int. Conf. on Acoustics, Speech and Signal Processing*, pages 73–76, Dallas, Texas, 1987.

[Roy83] H.-W. Royé. *Segmentierung und Hervorhebung in gesprochener deutscher Standardsprache*. Niemeyer, Tübingen, 1983.

[Rub96] Jeanne Rubner. Zur Not! oder Zur Not? *Süddeutsche Zeitung*, page XXXXXX, November, 7, 1996.

[Rul96] T. Ruland. Personal communication, May 1996.

[Rum86] D. Rumelhart and J. McClelland, editors. *Parallel Distributed Processing: Exploration in the Microstructure of Cognition, Vol. 1: Foundations*. MIT Press, Cambridge, MA, 1986.

[Sag90] G. Sagerer. *Automatisches Verstehen gesprochener Sprache*, volume 74 of *Reihe Informatik*. Bibliographisches Institut, Mannheim, 1990.

[Sch77] J. Schürmann. *Polynomklassifikatoren für die Zeichenerkennung*. Oldenbourg, München, 1977.

[Sch84] J. Schürmann and W. Doster. A Decision Theoretic Approach to Hierarchical Classifier Design. *Pattern Recognition*, 17(3):359–369, 1984.

[Sch85] R. Schwartz, Y. Chow, O. Kimball, S. Roucos, M. Krasner, and

J. Makhoul. Context–Dependent Modeling for Acoustic–Phonetic Recognition of Continuous Speech. In *Proc. Int. Conf. on Acoustics, Speech and Signal Processing*, pages 1205–1208, Tampa, FL, 1985.

[Sch93] D. Schnelle. Automatische Erkennung von prosodischen Phrasengrenzen. Bachelor's Thesis, Lehrstuhl für Mustererkennung (Informatik 5), Universität Erlangen–Nürnberg, 1993.

[Sch94] L.A. Schmid. Parsing Word Graphs Using a Linguistic Grammar and a Statistical Language Model. In *Proc. Int. Conf. on Acoustics, Speech and Signal Processing*, volume 2, pages 41–44, Adelaide, 1994.

[Sch95a] S. Schachtl. Personal communication, February 1995.

[Sch95b] B. Schmitz and K. Fischer. Pragmatisches Beschreibungsinventar für Diskurspartikeln und Routineformeln anhand der Demonstratorwortliste. Verbmobil Memo 75, Technische Universität Berlin, Berlin, 1995.

[Sch96a] S. Schachtl. Personal communication, February 1996.

[Sch96b] E. Schindler. *The Computer Speech Book*. Academic Press, London, 1996.

[Sen78] S. Seneff. Real–time harmonic pitch detector. *IEEE Trans. on Acoustics, Speech and Signal Processing*, ASSP-26(4):358–365, 1978.

[Sen88] S. Seneff. A Joint Synchrony/Mean Rate Model of Auditory Speech Processing. *J. Phonetics*, 16:55–76, 1988.

[Sen95] S. Seneff, M. McCandless, and V. Zue. Integrating Natural Language into the Word Graph Search for Simultaneous Speech Recognition and Understanding. In *Proc. European Conf. on Speech Communication and Technology*, volume 3, pages 1781–1784, Madrid, Spain, 1995.

[Sen96] S. Seneff. Comments on: "Towards Increasing Speech Recognition Error Rates" by H. Bourlard, H. Hermansky, and N. Morgan. *Speech Communication*, 18:253–255, 1996.

[SH92] S. Shattuck-Hufnagel. Stress Shift as Pitch Accent Placement: Within–word Early Accent Placement in American English. In *Int. Conf. on Spoken Language Processing*, volume 2, pages 747–750, Banff, 1992.

[Shi86] M.S. Shieber. *An Introduction to Unification–based Approaches to Grammar*, volume 4 of *CSLI Lecture Notes*. The University of Chicago Press, Stanford, 1986.

[Sil92a] K. Silverman, M. Beckman, J. Pitrelli, M. Ostendorf, C. Wightman, P. Price, J. Pierrehumbert, and J. Hirschberg. TOBI: A Standard for Labeling English Prosody. In *Int. Conf. on Spoken Language Processing*, volume 2, pages 867–870, Banff, 1992.

[Sil92b] K. Silverman, E. Blaauw, J. Spitz, and J.F. Pitrelli. A Prosodic Comparison of Spontaneous Speech and Read Speech. In *Int. Conf. on Spoken Language Processing*, volume 2, pages 1299–1302, Banff, 1992.

[Sil93] K.E.A. Silverman. On Customizing Prosody in Speech Synthesis:

Names and Address as a Case in Point. In *Human Language Technology — Proc. of the ARPA Workshop*, pages 317–322, Plainsboro, 1993.

[Sin92] H. Singer and S. Sagayama. Pitch Dependent Phone Modelling for HMM Based Speech Recognition. In *Proc. Int. Conf. on Acoustics, Speech and Signal Processing*, volume 1, pages 273–276, San Francisco, CA, 1992.

[Sle91] D. Sleator and D. Temperley. Parsing English with a Link Grammar. Technical Report CMU–CS–91–196, Carnegie Mellon University, School of Computer Science, 1991.

[SPI97] DER SPIEGEL. Mit Weltwissen gefüttert. *DER SPIEGEL*, (5):164–165, February 1997.

[ST92] E.G. Schukat-Talamazzini, H. Niemann, W. Eckert, T. Kuhn, and S. Rieck. Acoustic Modelling of Subword Units in the ISADORA Speech Recognizer. In *Proc. Int. Conf. on Acoustics, Speech and Signal Processing*, volume 1, pages 577–580, San Francisco, CA, 1992.

[ST94] E.G. Schukat-Talamazzini, T. Kuhn, and H. Niemann. Speech Recognition for Spoken Dialogue Systems. In H. Niemann, R. De Mori, and G. Hanrieder, editors, *Progress and Prospects of Speech Research and Technology: Proc. of the CRIM / FORWISS Workshop*, PAI 1, pages 110–120. Infix, 1994.

[ST95a] E. G. Schukat-Talamazzini. *Automatische Spracherkennung – Grundlagen, statistische Modelle und effiziente Algorithmen.* Vieweg, Braunschweig, 1995.

[ST95b] E.G. Schukat-Talamazzini. Stochastic Language Models. In *Electrotechnical and Computer Science Conference*, Portorož, Slovenia, 1995.

[ST95c] E.G. Schukat-Talamazzini, R. Hendrych, R. Kompe, and H. Niemann. Permugram language models. In *Proc. European Conf. on Speech Communication and Technology*, volume 3, pages 1773–1776, Madrid, 1995.

[ST95d] E.G. Schukat-Talamazzini, J. Hornegger, and H. Niemann. Optimal Linear Feature Transformations for Semi–continuous Hidden Markov Models. In *Proc. Int. Conf. on Acoustics, Speech and Signal Processing*, volume 1, pages 369–372, Detroit, 1995.

[Ste89] M. Steedman. Intonation and Syntax in Spoken Language Systems. In *Proc. Speech and Natural Language Workshop*, pages 222–227. DARPA, 1989.

[Ste92] M. Steedman. Grammar, Intonation and Discourse Information. In G. Görz, editor, *KONVENS 92*, Informatik aktuell, pages 21–28. Springer–Verlag, Berlin, 1992.

[Str93] V. Strom. Verbesserung der Grundfrequenzbestimmung durch Optimierung der Stimmhaft/Stimmlos–Entscheidung und besondere Behandlung von Laryngalisierungen, 1993. Memo, Lehrstuhl für Mustererkennung (Informatik 5), Universität Erlangen–Nürnberg.

[Str95] V. Strom. Detection of Accents, Phrase Boundaries and Sentence Modality in German with Prosodic Features. In *Proc. European Conf. on Speech Communication and Technology*, volume 3, pages 2039–2041, Madrid, 1995.

[Str96] V. Strom and C. Widera. What's in the "Pure" Prosody? In *Int. Conf. on Spoken Language Processing*, Philadelphia, 1996.

[Suh95] B. Suhm, P. Geutner, T. Kemp, A. Lavie, L. Mayfield, A.E.McNair, I. Rogina, T. Sloboda, W. Ward, M. Woszczyna, and A. Waibel. Janus: Towards Multilingual Spoken Language Translation. In *Proc. of the ARPA Spoken Language Systems Technology Workshop*, pages 221–226, San Francisco, CA, 1995. Morgan Kaufman.

[Suz95] M. Suzuki, N. Inoue, F. Yato, K. Takeda, and S. Yamamoto. A Prototype of Japanese–Korean Real–time Speech Translation System. In *Proc. European Conf. on Speech Communication and Technology*, volume 3, pages 1951–1954, Madrid, Spain, 1995.

[Swe92] M. Swerts, R. Geluykens, and J. Terken. Prosodic Correlates of Discourse Units in Spontaneous Speech. In *Int. Conf. on Spoken Language Processing*, volume 1, pages 421–424, Banff, 1992.

[Tay95] P. Taylor. Using Neural Networks to Locate Pitch Accents. In *Proc. European Conf. on Speech Communication and Technology*, volume 2, pages 1345–1348, Madrid, 1995.

[tH90] J. 't Hart, R. Collier, and A. Cohen. *A Perceptual Study of Intonation — An Experimental–phonetic Approach to Speech Melody*. Cambridge University Press, Cambridge, MA, 1990.

[Tho96] Thomas Thorwaldsen. Verbmobil macht so manchen Konkurrenten sprachlos. *Computer Zeitung*, (47):28, November, 21 1996.

[Til95] H.G. Tillmann and B. Tischer. Collection and Exploitation of Spontaneous Speech Produced in Negotiation Dialogues. In P. Dalsgaard, L.B. Larsen, L. Boves, and I. Thomsen, editors, *Proc. of the ESCA Tutorial and Research Workshop on Spoken Dialogue Systems*, pages 217–220. ESCA, Vigsø, Denmark, 1995.

[Tis93] Bernd Tischer. *Die vokale Kommunikation von Gefühlen*, volume 18 of *Fortschritte der psychologischen Forschung*. Psychologie Verlags Union, Weinheim, 1993.

[Tom86] M. Tomita. *Efficient Parsing for Natural Language: A Fast Algorithm for Practical Systems*. Kluwer Academic Publishers, 1986.

[Tro94] H. Tropf. Spontansprachliche syntaktische Phänomene: Analyse eines Korpus aus der Domäne "Terminabsprache". Technical report, Siemens AG, ZFE ST SN 54, München, 1994.

[Uhm91] S. Uhmann. *Fokusphonologie. Eine Analyse deutscher Intonationskonturen im Rahmen der nicht–linearen Phonologie.* Niemeyer, Tübingen, 1991.

[Ume94] N. Umeda and T. Wedmore. A Rhythm Theory for Spontaneous Speech: The Role of Vowel Amplitude in the Rhythmic Hierarchy. In *Int. Conf. on Spoken Language Processing*, volume 2, pages 1095–1098, Yokohama, Japan, 1994.

[Vai88] J. Vaissière. The Use of Prosodic Parameters in Automatic Speech Recognition. In H. Niemann, M. Lang, and G. Sagerer, editors, *Recent Advances in Speech Understanding and Dialog Systems*, volume 46 of *NATO ASI Series F*, pages 71–99. Springer–Verlag, Berlin, 1988.

[Vei90] N.M. Veilleux, M. Ostendorf, P.J. Price, and S. Shattuck-Hufnagel. Markov Modeling of Prosodic Phrase Structure. In *Proc. Int. Conf. on Acoustics, Speech and Signal Processing*, volume 2, pages 777–780, Albuquerque, 1990.

[Vei92] N.M. Veilleux, M. Ostendorf, and C.W. Wightman. Parse Scoring with Prosodic Information. In *Int. Conf. on Spoken Language Processing*, volume 3, pages 1605–1608, Banff, 1992.

[Vei93a] N.M. Veilleux and M. Ostendorf. Probabilistic Parse Scoring with Prosodic Information. In *Proc. Int. Conf. on Acoustics, Speech and Signal Processing*, volume 2, pages 51–54, Minneapolis, MN, 1993.

[Vei93b] N.M. Veilleux and M. Ostendorf. Prosody/Parse Scoring and its Application in ATIS. In *Human Language Technology — Proc. of the ARPA Workshop*, pages 335–340, Plainsboro, 1993.

[Vid94] E. Vidal. Language Learning, Understanding and Translation. In H. Niemann, R. De Mori, and G. Hanrieder, editors, *Progress and Prospects of Speech Research and Technology: Proc. of the CRIM / FORWISS Workshop*, PAI 1, pages 131–140. Infix, 1994.

[Vit67] A.J. Viterbi. Error Bounds for Convolutional Codes and an Asymptotically Optimum Decoding Algorithm. *IEEE Trans. on Information Theory*, 13:260–269, 1967.

[Wah93] W. Wahlster. Verbmobil — Translation of Face–To–Face Dialogs. In *Proc. European Conf. on Speech Communication and Technology*, volume "Opening and Plenary Sessions", pages 29–38, Berlin, 1993.

[Wai88] A. Waibel. *Prosody and Speech Recognition.* Morgan Kaufmann Publishers Inc., San Mateo, California, 1988.

[Wai89a] A. Waibel, T. Hanazawa, G.E. Hinton, K. Shikano, and K.J. Lang. Phoneme Recognition Using Time–Delay Neural Networks. *IEEE Trans. on Acoustics, Speech and Signal Processing*, 37(3):328–339, 1989.

[Wai89b] A. Waibel, H. Sawai, and K. Shikano. Modularity and Scaling in Large Phonemic Neural Networks. *IEEE Trans. on Acoustics, Speech and*

Signal Processing, 37(12):1888–1898, 1989.

[Wai95] A. Waibel and M. Woszczyna. Recent Advances in JANUS: A Speech Translation System. In A.J. Rubio Ayuso and J.M. López Soler, editors, *Speech Recognition and Coding. New Advances and Trends*, volume 147 of *NATO ASI Series F*, pages 457–472. Springer, Berlin, 1995.

[Wai96] A. Waibel, M. Finke, D. Gates, M. Gavalda, T. Kemp, A. Lavie, L. Levin, M. Maier, L. Mayfield, A. McNair, K. Shima, T. Sloboda, M. Woszczyna, T. Zeppenfeld, and P. Zhan. JANUS–II — Translation of Spontaneous Conversational Speech. In *Proc. Int. Conf. on Acoustics, Speech and Signal Processing*, volume 1, pages 409–412, Atlanta, 1996.

[Wan92] M.Q. Wang and J. Hirschberg. Automatic Classification of Intonational Phrase Boundaries. *Computer Speech & Language*, 6(2):175–196, 1992.

[War91] W. Ward. Understanding Spontaneous Speech: the PHOENIX System. In *Proc. Int. Conf. on Acoustics, Speech and Signal Processing*, pages 365–367, Toronto, 1991.

[War95] V. Warnke. Landessprachenklassifikation. Bachelor's Thesis, Lehrstuhl für Mustererkennung (Informatik 5), Universität Erlangen–Nürnberg, 1995.

[War96] V. Warnke. Topik– und Topikgrenzen–Spotting. Diploma Thesis, Lehrstuhl für Mustererkennung (Informatik 5), Universität Erlangen–Nürnberg, 1996.

[Wei75] A. Weiss. *Syntax spontaner Gespräche*. Pädagogischer Verlag Schwann, Düsseldorf, 1975.

[Whi89] H. White. Learning in Artificial Neural Networks: A Statistical Perspective. *Neural Computation*, 1(3):425–464, 1989.

[Wid88] B. Widrow. Neural Networks for Adaptive Filtering and Pattern Recognition. *IEEE Computer*, 21(3):25–39, 1988.

[Wig92a] C. Wightman and M. Ostendorf. Automatic Recognition of Intonational Features. In *Proc. Int. Conf. on Acoustics, Speech and Signal Processing*, volume 1, pages I–221–I–224, San Francisco, CA, 1992.

[Wig92b] C.W. Wightman. *Automatic Detection of Prosodic Constituents*. PhD thesis, Boston University, 1992.

[Wig92c] C.W. Wightman, S. Shattuck-Hufnagel, M. Ostendorf, and P.J. Price. Segmental Durations in the Vicinity of Prosodic Boundaries. *Journal of the Acoustic Society of America*, 91:1707–1717, 1992.

[Wig94] C.W. Wightman and M. Ostendorf. Automatic Labeling of Prosodic Patterns. *IEEE Trans. on Speech and Audio Processing*, 2(3):469–481, 1994.

[Wil88] E.M. Willkop. *Gliederungspartikeln im Dialog*. Iudicium Verlag, München, 1988.

[Woo95] P.C. Woodland, C.J. Legetter, J.J. Odell, V. Valtchev, and S.J. Young. The 1994 HTP Large Vocabulary Speech Recognition System. In *Proc. Int. Conf. on Acoustics, Speech and Signal Processing*, volume 1, pages 73–76, Detroit, 1995.

[Zha94] Y. Zhao, R. Schwartz, J. Makhoul, and G. Zavaliagkos. Segmental Neural Net Optimization for Continuous Speech Recognition. In J.D. Cowan, G. Tesauro, and J. Alspector, editors, *Advances in Neural Information Processing Systems 6*, pages 1059–1066. Morgan Kaufmann, San Francisco, CA, 1994.

[Zha95] Y. Zhao, R. Schwartz, J Sroka, and J. Makhoul. Hierarchical Mixture of Experts Methodology Applied to Continuous Speech Recognition. In G. Tesauro, D.S. Touretzky, and T.K. Leen, editors, *Advances in Neural Information Processing Systems 7*, pages 859–865. Morgan Kaufmann, San Francisco, 1995.

[Zot94] A. Zottmann. Verbesserung der Worterkennung durch intonatorische Normierung cepstraler Merkmale. Diploma Thesis, Lehrstuhl für Mustererkennung (Informatik 5), Universität Erlangen–Nürnberg, 1994.

[Zwi67] E. Zwicker and R. Feldtkeller. *Das Ohr als Nachrichtenempfänger*. Hirzel Verlag, Stuttgart, 1967.

[Zwi90] E. Zwicker and H. Fastl. *Psychoacoustics. Facts and Models*, volume 22 of *Series in Information Sciences*. Springer–Verlag, Berlin, 1990.

Glossary

Abbreviations and Acronyms

ASU	automatic speech understanding	1
ATIS	air travel information system	128
BS_TEST	VERBMOBIL test corpus annotated with prosodic labels	154
BS_TRAIN	VERBMOBIL training corpus annotated with prosodic labels	155
CT	classification tree	54
DP	dynamic programming	43
DRS	discourse representation structures	86
D_TEST	VERBMOBIL test corpus annotated with dialog act labels	155
D_TRAIN	VERBMOBIL training corpus annotated with dialog act labels	155
EM	expectation maximization	23
ERNEST	the Erlangen semantic network system	74
EVAR	a speech understanding and dialog system for train time tables	74
F0	fundamental frequency	96
GPD	generalized probabilistic descent	39
HMM	hidden Markov model	32
HPSG	head driven phrase structural grammar	16
LM	any polygram language model	221
LM_j	polygram language model containing n-grams with $n \leq j$	226
LUD	labeled underspecified DRS	87
MAP	maximize a posteriori probabilities	39
MLP	multi–layer perceptron	26
MMI	maximum mutual information	39
MSCT	multi–level semantic classification tree	55
M_TRAIN	VERBMOBIL training corpus annotated with the prosodic–syntactic M labels	155
NN	neural network	26
NN–Feat/HMM	NN/HMM hybrid, where the NN performs a feature transformation	48

NN–Prob/HMM	NN/HMM hybrid, where the NN compute the observation probabilities	52
PP	prepositional phrase	115
SAMPA	a machine–readable phonetic alphabet	97
SCT	semantic classification tree	54
SPONTAN	corpus of spontaneous dialogs and reread texts	184
S_TRAIN	VERBMOBIL training corpus annotated with the syntactic S labels	155
LING_TEST	VERBMOBIL corpus for the final test of linguistic modules	155
LING_BIG	VERBMOBIL corpus for the evaluation of linguistic modules	155
LING_SMALL	small VERBMOBIL corpus for the evaluation of linguistic modules	155
TIME–OF–DAY	corpus of read time–of–day expressions	183
ToBI	tones and break indices – a system for prosodic transcription	131
TDNN	time–delay NN	30
TUG	trace unification grammar	83
VIT	verbmobil interface term	87
VSS	voice source signal	187
WSJ	Wall Street Journal	127
Paul	the underline in examples marks accented words or syllables	9
UNBELIEVABLE	an underlined word in small capital letters denotes extraordinary strong (emphatic) accent	123

Prosodic Labels

A	abbreviation for the union of PA, NA, and EK	204
B0	ERBA: no syntactic boundary	141
B0	VERBMOBIL: not prosodically marked word boundary	159
B1	ERBA: syntactic constituent boundary, expected *not* to be marked prosodically	141
B2	ERBA: syntactic constituent boundary, expected to be marked prosodically	141
B2	VERBMOBIL: prosodic constituent boundary	159
B3	ERBA: syntactic clause boundary	141
B3	VERBMOBIL: prosodic clause boundary	159
B9	VERBMOBIL: irregular prosodic boundary / hesitation lengthening	159
B	prosodic boundary labels for VERBMOBIL	159
CR	continuation–rise, often at boundaries of subordinary clauses	103
D0	no dialog act boundary	174
D3	boundary between two dialog acts	174
EK	VERBMOBIL: emphatic or contrastive accent	159
F	falling pitch, usually corresponds to a statement	103
M0	VERBMOBIL: no prosodic–syntactic clause boundary	165
M3	VERBMOBIL: prosodic–syntactic clause boundary	165
MB0	no prosodic–syntactic clause boundary: either M0 or MU+B[029]	203
MB3	prosodic–syntactic clause boundary: either M3 or MU+B3	203
MU	VERBMOBIL: ambiguous prosodic–syntactic boundary	165
M	VERBMOBIL: prosodic–syntactic boundary labels	165
NA	VERBMOBIL: marks all other accented words as carrying a secondary accent	159
PA	VERBMOBIL: most prominent (primary) accent within a prosodic clause	159
PSCB	prosodic–syntactic clause boundary symbol used in the VERBMOBIL parsers	248
R	rising pitch, usually corresponds to a question	103
S3+	VERBMOBIL: syntactic clause boundary	163
S3–	VERBMOBIL: no syntactic clause boundary	164
S3?	VERBMOBIL: ambiguous syntactic boundary	164
UA	VERBMOBIL: unaccented words	159

Mathematical Symbols

C	the acoustic score of a word hypothesis	72
I	number of HMM states	33
$I_{l_i}^{l_j}$	a word hypothesis	72
J	optional information attached to a word hypothesis, e.g., scores for prosodic attributes	72
K	number of NN nodes excluding bias	27
L	number of discrete/semi–continuous HMM observations	33
L_λ	number of normal densities contained in density λ	23
M	number of NN input nodes	27
N	number of NN output nodes	27
N_i'	set of NN nodes to which edges lead starting at n_i	29
N_i	set of NN nodes with edges leading to n_i	27
O_l	a particular discrete HMM observation	33
R	number of training samples	22
S_i	a particular HMM state	33
T	length of an utterance in terms of the number of feature vectors	30
U	an alphabet of prosodic class symbols	221
$U_l \in U$	a prosodic class symbol	221
V	set of k symbols V_i	40
$\boldsymbol{A}$	matrix of HMM transition probabilities	32
$\boldsymbol{B}$	matrix of HMM observation probabilities or probability densities	32
$\boldsymbol{\Omega}$	sequence of classes corresponding to one utterance	37
$\boldsymbol{\Sigma}$	covariance matrix	23
$\boldsymbol{\delta}$	vector of ideal decisions, or desired NN outputs	23
$\boldsymbol{\mu}$	mean vector	23
$\boldsymbol{\pi}_i$	initial probability of HMM state S_i	32
$\boldsymbol{a}$	parameters of a distribution free classifier	23
$\boldsymbol{c}$	feature vector	22
$\boldsymbol{d}^*$	vector of optimal decision functions	23
$\boldsymbol{d}$	vector of decision functions	23
$\boldsymbol{f}(\boldsymbol{x}), \boldsymbol{h}(\boldsymbol{x})$	patterns	21

$\boldsymbol{o}$	HMM observation sequence	33
$\boldsymbol{s}$	HMM state sequence	33
$\boldsymbol{v}$	sequence of symbols	40
Z	set of categories of symbols	41
Z_i	category of symbols	41
Ω_κ	name of a class to be recognized	22
Φ	backward pointer to the optimal predecessor during Viterbi search	43
Θ	NN activation function	27
α	back–propagation learning rate	28
α_{ni}	HMM forward variables	35
β	back–propagation momentum weight	28
β_{ni}	HMM backward variables	35
$\chi(n_i)$	remaining costs of the optimal path in a search graph leading form node n_i to a final node	45
δ_{ni}	in Viterbi search: inverse to the costs of a path ending in state S_i at time t_n	43
$\boldsymbol{p}$	duration of a particular realization of the phone p	348
ϵ	NN output error function	28
γ	threshold for the NN weight change during quick–propagation	30
$\gamma_{ni}, \gamma_{ni\nu}$	auxiliary variables for HMM training	39
ω_κ	sample patterns of class Ω_κ	23
$\psi(n_i)$	costs of the optimal path in a search graph leading form the initial node n_0 to node n_i	45
τ_I	average relative phone duration in a certain time interval I	348
θ	threshold	44
$\varphi(n_i)$	costs of a solution path in a search graph leading through node n_i	45
ϱ	index of one feature vector in the training set	28
$\vartheta(n_i, n_j)$	costs in a graph search for the transformation of node n_i into node n_j where n_j is an immediate successor of n_i	45
ξ	polygram interpolation weights	41
ξ_1	language model weight in the A* cost function of the word graph parser	258
ξ_2	prosody weight in the A* cost function of the word graph parser	258

ξ_3	weight balancing between the false–alarms rate and the correctly recognized MB3s	264
a_{ij}	HMM state transition probability	32
b_i	HMM observation probability density	32
d_i	decision function	23
f_i	activation of NN node n_i	27
k	number of classes to be distinguished, or vocabulary size	22
l_i	the index of a logical word graph node	72
m	number of classes (words) contained in an utterance	24
n	if not used as an index: the order of an n-gram model	40
n_i	NN, CT, or search graph node	27
o_n	random variable denoting an HMM observation	33
q	$c \in \mathbb{R}^q$	22
s_n	random variable denoting an HMM state	33
t_b, t_e	the time position where a word hypothesis begins/ends	72
t_n	the n–th discrete point in time	35
u_i	the prosodic class associated with the i–th word in an utterance	221
v_n	the n–th symbol out of a sequence	40
$w_{\lambda\nu}$	weights	23
w_{ij}	weight associated with NN edge from n_i to n_j	27
y_i	input to NN node n_i	27
z_i	random variable denoting a category of symbols	41
CRR	average of all $RR(\Omega_\kappa)$	25
RA	accuracy of a continuous speech recognizer	25
RR	recognition rate	25
$RR(\Omega_\kappa)$	recognition rate of class Ω_κ	25

Index

Appendix A

Equations and Tables

A.1 Energy Computation

In the experiments for this book, the signal energy was computed after the following formulas, which are taken from [Kie97]; they are a simplification of the definition of loudness as it is given in in [Pau72] or in [Zwi90, Sec. 8.7].
Using a Hamming window w_n^H of width 40 ms the intensity I_m of (a 10 msec) frame m is approximated by

$$\tilde{I}_m = \frac{\sum\limits_{n=0}^{N-1} f_{m+n}^2 w_n^H}{\sum\limits_{n=0}^{N-1} w_n^H} \tag{A.1}$$

where f_{m+n} denotes a signal sample value. Finally the energy Lh_m of frame m is computed by

$$Lh_m = \left(\frac{\tilde{I}_m}{\tilde{I}_0}\right)^{0.3} \tag{A.2}$$

where the reference intensity I_0 is defined as

$$\tilde{I}_0 \approx 1073.74 \tag{A.3}$$

A.2 Intrinsic Normalization of the Phone Duration

The phone duration plotted in the Figures 4.3, 4.4, and 4.5 as well as the one used as input to the acoustic–prosodic classifiers as described in Section 6.4 has been normalized with respect to the current speaking rate and to the phone intrinsic values according to equations, which have been theoretically and empirically justified in [Wig92c]. The relative duration $\widehat{d_p}$ of a specific phone p is determined by

$$\widehat{D_p} = \frac{D_p - \tau_I * \mu_p}{\tau_I * \sigma_p}$$

where D_p is the actual duration in msec of a particular realization of the phone p, and

$$\mu_p = \frac{1}{N_p} \sum_{p \in \omega} D_p$$

is the average intrinsic phone duration and

$$\sigma_p = \frac{1}{N_p} \sum_{p \in \omega} D_p^2 - \mu_p^2$$

is the standard deviation of the duration of this phone, both determined over some training speech data ω. N_p denotes the frequency of phone p in the training speech data. With

$$\tau_I = \sum_{p \in I} \frac{D_p}{\mu_p}$$

the average relative phone duration, the reciprocal of the *speaking rate*, in a certain time interval I is computed. The time interval I can, for example, be a small time window around a word, or an entire turn.

A.3 Phonetic Alphabets

PLOSIVES			
Papa	pie	p	p
Bube	by	b	b
Tod	tie	t	t
müde	dye	d	d
Kakao	kye	k	k
gegen	guy	g	g
FRICATIVES			
Tasse	sea	s	s
Sieb	mizzen	z	z
Tasche	shy	ʃ	S
Garage	vision	ʒ	Z
Fisch	fee	f	f
Wasser	vie	v	v
Licht	—	ç	C
Buch	—	x	x
Kachel	—	χ	x
(uvular fricative)	—	ʁ	R
Hose	how	h	h
—	thin	θ	T
—	breathe	ð	D
AFFRICATES			
Quatsch	cheap	ʧ	tS
—	jive	ʤ	dZ
Pfirsich	—	pf	pf
LIQUIDS			
—	read	ɹ	r
(alveolar 'r')	—	r	r
(uvular vibrant)	—	ʀ	R
lallen	lack	l	l
GLIDES			
Jacke	yack	j	j
—	whack	w	w

VOWELS			
Land	—	a	a
—	hot	ɑ	Q
Butter	—	ɐ	6
bitte	cut	ə	@
—	her	ɚ	3
—	bad	æ	{
Leben	—	e	e
Bett	bed	ɛ	E
Biene	heat	i	i
Kind	hit	ɪ	I
boden	—	o	o
Onkel	haw	ɔ	O
schön	—	ø	2
Vὂlle	—	œ	9
Kuh	hoot	u	u
Kuß	hood	ʊ	U
Gefühl	—	y	y
Trüffel	—	Y	Y
DIPHTHONGS			
Wein	bite	aɪ	aI
Pause	bout	aʊ	aU
teuer	boy	ɔY	OY
—	here	iɹ	I@
—	hair	ɛɹ	e@
—	hired	aɪɹ	aI3
NASALS			
Mama	prism	m	m
Nenner	prison	n	n
schwanger	sing	ŋ	N
OTHERS			
(lexical accent)		'	'
(lengthening of preceding vowel)		:	:

Table A.1: Transcription symbols of German and English phones after IPA [Int63] and SAMPA [Fou89, pp. 141–159] alphabets in columns 3 and 4 respectively; if applicable German and American English examples are given in columns 1 and 2, after [ST95a, p. 224] and [Lad82, Chap. 2] respectively. The phonetic realization of the German consonant r depends on dialect, style, and phonemic context and can be r, ʀ, or ʁ.

A.4 Examples for Feature Sets for the Acoustic–prosodic Classification

In Section 6.4.2 we described the types of features used as input for the classifiers in Section 6.4.3. Different feature sets were used for the different classifiers. As an example, Table A.2 shows the full set of features used for the best NN for B3/B[029] classification on VERBMOBIL, cf. Table 6.5. In total one feature vector used as input to the NN for the classification of one word boundary consists of 274 syllable based features computed over a context of ± 2 syllables or ± 2 words around the final syllable before the boundary. Table A.3 shows the best feature set used for the accent and boundary classification on ERBA, where the classes A[01]+B[01], A[01]+B2, A[01]+B3, A[23]+B[01], A[23]+B2, and A[23]+B3 are distinguished, cf. Section 6.4.3, Table 6.4. In Section 7.1 a preliminary version of this classifier is used for the NN/HMMM hybrid system. The 142 features used in this classifier are given in Table A.4.

The first column in the tables shows the type of the features, the other columns specify the intervals in the speech signal over which the feature is computed. The numbers refer to units which are either syllable nuclei, syllables or words. All numbers in the specifications of the intervals are relative to the unit preceding the boundary: "0" denotes the unit preceding the boundary, "-1" is the unit preceding unit "0", "1" refers to the unit succeeding the boundary, and so on. A notion like "[-2;0]" in the column "syllable intervals" corresponds to the interval from the beginning of syllable "2" to the end of the syllable preceding the boundary. The same interval specification in the column "syllable nuclei" refers to the same syllables, but only the speech segments corresponding to the syllable nuclei in these syllables are used for the feature extraction. The abbreviations in the first column of the table have the following meaning:

- "Duration": average (normalized) phone duration in the interval.
- "Ene": features computed from the energy contour.
- "F0": features computed from the non–zero F0 values.
- "Max","Min": the maximum/minimum in the specified interval is computed.
- "Onset","Offset": the first/final non–zero F0 value in the the specified interval.
- "MaxPos","MinPos","OnsetPos", "OffsetPos": the position of the respective value relative to the end of syllable "0".
- "Mv": the mean value in the interval.
- "Rc": the regression coefficient of the regression line computed over the specified interval, cf. Section 6.3.

- "Rr": the square root of the mean of the squared differences between the actual F0 or energy values and the value of the regression line at the same position.
- "Norm": phone intrinsic normalization.
- "Norm(WorAcc)": context–dependent phone intrinsic normalization: different intrinsic mean and standard deviation are used for the phone in a word accent carrying syllable and the same phone in any other syllable.
- "Norm(SylPos)": in addition to "Norm(WorAcc)" it is distinguished if the phone occurs in the first, last or in a word–internal syllable.
- "Syllable_nucleus_index": an integer corresponding to the phone being in nucleus position; each phone in the inventory is assigned a unique integer. ENDE:
- "Is_lex_WorAcc": a flag; it is one if the syllable carries the lexical word accent, 0 else.
- "Is_word_final": a flag; it is one if the syllable is in word final position, zero else.

The feature "Pause_length_after_the_word" only differs from "Pause_length_after_the_syllable" in the case of syllable–wise (accent) classification.

type of feature	syllable interval	syllable nucleus interval	word interval
Duration	[-2;-2] [-1;-1] [0;0] [1;1] [2;2]	[-2;-2] [-1;-1] [0;0] [1;1] [2;2]	[-2;-1] [-1;-1] [0;0] [1;1] [1;2]
Duration.Norm	[-2;-2] [-1;-1] [0;0] [1;1] [2;2]	[-2;-2] [-1;-1] [0;0] [1;1] [2;2]	[-2;-1] [-1;-1] [0;0] [1;1] [1;2]
Duration.Norm(WorAcc)	[-2;-2] [-1;-1] [0;0] [1;1] [2;2]	[-2;-2] [-1;-1] [0;0] [1;1] [2;2]	[-2;-1] [-1;-1] [0;0] [1;1] [1;2]
Duration.Norm(SylPos)	[-2;-2] [-1;-1] [0;0] [1;1] [2;2]	[-2;-2] [-1;-1] [0;0] [1;1] [2;2]	[-2;-1] [-1;-1] [0;0] [1;1] [1;2]
Ene-Max	[-2;-2] [-1;-1] [0;0] [1;1] [2;2]		[-2;-1] [-1;-1] [0;0] [1;1] [1;2]
Ene-Max.Norm	[-2;-2] [-1;-1] [0;0] [1;1] [2;2]		[-2;-1] [-1;-1] [0;0] [1;1] [1;2]
Ene-Max.Norm(WorAcc)	[-2;-2] [-1;-1] [0;0] [1;1] [2;2]		[-2;-1] [-1;-1] [0;0] [1;1] [1;2]
Ene-Max.Norm(SylPos)	[-2;-2] [-1;-1] [0;0] [1;1] [2;2]		[-2;-1] [-1;-1] [0;0] [1;1] [1;2]
Ene-MaxPos	[-2;-2] [-1;-1] [0;0] [1;1] [2;2]		[-2;-1] [-1;-1] [0;0] [1;1] [1;2]
Ene-Mv	[-2;-2] [-1;-1] [0;0] [1;1] [2;2]		[-2;-1] [-1;-1] [0;0] [1;1] [1;2]
Ene-Mv.Norm	[-2;-2] [-1;-1] [0;0] [1;1] [2;2]		[-2;-1] [-1;-1] [0;0] [1;1] [1;2]
Ene-Mv.Norm(WorAcc)	[-2;-2] [-1;-1] [0;0] [1;1] [2;2]		[-2;-1] [-1;-1] [0;0] [1;1] [1;2]
Ene-Mv.Norm(SylPos)	[-2;-2] [-1;-1] [0;0] [1;1] [2;2]		[-2;-1] [-1;-1] [0;0] [1;1] [1;2]
Ene-Rc			[-2;-1] [-1;-1] [0;0] [1;1] [1;2] [-1;0] [-2;0] [-1;1] [-2;1] [-1;2] [-2;2]
Ene-Rr			[-2;-1] [-1;-1] [0;0] [1;1] [1;2] [-1;0] [-2;0] [-1;1] [-2;1] [-1;2] [-2;2]
F0-Max	[-2;-2] [-1;-1] [0;0] [1;1] [2;2]		[-2;-1] [-1;-1] [0;0] [1;1] [1;2]
F0-MaxPos	[-2;-2] [-1;-1] [0;0] [1;1] [2;2]		[-2;-1] [-1;-1] [0;0] [1;1] [1;2]
F0-Min	[-2;-2] [-1;-1] [0;0] [1;1] [2;2]		[-2;-1] [-1;-1] [0;0] [1;1] [1;2]
F0-MinPos	[-2;-2] [-1;-1] [0;0] [1;1] [2;2]		[-2;-1] [-1;-1] [0;0] [1;1] [1;2]
F0-Mv	[-2;-2] [-1;-1] [0;0] [1;1] [2;2]		[-2;-1] [-1;-1] [0;0] [1;1] [1;2]
F0-Offset			[-1;-1] [0;0]
F0-OffsetPos			[-1;-1] [0;0]
F0-Onset			[0;0] [1;1]
F0-OnsetPos			[0;0] [1;1]
F0-Rc			[-2;-1] [-1;-1] [0;0] [1;1] [1;2] [-1;0] [-2;0] [-1;1] [-2;1] [-1;2] [-2;2]
F0-Rr			[-2;-1] [-1;-1] [0;0] [1;1] [1;2] [-1;0] [-2;0] [-1;1] [-2;1] [-1;2] [-2;2]
Syllable_nulceus_index	[-2;-2] [-1;-1] [0;0] [1;1] [2;2]		
Is_lex_WorAcc	[-2;-2] [-1;-1] [0;0] [1;1] [2;2]		
Is_word_final	[-2;-2] [-1;-1] [0;0] [1;1] [2;2]		
Speaking_rate, Speaking_rate.Norm(WorAcc), Speaking_rate.Norm(SylPos)			
Pause_length_after_the_syllable, Pause_length_before_the_syllable			
Pause_length_after_the_word, Pause_length_before_the_word			

Table A.2: Features used as input for the best NN for B3/B[029] classification on VERBMOBIL, after [Kie97].

type of feature	syllable interval	syllable nucleus interval
Duration	[0;0]	[0;0]
Duration.Norm		[-4;-4] [-3;-3] [-2;-2] [-1;-1] [0;0] [1;1] [2;2] [3;3] [4;4] [-6;-6] [-5;-5] [-6;6] [5;5] [6;6]
Duration.Norm(WorAcc)	[0;0]	[0;0]
Duration.Norm(SylPos)	[0;0]	[0;0]
Ene-Max	[-4;-4] [-3;-3] [-2;-2] [-1;-1] [0;0] [1;1] [2;2] [3;3] [4;4]	
Ene-MaxPos	[-4;-4] [-3;-3] [-2;-2] [-1;-1] [0;0] [1;1] [2;2] [3;3] [4;4]	
Ene-Rc	[-6;0] [-5;0] [-4;0] [-2;0] [0;2] [0;4] [0;5] [0;6]	
F0-Max	[-4;-4] [-3;-3] [-2;-2] [-1;-1] [0;0] [1;1] [2;2] [3;3] [4;4] [-6;-6] [-5;-5] [5;5] [6;6]	
F0-MaxPos	[-4;-4] [-3;-3] [-2;-2] [-1;-1] [0;0] [1;1] [2;2] [3;3] [4;4] [-6;-6] [-5;-5] [5;5] [6;6]	
F0-Min	[-4;-4] [-3;-3] [-2;-2] [-1;-1] [0;0] [1;1] [2;2] [3;3] [4;4] [-6;-6] [-5;-5] [5;5] [6;6]	
F0-Mv	[-4;-4] [-3;-3] [-2;-2] [-1;-1] [0;0] [1;1] [2;2] [3;3] [4;4] [-6;-6] [-5;-5] [5;5] [6;6]	
F0-Rc	[-6;0] [-5;0] [-4;0] [-3;0] [-2;0] [-1;0] [0;0] [0;1] [0;2] [-2;-1] [1;2] [0;3] [0;4] [0;5] [0;6]	
Is_lex_WorAcc	[-6;-6] [-5;-5] [-4;-4] [-3;-3] [-2;-2] [-1;-1] [0;0] [1;1] [2;2] [3;3] [4;4] [5;5] [6;6]	
Is_word_final	[-6;-6] [-5;-5] [-4;-4] [-3;-3] [-2;-2] [-1;-1] [0;0] [1;1] [2;2] [3;3] [4;4] [5;5] [6;6]	
Pause_length_after_the_syllable, Pause_length_before_the_syllable		
Pause_length_after_the_word, Pause_length_before_the_word		

Table A.3: Features used as input for the best NN for combined boundary and accent classification on ERBA, cf. Section 6.4.3.

type of feature	syllable interval	syllable nucleus interval
Duration	[0;0]	[0;0]
Duration.Norm	[0;0]	[0;0]
Duration.Norm(WorAkz)	[0;0]	[0;0]
Duration.Norm(SylPos)	[0;0]	[0;0]
Ene-Max	[-2;-2] [-1;-1] [0;0] [1;1] [2;2] [-2;-1] [1;2]	
Ene-Max.Norm	[-2;-2] [-1;-1] [0;0] [1;1] [2;2] [-2;-1] [1;2]	
Ene-Max.Norm(WorAcc)	[-2;-2] [-1;-1] [0;0] [1;1] [2;2] [-2;-1] [1;2]	
Ene-Max.Norm(SylPos)	[-2;-2] [-1;-1] [0;0] [1;1] [2;2] [-2;-1] [1;2]	
Ene-MaxPos	[-2;-2] [-1;-1] [0;0] [1;1] [2;2] [-2;-1] [1;2]	
Ene-Mv	[-2;-2] [-1;-1] [0;0] [1;1] [2;2] [-2;-1] [1;2]	
Ene-Mv.Norm	[-2;-2] [-1;-1] [0;0] [1;1] [2;2] [-2;-1] [1;2]	
Ene-Mv.Norm(WorAcc)	[-2;-2] [-1;-1] [0;0] [1;1] [2;2] [-2;-1] [1;2]	
Ene-Mv.Norm(SylPos)	[-2;-2] [-1;-1] [0;0] [1;1] [2;2] [-2;-1] [1;2]	
F0-Max	[-2;-2] [-1;-1] [0;0] [1;1] [2;2] [-2;-1] [1;2]	
F0-MaxPos	[-2;-2] [-1;-1] [0;0] [1;1] [2;2] [-2;-1] [1;2]	
F0-Min	[-2;-2] [-1;-1] [0;0] [1;1] [2;2] [-2;-1] [1;2]	
F0-MinPos	[-2;-2] [-1;-1] [0;0] [1;1] [2;2] [-2;-1] [1;2]	
F0-Off	[-2;-2] [-1;-1] [0;0] [1;1] [2;2] [-2;-1] [1;2]	
F0-OffsetPos	[-2;-2] [-1;-1] [0;0] [1;1] [2;2] [-2;-1] [1;2]	
F0-On	[-2;-2] [-1;-1] [0;0] [1;1] [2;2] [-2;-1] [1;2]	
F0-OnsetPos	[-2;-2] [-1;-1] [0;0] [1;1] [2;2] [-2;-1] [1;2]	
F0-Rc	[-2;0] [-1;0] [0;0] [0;1] [0;2] [-2;-1] [1;2] [-4;0] [0;4]	
Is_lex_WorAcc	[0;0]	
Is_word_final	[0;0]	
Speaking_rate, Speaking_rate.Norm(WorAcc), Speaking_rate.Norm(SylPos)		
Pause_length_after_the_syllable		

Table A.4: Features used as input for a preliminary NN for combined boundary and accent classification on ERBA, cf. Section 7.1.

Appendix B

Examples

B.1 Ja zur Not geht's auch am Samstag

The following word chain is taken from the VERBMOBIL corpus:

ja | zur Not | geht's | auch | am Samstag

Vertical bars separate different parts of the turn and at the same time indicate possible positions of syntactic boundaries. A combination of these parts results in 36 alternative syntactic structures, which are listed in Table B.1; for all of them a context can be found, where a meaningful interpretation is possible, but not each pair of these cases differs in the meaning. In Section 4.2 for four of these alternatives the meaning is explained within a possible dialog context. Note, that these are not all possibilities to place punctuation marks.

B.2 Speech Signals on the WORLD WIDE WEB

This part of the appendix actually resides on the WORLD WIDE WEB under

http://www5.informatik.uni–erlangen.de/Private/kp/diss–figs/diss–figs.html

and contains enlarged versions of Figures 4.3, 4.4 and 4.5 and it allows to listen to the speech signals plotted in these figures. This should support the comprehensibility of the explanation of these examples.
A number of transliterations of VERBMOBIL turns were given as examples in this book. For those examples, where it was considered important to be able to listen to the speech signal, these can also be obtained from the WORLD WIDE WEB page.

1	Ja		zur Not		geht's		auch		am Samstag	.
2	Ja		zur Not		geht's		auch		am Samstag	?
3	Ja	.	Zur Not		geht's		auch		am Samstag	.
4	Ja	.	Zur Not		geht's		auch		am Samstag	?
5	Ja	?	Zur Not		geht's		auch		am Samstag	.
6	Ja	?	Zur Not		geht's		auch		am Samstag	?
7	Ja		zur Not	.	Geht's		auch		am Samstag	?
8	Ja		zur Not	?	Geht's		auch		am Samstag	?
9	Ja	?	Zur Not	.	Geht's		auch		am Samstag	?
10	Ja	?	Zur Not	?	Geht's		auch		am Samstag	?
11	Ja	.	Zur Not	.	Geht's		auch		am Samstag	?
12	Ja	.	Zur Not	?	Geht's		auch		am Samstag	?
13	Ja		zur Not		geht's	.	Auch		am Samstag	.
14	Ja		zur Not		geht's	?	Auch		am Samstag	.
15	Ja		zur Not		geht's	?	Auch		am Samstag	?
16	Ja		zur Not		geht's	.	Auch		am Samstag	?
17	Ja	.	Zur Not		geht's	.	Auch		am Samstag	.
18	Ja	.	Zur Not		geht's	.	Auch		am Samstag	?
19	Ja	.	Zur Not		geht's	?	Auch		am Samstag	.
20	Ja	.	Zur Not		geht's	?	Auch		am Samstag	?
21	Ja	?	Zur Not		geht's	.	Auch		am Samstag	.
22	Ja	?	Zur Not		geht's	.	Auch		am Samstag	?
23	Ja	?	Zur Not		geht's	?	Auch		am Samstag	.
24	Ja	?	Zur Not		geht's	?	Auch		am Samstag	?
25	Ja		zur Not		geht's		auch	.	Am Samstag	.
26	Ja		zur Not		geht's		auch	?	Am Samstag	.
27	Ja		zur Not		geht's		auch	.	Am Samstag	?
28	Ja		zur Not		geht's		auch	?	Am Samstag	?
29	Ja	.	Zur Not		geht's		auch	.	Am Samstag	.
30	Ja	.	Zur Not		geht's		auch	?	Am Samstag	.
31	Ja	.	Zur Not		geht's		auch	.	Am Samstag	?
32	Ja	.	Zur Not		geht's		auch	?	Am Samstag	?
33	Ja	?	Zur Not		geht's		auch	.	Am Samstag	.
34	Ja	?	Zur Not		geht's		auch	?	Am Samstag	.
35	Ja	?	Zur Not		geht's		auch	.	Am Samstag	?
36	Ja	?	Zur Not		geht's		auch	?	Am Samstag	?

Table B.1: Ja zur Not geht's auch am Samstag: Different syntactic structures.

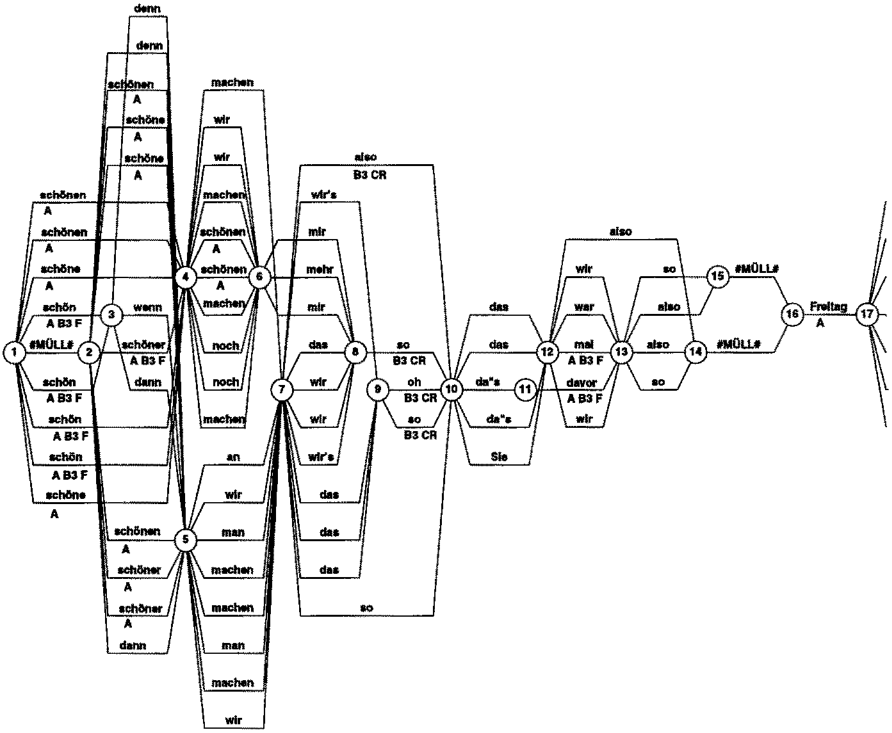


Figure B.1: A prosodically classified word graph, which was generated with the VERBMOBIL recognizer of Daimler Benz for a turn of the VERBMOBIL corpus. As classifier the combination of B3/B[029]–NN and M3/M0–LM was used.

B.3 A Prosodically Classified Word Graph

In Figure 8.1 we have shown a word graph which was prosodically scored by the B3/B[029]–NN. There we wanted to show that different edges ending in the same word graph node might be classified differently depending on the time alignment of the word hypotheses and the phone intrinsic normalization. In order to show the impact of the polygram on the prosodic word graph scoring, the same word graph is also depicted in Figure B.1, however, this time it was scored using the combined "B3/B[029]–NN & M3/M0–LM" classifier as used in the VERBMOBIL system. It can be seen that the polygram corrects obvious errors made by the B3/B[029]–NN.

Lecture Notes in Computer Science

Vol. 1281: M. Abadi, T. Ito (Eds.), Theoretical Aspects of Computer Software. Proceedings, 1997. XI, 639 pages. 1997.

Vol. 1282: D. Garlan, D. Le Métayer (Eds.), Coordination Languages and Models. Proceedings, 1997. X, 435 pages. 1997.

Vol. 1283: M. Müller-Olm, Modular Compiler Verification. XV, 250 pages. 1997.

Vol. 1284: R. Burkard, G. Woeginger (Eds.), Algorithms — ESA '97. Proceedings, 1997. XI, 515 pages. 1997.

Vol. 1285: X. Jao, J.-H. Kim, T. Furuhashi (Eds.), Simulated Evolution and Learning. Proceedings, 1996. VIII, 231 pages. 1997. (Subseries LNAI).

Vol. 1286: C. Zhang, D. Lukose (Eds.), Multi-Agent Systems. Proceedings, 1996. VII, 195 pages. 1997. (Subseries LNAI).

Vol. 1287: T. Kropf (Ed.), Formal Hardware Verification. XII, 367 pages. 1997.

Vol. 1288: M. Schneider, Spatial Data Types for Database Systems. XIII, 275 pages. 1997.

Vol. 1289: G. Gottlob, A. Leitsch, D. Mundici (Eds.), Computational Logic and Proof Theory. Proceedings, 1997. VIII, 348 pages. 1997.

Vol. 1291: D.G. Feitelson, L. Rudolph (Eds.), Job Scheduling Strategies for Parallel Processing. Proceedings, 1997. VII, 299 pages. 1997.

Vol. 1292: H. Glaser, P. Hartel, H. Kuchen (Eds.), Programming Languages: Implementations, Logigs, and Programs. Proceedings, 1997. XI, 425 pages. 1997.

Vol. 1294: B.S. Kaliski Jr. (Ed.), Advances in Cryptology — CRYPTO '97. Proceedings, 1997. XII, 539 pages. 1997.

Vol. 1295: I. Prívara, P. Ružička (Eds.), Mathematical Foundations of Computer Science 1997. Proceedings, 1997. X, 519 pages. 1997.

Vol. 1296: G. Sommer, K. Daniilidis, J. Pauli (Eds.), Computer Analysis of Images and Patterns. Proceedings, 1997. XIII, 737 pages. 1997.

Vol. 1297: N. Lavrač, S. Džeroski (Eds.), Inductive Logic Programming. Proceedings, 1997. VIII, 309 pages. 1997. (Subseries LNAI).

Vol. 1298: M. Hanus, J. Heering, K. Meinke (Eds.), Algebraic and Logic Programming. Proceedings, 1997. X, 286 pages. 1997.

Vol. 1299: M.T. Pazienza (Ed.), Information Extraction. Proceedings, 1997. IX, 213 pages. 1997. (Subseries LNAI).

Vol. 1300: C. Lengauer, M. Griebl, S. Gorlatch (Eds.), Euro-Par'97 Parallel Processing. Proceedings, 1997. XXX, 1379 pages. 1997.

Vol. 1301: M. Jazayeri (Ed.), Software Engineering - ESEC/FSE'97. Proceedings, 1997. XIII, 532 pages. 1997.

Vol. 1302: P. Van Hentenryck (Ed.), Static Analysis. Proceedings, 1997. X, 413 pages. 1997.

Vol. 1303: G. Brewka, C. Habel, B. Nebel (Eds.), KI-97: Advances in Artificial Intelligence. Proceedings, 1997. XI, 413 pages. 1997. (Subseries LNAI).

Vol. 1304: W. Luk, P.Y.K. Cheung, M. Glesner (Eds.), Field-Programmable Logic and Applications. Proceedings, 1997. XI, 503 pages. 1997.

Vol. 1305: D. Corne, J.L. Shapiro (Eds.), Evolutionary Computing. Proceedings, 1997. X, 313 pages. 1997.

Vol. 1307: R. Kompe, Prosody in Speech Understanding Systems. XIX, 357 pages. 1997. (Subseries LNAI).

Vol. 1308: A. Hameurlain, A M. Tjoa (Eds.), Database and Expert Systems Applications. Proceedings, 1997. XVII, 688 pages. 1997.

Vol. 1309: R. Steinmetz, L.C. Wolf (Eds.), Interactive Distributed Multimedia Systems and Telecommunication Services. Proceedings, 1997. XIII, 466 pages. 1997.

Vol. 1310: A. Del Bimbo (Ed.), Image Analysis and Processing. Proceedings, 1997. Volume I. XXII, 722 pages. 1997.

Vol. 1311: A. Del Bimbo (Ed.), Image Analysis and Processing. Proceedings, 1997. Volume II. XXII, 794 pages. 1997.

Vol. 1312: A. Geppert, M. Berndtsson (Eds.), Rules in Database Systems. Proceedings, 1997. VII, 214 pages. 1997.

Vol. 1313: J. Fitzgerald, C.B. Jones, P. Lucas (Eds.), FME'97: Industrial Applications and Strengthened Foundations of Formal Methods. Proceedings, 1997. XIII, 685 pages. 1997.

Vol. 1314: S. Muggleton (Ed.), Inductive Logic Programming. Proceedings, 1996. VIII, 397 pages. 1997. (Subseries LNAI).

Vol. 1315: G. Sommer, J.J. Koenderink (Eds.), Algebraic Frames for the Perception-Action Cycle. Proceedings, 1997. VIII, 395 pages. 1997.

Vol. 1317: M. Leman (Ed.), Music, Gestalt, and Computing. IX, 524 pages. 1997. (Subseries LNAI).

Vol. 1320: M. Mavronicolas, P. Tsigas (Eds.), Distributed Algorithms. Proceedings, 1997. X, 333 pages. 1997.

Vol. 1321: M. Lenzerini (Ed.), AI*IA 97: Advances in Artificial Intelligence. Proceedings, 1997. XII, 459 pages. 1997. (Subseries LNAI).

Vol. 1324: C. Peters, C. Thanos (Eds.), Research and Advanced Technology for Digital Libraries. Proceedings, 1997. XI, 423 pages. 1997.

Lecture Notes in Artificial Intelligence (LNAI)

Vol. 1159: D.L. Borges, C.A.A. Kaestner (Eds.), Advances in Artificial Intelligence. Proceedings, 1996. XI, 243 pages. 1996.

Vol. 1160: S. Arikawa, A.K. Sharma (Eds.), Algorithmic Learning Theory. Proceedings, 1996. XVII, 337 pages. 1996.

Vol. 1168: I. Smith, B. Faltings (Eds.), Advances in Case-Based Reasoning. Proceedings, 1996. IX, 531 pages. 1996.

Vol. 1171: A. Franz, Automatic Ambiguity Resolution in Natural Language Processing. XIX, 155 pages. 1996.

Vol. 1177: J.P. Müller, The Design of Intelligent Agents. XV, 227 pages. 1996.

Vol. 1187: K. Schlechta, Nonmonotonic Logics. IX, 243 pages. 1997.

Vol. 1188: T.P. Martin, A.L. Ralescu (Eds.), Fuzzy Logic in Artificial Intelligence. Proceedings, 1995. VIII, 272 pages. 1997.

Vol. 1193: J.P. Müller, M.J. Wooldridge, N.R. Jennings (Eds.), Intelligent Agents III. XV, 401 pages. 1997.

Vol. 1195: R. Trappl, P. Petta (Eds.), Creating Personalities for Synthetic Actors. VII, 251 pages. 1997.

Vol. 1198: H. S. Nwana, N. Azarmi (Eds.), Software Agents and Soft Computing: Towards Enhancing Machine Intelligents. XIV, 298 pages. 1997.

Vol. 1202: P. Kandzia, M. Klusch (Eds.), Cooperative Information Agents. Proceedings, 1997. IX, 287 pages. 1997.

Vol. 1208: S. Ben-David (Ed.), Computational Learning Theory. Proceedings, 1997. VIII, 331 pages. 1997.

Vol. 1209: L. Cavedon, A. Rao, W. Wobcke (Eds.), Intelligent Agent Systems. Proceedings, 1996. IX, 188 pages. 1997.

Vol. 1211: E. Keravnou, C. Garbay, R. Baud, J. Wyatt (Eds.), Artificial Intelligence in Medicine. Proceedings, 1997. XIII, 526 pages. 1997.

Vol. 1216: J. Dix, L. Moniz Pereira, T.C. Przymusinski (Eds.), Non-Monotonic Extensions of Logic Programming. Proceedings, 1996. XI, 224 pages. 1997.

Vol. 1221: G. Weiß (Ed.), Distributed Artificial Intelligence Meets Machine Learning. Proceedings, 1996. X, 294 pages. 1997.

Vol. 1224: M. van Someren, G. Widmer (Eds.), Machine Learning: ECML-97. Proceedings, 1997. XI, 361 pages. 1997.

Vol. 1227: D. Galmiche (Ed.), Automated Reasoning with Analytic Tableaux and Related Methods. Proceedings, 1997. XI, 373 pages. 1997.

Vol. 1228: S.-H. Nienhuys-Cheng, R. de Wolf, Foundations of Inductive Logic Programming. XVII, 404 pages. 1997.

Vol. 1236: E. Maier, M. Mast, S. LuperFoy (Eds.), Dialogue Processing in Spoken Language Systems. Proceedings, 1996. VIII, 220 pages. 1997.

Vol. 1237: M. Boman, W. Van de Velde (Eds.), Multi-Agent Rationality. Proceedings, 1997. XII, 254 pages. 1997.

Vol. 1244: D. M. Gabbay, R. Kruse, A. Nonnengart, H.J. Ohlbach (Eds.), Qualitative and Quantitative Practical Reasoning. Proceedings, 1997. X, 621 pages. 1997.

Vol. 1249: W. McCune (Ed.), Automated Deduction – CADE-14. Proceedings, 1997. XIV, 462 pages. 1997.

Vol. 1257: D. Lukose, H. Delugach, M. Keeler, L. Searle, J. Sowa (Eds.), Conceptual Structures: Fulfilling Peirce's Dream. Proceedings, 1997. XII, 621 pages. 1997.

Vol. 1263: J. Komorowski, J. Zytkow (Eds.), Principles of Data Mining and Knowledge Discovery. Proceedings, 1997. IX, 397 pages. 1997.

Vol. 1266: D.B. Leake, E. Plaza (Eds.), Case-Based Reasoning Research and Development. Proceedings, 1997. XIII, 648 pages. 1997.

Vol. 1265: J. Dix, U. Furbach, A. Nerode (Eds.), Logic Programming and Nonmonotonic Reasoning. Proceedings, 1997. X, 453 pages. 1997.

Vol. 1285: X. Jao, J.-H. Kim, T. Furuhashi (Eds.), Simulated Evolution and Learning. Proceedings, 1996. VIII, 231 pages. 1997.

Vol. 1286: C. Zhang, D. Lukose (Eds.), Multi-Agent Systems. Proceedings, 1996. VII, 195 pages. 1997.

Vol. 1297: N. Lavrač, S. Džeroski (Eds.), Inductive Logic Programming. Proceedings, 1997. VIII, 309 pages. 1997.

Vol. 1299: M.T. Pazienza (Ed.), Information Extraction. Proceedings, 1997. IX, 213 pages. 1997.

Vol. 1303: G. Brewka, C. Habel, B. Nebel (Eds.), KI-97: Advances in Artificial Intelligence. Proceedings, 1997. XI, 413 pages. 1997.

Vol. 1307: R. Kompe, Prosody in Speech Understanding Systems. XIX, 357 pages. 1997.

Vol. 1314: S. Muggleton (Ed.), Inductive Logic Programming. Proceedings, 1996. VIII, 397 pages. 1997.

Vol. 1317: M. Leman (Ed.), Music, Gestalt, and Computing. IX, 524 pages. 1997.

Vol. 1321: M. Lenzerini (Ed.), AI*IA 97: Advances in Artificial Intelligence. Proceedings, 1997. XII, 459 pages. 1997.