Large Vocabulary Continuous Speech Recognition Using Linguistic Features and Constraints

by

Min Tang

Submitted to the Department of Electrical Engineering and Computer Science
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy in Computer Science

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

May 2005

Author ....
Department of Electrical Engineering and Computer Science
May 19, 2005

Certified by..
Stephanie Seneff
Principle Research Scientist
Thesis Supervisor

Certified by..
Victor W. Zue
Professor
Thesis Supervisor

Accepted by ....
Arthur C. Smith
Chairman, Department Committee on Graduate Students

Large Vocabulary Continuous Speech Recognition Using Linguistic Features and Constraints

by

Min Tang

Submitted to the Department of Electrical Engineering and Computer Science
on May 19, 2005, in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy in Computer Science

Abstract

Automatic speech recognition (ASR) is a process of applying constraints, as encoded in the computer system (the recognizer), to the speech signal until ambiguity is satisfactorily resolved to the extent that only one sequence of words is hypothesized. Such constraints fall naturally into two categories. One deals with the the ordering of words (syntax) and organization of their meanings (semantics, pragmatics, etc). The other governs how speech signals are related to words, a process often termed as "lexical access".

This thesis studies the Huttenlocher-Zue lexical access model, its implementation in a modern probabilistic speech recognition framework and its application to continuous speech from an open vocabulary. The Huttenlocher-Zue model advocates a two-pass lexical access paradigm. In the first pass, the lexicon is effectively pruned using broad linguistic constraints. In the original Huttenlocher-Zue model, the authors had proposed six linguistic features motivated by the manner of pronunciation. The first pass classifies speech signals into a sequence of linguistic features, and only words that match this sequence – the cohort – are activated. The second pass performs a detailed acoustic phonetic analysis *within* the cohort to decide the identity of the word. This model differs from the lexical access model nowadays commonly employed in speech recognizers where detailed acoustic phonetic analysis is performed directly and lexical items are retrieved in one pass.

The thesis first studies the implementation issues of the Huttenlocher-Zue model. A number of extensions to the original proposal are made to take advantage of the existing facilities of a probabilistic, graph-based recognition framework and, more importantly, to model the broad linguistic features in a data-driven approach. First, we analyze speech signals along the two diagonal dimensions of manner and place of articulation, rather than the manner dimension alone. Secondly, we adopt a set of feature-based landmarks optimized for data-driven modeling as the basic recognition units, and Gaussian mixture models are trained for these units. We explore information fusion techniques to integrate constraints from both the manner and place dimensions, as well as examining how to integrate constraints from the feature-based first pass with the second pass of detailed acoustic phonetic analysis. Our experiments on a large-vocabulary isolated word recognition task show that, while constraints from each individual feature dimension provide only limited help in this lexical access model, the utilization of both dimensions and information fusion techniques leads to significant performance gain over a one-pass phonetic system.

The thesis then proposes to generalize the original Huttenlocher-Zue model, which limits itself to only isolated word tasks, to handle continuous speech. With continuous speech, the search space for both stages is infinite if all possible word sequences are allowed. We generalize the original cohort idea from the Huttenlocher-Zue proposal and use *the bag of words* of the N-best list of the first pass as cohorts for continuous speech. This approach transfers the constraints of broad linguistic features into a much reduced search space for the second stage. The thesis also studies how to recover from errors made by the first pass, which is not discussed in the original Huttenlocher-Zue proposal. In continuous speech recognition, a way of recovering from errors made in the first pass is vital to the performance of the over-all system. We find empirical evidence that such errors tend to occur around function words, possibly due to the lack of prominence, in meaning and henceforth in linguistic features, of such words. This thesis proposes an error-recovery mechanism based on empirical analysis on a development set for the two-pass lexical access model. Our experiments on a medium-sized, telephone-quality continuous speech recognition task achieve higher accuracy than a state-of-the-art one-pass baseline system.

The thesis applies the generalized two-pass lexical access model to the challenge of recognizing continuous speech from an open vocabulary. Telephony information query systems often need to deal with a large list of words that are not observed in the training data, for example the city names in a weather information query system. The large portion of vocabulary unseen in the training data – the open vocabulary – poses a serious data-sparseness problem to both acoustic and langauge modeling. A two-pass lexical access model provides a solution by activating a small cohort within the open vocabulary in the first pass, thus significantly reducing the data-sparseness problem. Also, the broad linguistic constraints in the first pass generalize better to unseen data compared to finer, context-dependent acoustic phonetic models. This thesis also studies a data-driven analysis of acoustic similarities among open vocabulary items. The results are used for recovering possible errors in the first pass. This approach demonstrates an advantage over a two-pass approach based on specific semantic constraints.

In summary, this thesis implements the original Huttenlocher-Zue two-pass lexical access model in a modern probabilistic speech recognition framework. This thesis also extends the original model to recognize continuous speech from an open vocabulary, with our two-stage model achieving a better performance than the baseline system. In the future, sub-lexical linguistic hierarchy constraints, such as syllables, can be introduced into this two-pass model to further improve the lexical access performance.

Thesis Supervisor: Stephanie Seneff
Title: Principle Research Scientist

Thesis Supervisor: Victor W. Zue
Title: Professor

Acknowledgments

I want to first thank my co-supervisors Dr. Stephanie Seneff and Prof. Victor Zue. I guess co-supervision is as difficult for the supervisors as it is for the student, yet Stephanie and Victor managed it so well! Stephanie is always ready to sit down with me and to work with me on the very details of my research. She is a source of inspiration, no matter what topic I am working on. Victor would always challenge me to think of the big picture (and to keep up with my schedule). I could not have accomplished this thesis without the guidance of my supervisors.

I also want to thank my thesis committee members Prof. Ken Stevens and Prof. Michael Collins. There is probably a bible for every field of study. But it is not always possible to have an author of one such bible to sit on a thesis committee. I am lucky enough to have Ken on my committee. Discussions with Ken are important in shaping up what is presented in this thesis. Discussion with Michael is like to have a different and fresh look at my own work. Michael is also instrumental when I was writing my thesis: he always encourages me to write for a greater audience.

I want to thank Dr. James Glass and everybody at the Spoken Language Systems group. My research was made much easier (and sometimes possible) thanks to the superb infrastructure we have at SLS. Chao Wang, Lee Hetherington, T.J. Hazen and Scott Cyphers have helped me a lot, along the way on various projects and problems. The graduate students at SLS are a vibrant group and an important part of my SLS experience. I want to thank them: Karen Livescu, Han Shu, Ed Filisko, Alex Park, Mitch Peabody, Ghinwa Choueiter and others, for making SLS a much more fun place to be.

My family has always been very, very supportive in all these years. I can not always see them because of the distance. But knowing that they are there if I need them is a warm enough feeling. At difficult times, they gave me peace and courage. Since they have long been wondering when I can finish my study, I hope they are now relieved to see me graduating. This thesis is dedicated to them.

To 汤水山，田菊香，蔡宏宇和汤熠

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Automatic speech recognition (ASR) refers to the process of recognizing speech by a computer system, as well as the computer systems that are capable of performing such a task. The input to an ASR system is human speech, either stored in the computer system or collected and fed to the ASR system on-the-fly. The output of an ASR system usually takes the form of a text transcription.

What makes ASR a very challenging problem, however, lies in the fact that speech exhibits great variability. The input speech to an ASR system is often a continuous signal that contains many variations and ambiguities. Many sources contribute to the ambiguity in speech. The meaning of a sentence can differ, under different contexts and possibly with different intonation patterns. A word can have multiple meanings and pronunciations. Different words share the same pronunciation (homophones). Demographical, educational, gender and age differences all contribute to so-called *inter-speaker* variations. However, even when one speaker utters the same word or sentence on different occasions, it can be pronounced differently. Such *intra-speaker* variations can also be due to different physical or emotional states of the speaker.

One important source of speech variability is the co-articulation effect. Speech production involves the coordinated movements of different parts of the speech production apparatus or the articulators. These articulators possess different degrees of freedom and different levels of inertia, but their movements are nonetheless mostly continuous. As a result, the speech waveform is a continuous signal and individual sound units are influenced by their surrounding sounds. The acoustic properties of a sound unit can vary significantly in different contexts [116, 102]. A sound unit, e.g., a *phoneme*, is rarely produced in isolation in speech communication. Rather, a sound unit is usually uttered in association with surrounding units. At the basic sound unit level, there is a very important source of speech variation, namely the co-articulation effect. Throughout this thesis, we loosely use the term "sound units" to refer to phone- or phoneme-sized units of speech sounds. For interested readers, [36] provides an extensive discussion on the distinction between phones and phonemes. However, we want to remind our readers that "sound units" in this thesis do not always correspond to phones. For example, the *plosive* part and the *closure* part of a stop consonant are usually considered as two distinct sound units in our work, because the acoustic properties of these two parts are quite different.

In this thesis, we study a very important problem for ASR, namely the lexical access problem. In this thesis, lexical access collectively refers to the process of retrieving lexical entries based on the input speech waveform. The lexical access process deals with speech variabilities in general, and co-articulation effects in particular. The performance of lexical access is in turn essential to the overall performance of an ASR system. The lexical access process utilizes various knowledge sources, or constraints, to accomplish its goal [19, 110, 66].

This thesis studies in particular a two-stage lexical access model [51, 52, 53, 98, 116, 119], which provides a promising solution to handle the co-articulation phenomena of human speech and to achieve a satisfactory performance. In fact, this thesis implements and applies this two-stage lexical access model in a modern, probabilistic speech recognition framework [37, 117]. This two-stage model makes explicit use of linguistic features and constraints, often neglected in current ASR systems, to improve ambiguity resolution [106, 107]. Specifically, we study two types of linguistic features in this thesis. The *manner of articulation* features describe the source property of the airstreams. The *place of articulation* features, on the other hand, describe how the airstreams are modulated as they pass through the constrictions formed by articulators. The manner and place features are chosen because they jointly define the basic identity of a speech unit. Individually, they are broader classes that can be modeled robustly at a lower cost. The spreading and assimilation of manner and place features among neighboring speech units are also the most important reason behind co-articulation efforts. A first stage recognizer, operating on these broad linguistic features, can produce a cohort in a fast-match procedure.

The remainder of this chapter is as follows: In Section 1.1 I briefly review psycholinguistic lexical access models. In Section 1.2, I review a two-stage computational model that may improve upon traditional lexical access and hence, the overall performance of ASR. In Sections 1.3 and 1.4, I review related work and introduce the SUMMIT [37] speech recognition framework, which is used extensively in experiments in this thesis. Finally, Section 1.5 lays out a road map to the remaining chapters of this thesis.

1.1 Psycholinguistic Models for Lexical Access

In this section, we review some psycholinguistic lexical access models. The aim is to gain insight into the human speech cognition process, as suggested by Lea [32, 66], rather than to provide a complete literature review or critique individual models. While researchers are still far from a full understanding of the details of the speech communication mechanism, there are nonetheless widely accepted theories that successfully explain some basic observations in psycholinguistic studies of speech production and perception.

It is commonly held that lexical access, or lexical processing, takes place as an important intermediate step in human speech perception. Words, or in general lexical entries, are stored in the long-term memory as acoustic-phonetic patterns, associated with higher level knowledge, including phonological, syntactic, semantic, and pragmatic properties. The aim of the lexical access stage is to activate words in the memory and retrieve the associated higher level properties so that they can be used for understanding. Ambiguity associated with speech variations is resolved mainly in this stage [32].

Constraints involved in this lexical access stage fall largely into two categories. The first category is representation. The speech perception apparatus, including the cochlea and the auditory nerves, transforms the mechanical air pressure variations of a waveform into temporal patterns of discharges in the auditory nerves. The auditory cortex then processes these temporal discharges and encodes them into acoustic-phonetic patterns. Representation constraints are imposed to identify and classify speech sounds and to activate lexical entries based on the stored acoustic-phonetic templates [101, 102, 70]. The second category includes contextual information, derived from higher level knowledge in the previous context. These phonological, syntactical and semantic constraints serve in part to narrow down the range of contact in acoustic-phonetic pattern matching. Linguists and psychologists have developed different models for the lexical access process. We will review the Logogen model [81], the Cohort Model [75, 74, 65] and the Trace Model [78, 23, 77].

1.1.1 The Logogen Model

Morton's *logogen* model [81] was developed for word recognition in speech communication as in reading. A logogen is an activation device for words. Each logogen is responsible for the activation of a particular word. The input to a logogen is:

1. Linguistic stimuli from sensory analysis

2. Contextual information

The output of a logogen is the response of the word, and all the semantic information related to the word, which is encoded in the logogen. The incoming information in the logogen models is often referred to as *features*. When a logogen matches a feature with an input, it increments a feature counter. When the counter rises above a *threshold*, the response of the associated word is made available.

The logogen system is *passive* in the sense that the operation of a logogen is completely independent of other logogen systems. In particular, no comparison is made with regard to the activation levels of other logogens. Because of the passive nature of the logogen systems, it is also assumed that feature counts in each logogen decay rapidly. Further, only the response of the first activated logogen is passed to the output buffer. Counts in the partially activated logogens are reset shortly because of the decay effect.

The logogen is also an *i*nteractive model in that the logogen monitors all available features, including both the high level semantic or syntactic contextual features and the low level sensory stimulus features. Finally, the activation threshold is determined by the frequency of each word. The logogen of a high frequency word has a lower threshold whereas the logogen of a low frequency word has a higher threshold and requires stronger contextual or sensory stimulation before it can be activated.

The logogen model provides an abstract scheme in which various constraints, e.g. sensory, contextual and frequency, are incorporated. While it can help to explain many observations on word recognition at the conceptual level, it provides no details on how these constraints really apply within a cohort of words.

1.1.2 The Cohort Model

The cohort model [75, 74, 65] makes an explicit distinction between the two stages in word recognition: an autonomous stage and an interactive stage. The first, autonomous stage uses only acoustic-phonetic knowledge, whereas in the second, interactive stage, both acoustic-phonetic and high level knowledge are used.

In the autonomous stage, the acoustic-phonetic information of the initial part of the input is used to activate all words that share the same word-initial pronunciation. The set of words that share a given initial pronunciation and is activated in the initial contact is called the *cohort*, or more specifically, the *word-initial cohort*. The cohort is only defined by the (initial) acoustic-phonetic information. At this stage, context information is only used to deactivate words and reduce the size of the cohort.

In the interactive stage, further acoustic-phonetic information as well as higher level, semantic or syntactic, information, is used to remove false candidates from the cohort. The cohort model is an active or competitive model in that word candidates in the cohort compete with one another. When there is only one word left in the cohort, with a satisfactory degree of activation, it is proposed as the output of the word recognition process.

A main limitation of the cohort model is the lack of an error recovery mechanism. Unlike the logogen model in which the activation set is constantly updating with the stimuli of new features, the *word initial cohort* is fixed after the autonomous stage. If an error is made during initial contact and the correct word is not included in the cohort, the interactive stage will not be able to recover from this error.

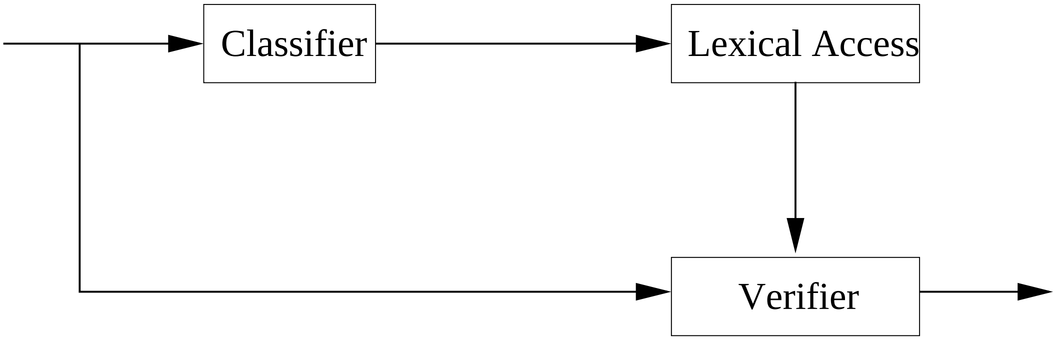
1.1.3 The Trace Model

The trace model [78, 23, 77] is a framework that integrates multiple simultaneous, and often mutual constraints to achieve the task of word recognition. Trace provides an interactive-activation mechanism that includes three levels of units: the *feature*, *phoneme* and *word* levels. Each unit corresponds to a particular perceptual object in time. The units are usually in a resting mode unless being activated by confirmatory inputs. If the activation level rises above a threshold, the unit is *a*ctivated.

The units are inter-connected by two types of links. There are links that transfer activation from lower level units to higher level units. These links are bi-directional and are excitatory. The second group of links connect units on the same level. These links are bi-directional and are only inhibitory. At the feature level, the feature units are banks of feature detectors. Each bank specializes in the detection of a particular dimension of the speech sound, such as vocalic, acuteness or diffuseness. Once a feature is detected and the corresponding unit activated, the activation is forwarded to the units at the phoneme level, via the excitatory links, to relevant phoneme detection units. In turn, an activation in the phoneme level will be processed by the units from the word level. There is a unit at the word level for each word. Connections between nodes at the same level are inhibitory. The activation of a phoneme, for example, would inhibit the activation of a competing phoneme. Finally, the trace framework – the entire network of units – self-updates along the time axis.

Trace is a model inspired in part by early ASR work [69]. The trace framework makes a great effort to deal with, in some sense, the “reality” of speech. Trace ac-

Figure 1-1: The Huttenlocher-Zue Two-Stage Model for Word Recognition



knowledges the continuity and the lack of clear boundaries in speech and, accordingly, makes no explicit segmentation of the speech signal. Rather, it makes implicit segmentation by assigning labels at different levels of the Trace network. Trace also assumes that the acoustic cues for each unit vary considerably with the context in which the unit appears. The Trace model allows such "lawful variability". The belief is that the perceptual system uses such information in word recognition. The context sensitivity is modelled in Trace by allowing the connections to be re-configured on-the-fly.

1.2 The Huttenlocher-Zue Model

In a series of papers [51, 52, 53, 98, 116, 119], Huttenlocher and Zue proposed a two-stage computational model for lexical access, which aims to incorporate more speech knowledge as constraints in word recognition. The model contains three modules: a classifier, which classifies the speech signal into broad phonetic classes; a lexical access module, which produces a cohort of candidate words based on the broad classification, and a verifier which performs detailed phonetic analysis to discriminate among the candidates. A scheme of the Huttenlocher-Zue model is shown in Figure 1-1.

Two studies at that time contributed directly to the development of this model. The two-stage continuously refining mechanism is inspired by early spectrogram reading experiments [16, 17, 118]; while the choice of broad phonetic classes is motivated by studies on phonotactic constraints [13, 98]. To better understand the Huttenlocher-

Zue model, we first review the most important findings that lay the foundations of this model.

The most encouraging finding of the spectrogram reading experiments is that the speech signal in general and the spectrogram representation in particular carries a lot more information for reliable phonetic identification than it was commonly believed at that time [62, 63, 88]. The spectrogram readers were able to locate almost all of the speech segments (above 97%), measured against transcriptions provided by phoneticians. Between 81% and 93% of the labels assigned by a spectrogram reader agreed with those provided by phonetic transcriptions. Most of the time the spectrogram readers assigned only one or two labels to each segment, further evidence that the spectrogram contains abundant and reliable acoustic cues.

Perhaps as important as these results are the dynamics of the spectrogram reading process revealed in these experiments. It has been noticed that spectrogram readers resorted to a myriad of acoustic cues in their efforts to label the spectrograms. However, it was observed that some acoustic cues are relatively easy to identify while, for others, their meanings and discriminative power become relevant only after the acoustic context is constructed from the previous cues. It is commonly held that the features used to distinguish speech segments may be organized in a hierarchical fashion [15]. The theory of *under-specification* further implies that the identification of certain acoustic cues precedes the identification of others [4], which is also taken into consideration in the Cohort [65] or the Trace models [78] discussed above. The fact that similar tactics are employed in dealing with the visual representations of speech suggests that it is computationally feasible to construct a reliable lexical access model in a multi-stage fashion.

The Huttenlocher-Zue model thus proposes a two-stage approach to solve the lexical access problem. In the first stage, a coarse classifier classifies speech segments into broad phonetic classes based on those acoustic cues that can be reliably estimated and can serve to establish acoustic-phonetic contexts for those under-specified acoustic cues. The broad phonetic classes suggested in the original proposal [52] are based on manner of articulation, as listed in Table 1.1. These manner-based classes are chosen because they are good invariant measurements across different speakers and/or in different phonetic contexts. In the Huttenlocher-Zue model, words whose pronunciation shares the same broad phonetic description form a cohort, an equivalent class for the lexical access purpose.

Broad classes:	vowel, nasal, stop, liquid or glide, strong fricative, weak fricative

Table 1.1: Six broad phonetic classes proposed in the Huttenlocher-Zue Model.

The quality of the cohort is the key to the success of this two-stage model. Here the authors drew on phonotactic constraint analysis to gauge the effectiveness of their proposal. In his seminal work [13], Church demonstrated how a phone sequence can be parsed into syllables and other supra-segmental units, using cues from allophonic variations. Church further argued that lexical retrieval consisted of a phonological parsing step and a matching step [14]. In the Huttenlocher-Zue model, the authors showed that, with a representation based on six broad phonetic classes, the lexical access module can retrieve a cohort about 0.1% of the original lexicon size. The analysis was done on the $20,000$-word Merriam Pocket Dictionary (MPD). Words in MDP are sorted into classes based on their pronunciations, represented by the six broad phonetic categories. These word classes would be the cohorts in a word recognition experiment on MPD. Some descriptive statistics are outlined in Table 1.2. On average, a cohort contains only 22 words (34 words if adjusted by word frequency), compared to the original size of $20,000$ words of MDP. While this is encouraging, we also note that it assumes no errors in broad classification.

	MPD (Equally Weighted)	MPD (Frequency Weighted)
Expected Class Size	22	34
Median Class Size	4	25
Maximum Class Size	223	223
% of Unique Classes	32	6

Table 1.2: Representing Words by Categories in the Huttenlocher-Zue model.

1.2.1 Early Implementations of the Huttenlocher-Zue Model

There were only limited efforts to implement the Huttenlocher-Zue model. The original proposal contained only the descriptions of a rule-based broad classifier while leaving most details out – most notably the lack of a discussion on the *v*erifier module [52]. Although the proposal was developed with isolated word recognition tasks in mind, the first implementation of the model was on continuous digit strings [10] where the limited number of words made lexical access even easier. The verifier was still missing in this work. It was in Italian that full-fledged, end-to-end speech recognition systems were built following the Huttenlocher-Zue model, for both isolated words and continuous speech [28, 29, 30].

Broad Classification

The original proposal outlined a method for acoustic segmentation and broad phonetic categorization. The acoustic signal is passed through a filter bank, and total energy is computed for each band every 5 milliseconds, using a 25-millisecond window. The energy contour was computed via step-wise approximation. This step transforms the continuous speech signal into a sequence of acoustic events characterized by relatively stable energy levels. The energy levels are further parameterized in terms of *magnitude*, i.e. `LOW, MEDIUM, HIGH`, and *relative magnitude*, i.e. `PEAK, VALLEY, ONSET, OFFSET`.

Parameterized energy levels of each band are combined for acoustic segmentation using predicates. This step not only segments the speech signal but also determines important acoustic cues for each segment, such as *voicing*. Finally, broad phonetic categorization is accomplished by a set of rules operating on the segmentation and the detected acoustic cues. These rules fire only when the input matches, in a non-competing fashion similar to the Logogen or Trace model, and can result in overlapped segmentations [52].

Recognition on Continuous Digit Strings

The experiments on continuous digit strings [10] can be considered the first attempt to implement the Huttenlocher-Zue model, although the experiments themselves are rather preliminary. A slightly different set of broad phonetic categories was used in these experiments: `sonorant, intervocalic nasal, stop, strong fricative, short voiced obstruent, vowel` and `weak fricative`. These categories are chosen so as to take advantage of some of the phonotactic constraints used in [13]. This extension enables the lexical access module to account for alternative pronunciations of each digit. The acoustic segmentation and broad categorization were done in a way similar to the original proposal, with segment overlapping and gapping allowed.

The lexical access module explicitly allowed alternative pronunciations of different digits. However, it undertook a "brute force" search scheme by matching all possible pronunciations against *any* possible subsequences of the broad classification output. The word-lattice is pruned by removing incomplete paths and by matching the realized pronunciation of each word with its context. These preliminary experiments did not implement the second, verification stage. Rather, system performance was measured by checking whether the true digit string survived in the remaining lattice.

Despite the inconclusiveness, these early experiments addressed the issue of speech variations and in particular, pronunciation alternatives. It also ventured into the recognition of continuous speech, albeit a search scheme more efficient than the brute-force search would surely be necessary for a more complicated domain.

Recognition Experiments in Italian

In [28, 29, 30], the authors compared two-stage speech recognition systems with one-stage, HMM-based systems, on isolated-word as well as continuous-speech tasks of Italian.

An HMM recognizer is also used in the two-stage system as the verifier in the second-stage, making the results of the one-stage and two-stage systems directly comparable. The first stage and the second stage, as well as the single stage system, also share the same feature extraction pre-processor. The differences in accuracy and computational load are attributed to the two-stage configuration.

In these experiments, broad phonetic classification is done *before* segmentation. In fact, mel cepstral feature vectors for 10 millisecond frames are classified into broad phonetic categories. Neighboring frames of the same category label are clustered into segments. This approach differs from the Huttenlocher-Zue proposal in that it does not require a separate segmentation step.

The lexical access module uses a dynamic programming procedure to search for a cohort. Words in the lexicon are organized into tree-like structures, with each node corresponding to a broad phonetic class. The terminal nodes of the trees point to a cohort of words. The dynamic programming procedure matches the input broad phonetic class lattice against the tree structures to find the candidate words.

The second stage verifier uses HMM recognizers which are tightly integrated with the first stage. The HMM recognizer expands only on words from the lexical access output and in the case of continuous speech, the word sequence, licensed by the lexical access module. In the second stage, the HMM recognizer discarded the segmentation generated in the first stage, as the authors claimed that such segmentations are of relatively poor quality due to the coarseness of the broad classification. Compared with a single-stage system, the final performance of the two-stage approach is slightly worse than a single-stage system. The authors attributed this degradation in performance to the errors propagated from the first stage.

1.2.2 Comments on the Huttenlocher-Zue Model

In our opinion, the Huttenlocher-Zue model provides an elegant proposal for solving the lexical access problem in ASR. The two most important characteristics of the Huttenlocher-Zue model, integration of multiple knowledge sources (constraints) and successive refining through multiple stages, now are widely accepted in psycholinguistic models for speech perception by a *human*. In this regard, the Huttenlocher-Zue model was truly visionary at the time.

Prior to the Huttenlocher-Zue model, Dennis Klatt had proposed an analysis-by-synthesis model for word verification in speech understanding systems [61]. In the HWIM system [25], this verifier was actually utilized to verify word candidates derived from a phonetic lattice.

On the other hand, the early efforts in implementing this model clearly met with only limited success, and a general adoption of this model has been lacking. We examine a few possible reasons.

Most notably, details on the verifier are missing from the original proposal. As the entire framework emphasizes an "analysis-by-synthesis" approach, one could imagine that the verifier will examine the under-specified acoustic cues, given the cohort as a context.

Secondly, the performance of this model is quite sensitive to speech segmentation. Unfortunately, speech segmentation is notoriously difficult even to this day.

Thirdly, the model lacks an error-recovery mechanism. To be fair, a fuzzy matching scheme that allows insertion, deletion and substitution of segments has been mentioned in [116] to alleviate such problems. In reality, fuzzy matching, like the dynamic programming scheme implemented in [28, 29, 30] has yet to achieve a satisfactory performance.

In hindsight, the rule-based discriminative approach employed in the original proposal for acoustic segmentation and categorization also made this model difficult to generalize, to handle large corpora or to integrate with a second-stage verifier.

1.3 Modeling Linguistic Features in ASR

The efforts to model linguistic features in speech recognition systems date back to some of the earliest research initiatives in this field. The methodologies have evolved from the early, expert-system approaches [5, 24, 25, 73, 113] to the more recent probabilistic approaches [22, 59, 91, 105].

1.3.1 Early Knowledge-based Systems

In [113] the authors proposed the concept of *acoustic-phonetic elements*, or *APELs*, which are phoneme-sized units. These APELs differ from the traditional phonemes in that they are only partially-specified. In other words, only some of the acoustic-phonetic properties of an APEL is specified. A phoneme, depending on the situation, can be one or more of several possible APELs.

The system in [113] recognizes speech by identifying APELs. It takes a two-step approach. The first step, or the preliminary segmentation and segment classification step, segments continuous speech into units. The units are classified into vowels, fricatives, stops and volume dip within vowels by a decision tree, using a mixture of filter-bank energy, derivatives and ratios.

The second step provides detailed classification on each of the units found in the first step. The second step employs four special-purpose classifiers. A detection and classification module detects and classifies some diphthongs, semivowels and nasals. A vowel identifier proposes three ranked choices of vowels. A fricative identifier and a stop identifier, classify fricative- and stop-like segments. These special-purpose classifiers resort to different acoustic cues and employ different classification strategies.

The output of [113] is a sequence of APELs, each of which corresponds to one or more possible phonemes, which may be fed into phonological parsing and language understanding systems, for example, for the recognition of words. The system performs well on a small test corpus. For example, the authors reported sentence recognition on a small number of sentences from a 250-word vocabulary.

The systems in [5, 25, 111, 24] have all taken a multi-pass strategy in modeling acoustic phonetic features. In particular, they all attempt to build a segmentation network and identify some aspects of the linguistic features of each segment before establishing the phonemic identity of each segment. We will focus on the SDC speech understanding system [5]. This system is made up of two components: the acoustic-phonetic processor, which attempts to label the vowel-sonorant areas that can be reliably handled. The acoustic-phonetic processor has a three-level labeling system: phoneme, feature and rough labels. It assigns the labels in a multi-pass, continuously refining manner. First it assigns vowel and sonorant labels to "reliable" areas, using a formant-based distance measure. It then proceeds to segment speech into phone-sized units, using rate-of-change, continuity and duration measurements. It matches LPC spectra against pre-computed templates to determine manner and place of articulation. It determines fricatives and plosives based on the manner and place identification. For regions where the acoustic evidence suffices, it attempts to further assign phoneme identities.

Based on these three-level labelings, the acoustic phonetic processor calculates prosodic information and identifies syllable and phrase boundaries. Finally, it uses the prosodic information to smooth the labels.

Among the systems developed for the ARPA Speech Understanding Research (SUR) project, the HARPY system in [73] differs notably from the other systems [5, 25, 111, 24, 113]. In [73] the lexical networks were generated using manner-based phonological rules.

1.3.2 Modeling Features with Neural Networks

The work in [59, 60, 58, 57, 31] models acoustic and articulatory features using neural networks (NNs) or multi-layer perceptrons (MLPs) in a connectionist recognition framework [9]. Whereas a traditional hybrid MLP/HMM speech recognition system uses MLPs for *phone* classification, in [59, 60, 58, 57, 31] MLPs are used for *feature* classification.

In [58, 57, 31], the researchers construct three comparable systems, corresponding to Chomsky-Halle SPE binary features [11], multi-value features and Harris' Governmental Phonology [47]. The aim of this research is to examine whether acoustic and articulatory features can be reliably estimated and provide useful information for a speech recognizer. Results on the TIMIT [35] database show a phone recognition accuracy of about 50% to 60%, with the multi-valued feature models achieving the highest accuracy of 63.5%, compared to an HMM-based baseline of 63.3%.

In [59, 60], the authors use MLPs to identify a set of multi-valued pseudo-articulatory features. The authors show improved performance of the articulatory feature system in noisy environments. They also explore different information fusion techniques that aim to combine articulatory information with acoustic models based on phones.

The hybrid MLP/HMM framework used in these systems requires that the acoustic observations be absorbed at a fixed rate by HMM states. These systems also take the view that a phone or a phoneme can be mapped deterministically to a feature-bundle, although, for decoding different levels of asynchrony are allowed. As the HMMs are organized as phone-sized units, essentially these systems are modeling broad classes of phonetic units organized at parallel feature dimensions.

1.3.3 Overlapping Features

Whereas the hybrid MLP/HMM systems just discussed employs a fixed-rate HMM framework, the overlapping features system [105] differs in that it uses a variable rate HMM framework.

In the overlapping features system, the recognizer takes the view of non-linear phonology and rewrites phonemes into feature bundles. However, unlike the systems in [59, 60, 58, 57, 31] where each feature tier is then being modeled separately, the recognizer seeks to compile all linguistic constraints, including higher level prosodic constraints as well as phonological constraints on top of the feature bundle streams. These higher level constraints mainly govern how features spread and assimilate, i.e. features extend their presence in neighboring sound units, and thus create an *overlapping* feature bundle representation. The recognizer then enumerates all possible configurations of feature bundles and treats them as the basic state variables in an asynchronous HMM system to model.

The authors show the distinct feature bundles to be an appealing set of units for acoustic modeling. The number of distinct feature bundles is significantly less than the number of context-dependent phonetic units one would encounter. Improvements for both phone recognition and word recognition are reported on the TIMIT database.

1.3.4 Hidden Articulators and Dynamic Bayesian Nets

There is now burgeoning research momentum in the field of Dynamic Bayesian Nets [54, 100, 120, 72, 114], which in some sense can be traced back to hidden articulatory Markov chains [91]. The hidden-articulator Markov models (HAMMs) provide another way of dealing with multiple feature streams in an HMM framework. HAMMs decode on the product HMM of the underlying, feature-based HMMs, which allows greater flexibility and asynchrony among features.

Dynamic Bayesian nets (DBNs) provide a more general framework for dealing with the inter-dependencies of different feature tiers. In DBN, features are also modeled at different tiers. Unlike in the Markov chains where time-dependency (and independency) is explicitly modeled, DBNs model the conditional distribution rather than time-dependencies. Further, the feature tiers (states) are observable, whereas in HAMM the articulatory states are unobservable. However, DBNs are in general time-consuming to train and to decode and so far have only seen experimental applications on controlled tasks such as digit strings or isolated words.

The hybrid MLP/HMM systems, HAMMs and DBNs, and to a lesser extent the overlapping feature systems, put more emphasis on the development of a statistical framework to handle the challenges resulting from multiple feature streams. It still remains unclear as to how to model the inter-tier dependencies, or the feature spreading and assimilation effects effectively. The hybrid MLP/HMM systems leave the inter-dependency to the higher level HMM. It has been observed that the recognizer may propose phonetically implausible feature alignments, which negatively affect system performance. HAMMs use a product HMM to explicitly model feature asynchrony. However, this results in a huge state product space. The overlapping feature system and DBNs, which allow system designers to incorporate more linguistic knowledge in choosing the model topology, are more powerful.

However, recent efforts contrast sharply with the analysis-by-synthesis model proposed in [103] and the paradigm adopted in the early systems. While these data-driven, generative models have a clear advantage in handling large training corpra, it is probably worth investigating speech recognition systems taking a discriminative approach in modeling linguistic features.

1.4 The SUMMIT Probabilistic Recognition Framework

In this thesis, we use the SUMMIT speech recognition framework for our experiments [37, 39]. SUMMIT incorporates four different knowledge sources into a speech recognizer: acoustic models, phonological knowledge, lexical constraints and language models.

SUMMIT uses landmark-based acoustic modeling [38]. A heuristic-based landmark detector identifies landmarks by looking for significant acoustic movements in the speech signal. Around these landmarks, feature vectors based on Mel frequency cepstral coefficients and derivatives are constructed. Usually, there are two types of landmarks modeled in SUMMIT. The most common type corresponds to significant acoustic movements between phonetic units, or the transitional landmarks. The other category corresponds to significant acoustic movements within phones, or the internal landmarks. Both types of landmarks are modeled by Gaussian mixture models.

Phonological knowledge in SUMMIT is represented as rules that parse phoneme sequences and propose a sequence of zero or more candidate context-dependent realizations – phones – for each phoneme. Lexical constraints are simply the lexicon and the baseform pronunciations for each word in the lexicon. Finally, SUMMIT uses a class n-gram for language modeling.

The knowledge sources are pre-compiled and stored as a finite state transducer [49] for efficient decoding. In SUMMIT, the construction process involves the composition of four FSTs, each representing a knowledge source discussed above:

$$R = C \cdot P \cdot L \cdot G \tag{1.1}$$

G represents the language model constraints and is usually in the form of class *n*-grams.

L refers to the lexicon, which provides pronunciations, in phoneme, for all the lexical items that may show up in language model G. Below is a sample entry from the lexicon which provides the pronunciation of the word *boston*.

```
boston                                  : b ( aa | ao ) s t ax n
```

P is a collection of manually designed phonological rules. These rules, in the form of context-dependent grammars, govern how phonemes – from the lexicon – are realized as context-dependent phones.

Finally, C maps context-dependent phones to diphones, which are the basic acoustic units modeled in SUMMIT.

The acoustic models in SUMMIT are Gaussian mixture models (GMM) trained with acoustic observations of diphones from a training database.

During recognition, the acoustic measurements, computed for the input speech, are modeled by the acoustic models. The Gaussian mixtures in the acoustic models hypothesize diphone identity based on the acoustic measurements. Based on the hypothesis from the acoustic models, SUMMIT then searches the FST, a process in which the acoustic information is combined with all the higher level constraints precompiled into the FST. The outcome of the search is a hypothesized word sequence.

1.5 Overview of the Thesis

In this thesis we revisit the Huttenlocher-Zue model and explore new methods of implementing this model in a state-of-the-art, probabilistic speech recognition framework. This thesis focuses on three aspects: the choice of basic recognition units, the extension to continuous speech recognition, and error-correction mechanisms.

The original Huttenlocher-Zue model uses six broad phonetic categories as the basic recognition units for the lexical access module. These broad classes are chosen mainly for their invariability across different contexts and among different speakers/dialects. In addition to manner-based features, we also study place-based broad classes. Place-related acoustic cues are also important in speech perception [79] and are sometime easier to capture than manner-related cues, such as voicing. We further optimize basic recognition units, based on the two sets of broad classes of manner and place of articulation, on a large training corpus, to achieve a better balance between model compactness and discriminative power.

This thesis extends the original proposal to handle continuous speech over both a medium-sized vocabulary and an open vocabulary. We study comparatively how the lexical access module functions in an isolated word task and in a continuous speech task, which leads to a new way of defining cohorts and the protocol between the lexical access module and the verifier.

The probabilistic framework in which we implement the two-stage model supports insertion, deletion and substitution of broad phonetic classes. This embedded error-recovery mechanism suffices for isolated word tasks. We develop data-driven error-recovery methods when we extend the original model to continuous speech.

The thesis is organized as follows: in Chapter 2 we discuss the choice of broad phonetic classes and basic recognition units, as well as techniques to model these features in a first-stage lexical access module in a probabilistic framework. We construct a full two-stage recognizer and study methods to integrate constraints from the first stage into the later, verifier stage. Our conclusion, drawn from a detailed analysis of experimental results on an isolated word task, is that performance of a two-stage recognition system is competitive compared against single-stage systems. In Chapter 3, we extend the original Huttenlocher-Zue model to handle continuous speech. We provide a natural generalization of the original definition of "cohort" in the continuous speech context, which overcomes some of the limitations observed in [28, 29]. We also carry out an analysis of the errors that tend to propagate from the first stage, and propose an empirical solution. Our experiments on two telephone-quality, medium-sized vocabulary continuous speech recognition tasks show some successes of this two-stage approach. We then proceed to extend this model to the challenging task of recognizing continuous speech from an open vocabulary in Chapter 4. We use the broad-class-based lexical access to prune a large set of semantically homogeneous lexical items rarely observed during training. Experiments show our approach more effective than some other methods proposed to solve similar problems. Finally, we provide a general discussion of these results and some on-going research efforts in Chapter 5.

Chapter 2

Modeling Linguistic Features in a Two-Stage Recognizer

In this chapter, we first study candidates that can be used as the basic recognition units for the lexical access module in a two-stage recognition framework. The original Huttenlocher-Zue proposal made a strong case of using six broad phonetic classes based on manner of articulation, a guidance which this thesis follows. However, this thesis studies a broader set of linguistic features of both manner and place of articulation. Whereas the original broad classes are motivated by a belief in their invariability across different contexts/speakers, this thesis fine tunes the set of feature-based recognition units, as inspired by linguistic knowledge, over a training corpus, to balance between model compactness and discriminative effectiveness.

In this thesis, we choose to model *both* manner and place of articulation in the first stage and we organize all speech units along these two orthogonal dimensions. We believe that manner and place features are natural organizational dimensions along which the speech units can be organized. This view is supported by our observations that, in automatic clustering analysis of speech units [67, 115, 38], the clusters naturally fall along both the manner and place dimensions. We also note that, in a data-driven clustering analysis, the manner and place dimensions compete for acoustic observations. The clustering results are usually sensitive to the acoustic representation and the number of final clusters. As a result, the alignment of the clusters is usually a mixture of both manner and place dimensions, with a majority of clusters aligned along the manner dimension. Instead of having the manner and place dimensions compete for speech units, we formally divide speech units into two groups of clusters along the two feature dimensions. The acoustic models hence derived reliably and consistently represent the constraints from *both* dimensions of manner and place of articulation. There is another challenge of modeling linguistic features associated with place of articulation, which has probably impeded its adoption as an organizational class. Traditionally place of articulation is handled differently for vowels and consonants. In [103], for example, the authors make distinctions between *articulator-free* features and *articulator-bound* features, with *round* being the only articulator-bound feature shared by vowels and consonants. Such an approach, however, would imply an embedded distinction of *manner* in modeling place of articulation. Instead, we coerce the vowel units into the traditional place space for consonants. Our experiments show that, while each feature dimension alone can provide powerful constraints already, information fusion techniques that can combine constraints from both feature dimensions provides much more effective constraints for the first stage. These techniques are discussed in detail in this chapter.

Finally, we develop a two-stage speech recognizer for an isolated word task. We analyze the performance of the cohort generated by the first stage, using manner or place features and with different information fusion techniques. The final two-stage system, using a state-of-the-art phone-based recognizer at the second stage, has achieved improved performance on the isolated word task, when compared to a state-of-the-art phone-based recognizer.

2.1 Linguistic Features

In this thesis we adopt the definition of linguistic features as the properties that can jointly define a sound unit, most importantly, a phoneme. Commonly studied linguistic features include manner of articulation, place of articulation and voicing. The phoneme `/t/`, for example, has the manner of articulation of a *stop* sound, the place of articulation of *alveolar* and is *voiceless.* In fact, in English, these three properties suffice to define the underlying phoneme of /t/.

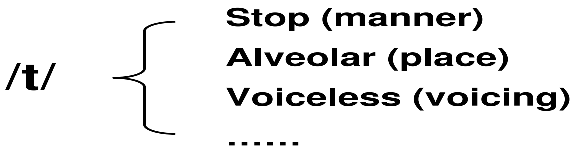


Figure 2-1: Linguistic features are defined as properties that can jointly define a phoneme. In this example, the phoneme /t/ is defined by its manner of articulation, its place of articulation, and voicing properties.

In this thesis we study manner and place of articulation features as constraints for lexical access. According to non-linear phonology, the speech stream can be viewed as a sequence of "feature bundles." organized along auto-segmental tiers [103, 40]. Manner and place of articulation are two classes of the auto-segmental features, grouped together in part based on their roles in phonological rules [94]. They are attractive as classes because members of the same manner/place class usually share common acoustic properties. We collectively refer to manner and place of articulation as *linguistic features.*

Manner and place features have strong articulatory and auditory correlates. Manner of articulation, in general, corresponds to the nature of the airflow in the course of speech production. Place of articulation corresponds roughly to the major constriction points of the vocal tract through which the airflow is modulated. In the source-channel model of speech production, manner and place of articulation describe the first order properties of the airflow [103, 26]. As noted in the original Huttenlocher-Zue model, the manner features of speech demonstrate excellent invariability and can be reliably measured by energy-based segmental measurements. Place features, on the other hand, often reveal themselves as time-varying patterns in spectral analysis of speech signal. They can also be measured reliably, albeit with more complicated acoustic measurements typically associated with landmarks [37]. Due to the auditory correlate of linguistic features, it is hypothesized that the human speech perception apparatus has evolved to be well-adapted to the perception of acoustic cues associated with these features.

When we organize the speech units along the auto-segmental tiers of manner and place of articulation, it becomes much easier to describe the co-articulation effects and associated speech variations. Speech production involves the coordinated efforts of a variety of articulators, with different patterns of motion and varying degrees of inertia. However for understanding purposes it suffices to consider speech as a sequence of phonemes, each defined by a distinct set of linguistic features. At the perception level, speech is a continuous, time-varying signal. The features that define each speech unit, e.g., phoneme, rarely change discretely, unless the corresponding change in underlying configurations of articulators can be accomplished abruptly. Most of the time, the articulators move in continuous re-configuration. The speech signal, or the features that characterize the speech units, change in a way that reflects this continuity. The actual realization of each phoneme, a phone, often shows characteristics of its local context. This phenomenon, commonly known as *co-articulation*, is common in speech and is a major source of speech variability. Co-articulation can be easily explained and modeled by feature spreading and assimilation [94, 21, 40, 64]. As such, the speech signal can sometimes be more naturally represented with feature tiers, as shown in Figure 2-2.

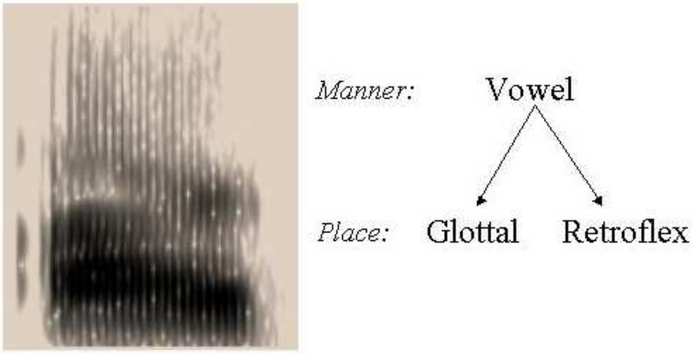


Figure 2-2: An /ɑ/ + /r/ sequence in a stressed syllable, e.g. as in *car*, usually demonstrates a strong co-articulation effect. The vowel /ɑ/ becomes retroflexed, and the semi-vowel /r/ is more vowel-like. While it is difficult to distinguish the boundary between the two phonemes, the entire segment can be naturally described in terms of manner, vowel, and a transition of place of articulation from glottal to retroflex.

The linguistic features are powerful constraints when we seek to distinguish lexical items. Features are believed to activate the stored acoustic-phonetic patterns associated with lexical items. Features as well as feature spreading/assimiliation provide useful information for recognizing syllables and words, a view shared by the original Huttenlocher-Zue proposal and many other researchers [13, 90]. Glottalization of /*t*/, for example, usually happens at the end of a syllable, between a preceding vowel and a following stop or nasal, as in *button*. In [13, 90], for example, comprehensive rules have been developed which govern syllable structure.

Lastly, empirical analysis of speech data supports the view that manner and place are natural organizational dimensions of speech. In [38], a bottom-up data-driven analysis was carried out seeking regularities of speech units, as characterized by spectral vectors. A ten-class clustering result is reproduced in Table 2.1. The results can largely be explained by the manner and place of articulation features. For example, Cluster 1 contains front vowel and semi-vowels, or diphthongs ending with a front position. Most clusters are manner-driven, which again reflects the prominence of manner features, especially when the speech units are represented by spectral coefficients. Place features nonetheless manifest themselves, for example, as the *palatal* cluster in Cluster 10.

Cluster		**Phones**
1	:	i^{y} ɪ e^{y} ɨ j ü
2	:	o^{w} ɔ ə u^{w} l̩ l w
3	:	ɚ ɝ r
4	:	ʊ ɔ^{j} ʌ ɑ^{w} ɑ ɑ^{y} ɛ æ
5	:	b̚ d̚ g̚ p̚ t̚ k̚ v
6	:	m n ŋ m̩ ŋ̩ n̩
7	:	ʔ ɾ θ ð f
8	:	t d b p h k g
9	:	z s
10	:	č ǰ š ž

Table 2.1: Clusters derived through a data-driven approach from [38].

Similar phenomena re-surface in clustering analysis of basic recognition units in many phone-based speech recognition systems. As discussed earlier, in a phone-based speech recognizer, significant performance improvements can be realized by utilizing context-dependent models to capture the acoustic variabilities of these sub-word units. But such performance gains are achieved with a significant cost. The more elaborate these units are, the more severe the computation and storage demands are during training/testing, and, perhaps more importantly, the more data are needed to train these models, in order to avoid sparse data problems. To alleviate such problems, researchers have often adopted a data-driven approach of first spawning a large number of highly-specific context-dependent units (e.g., tri-phones), and then combining units with similar context using automatic clustering techniques [67, 115]. This has resulted in more robust models, especially for rare combinations, and subsequently better overall performance. When one examines the outputs of the automatic clustering algorithms, it is often the case that members of a cluster fall along natural linguistic dimensions such as manner or place of articulation. For instance, the following diphone cluster was generated by a decision-tree-based clustering process [112]:

č|t̚ š|t̚ ǰ|t̚ ž|t̚

This cluster contains four di-phones between consonants and the voiceless closure (/t̚/). The left contexts in this cluster, namely /š/, /č/, /ǰ/, and /ž/, all share the same place of articulation, *palatal*

The feature system adopted in this thesis differs from Chomsky and Halle's "distinctive features" system [11] in the sense that we consider multi-valued features of manner and place, whereas the "distinctive features" system allows only binary-valued features. A feature in the distinctive features system often corresponds to a *feature value* in our system. The distinctive features system is very powerful to analyze speech sounds in a top-down fashion. In our study, we use multi-valued features to facilitate a bottom-up modeling of speech sounds in recognition. We take into account practical engineering considerations when we design the feature system for this study. Most importantly, the multi-valued feature system can be easily incorporated into a probabilistic speech recognition framework.

2.1.1 Manner of Articulation

Manner of articulation describes primarily the nature of air flow, or the source, of speech production, with the exception of the nasal manner which accounts for a very particular configuration of the vocal tract. In this thesis, our treatment of the manner classes is somewhat unconventional: they are empirically chosen based on their relative acoustic differences. Thus, the vowel class is divided into three subclasses: `vowel`, `schwa` and `diphthong`. The decision is based on energy, duration and dynamic movements. The `schwa` class is shorter, less prominent acoustically and more susceptible to contextual influences. The `diphthong`, comparably, is longer in duration than ordinary `vowel` and features dynamic movements of formants unusual for `vowel`. Table 2.2 lists the manner assignments adopted in this thesis.

Manner		**Phones**
Schwa	:	ə ɨ ɚ
Vowel	:	ʌ ɛ ɪ ʊ ɑ æ ɔ ɝ iʲ uʷ
Diphthong	:	eʲ ɑʲ ɔʲ ɑʷ oʷ
Semi-Vowel	:	w j r l l̩
Plosive	:	b d g p t k
Closure	:	b̚ d̚ g̚ p̚ t̚ k̚ ʔ ɾ
Fricative	:	f v θ ð s š z ž h
Affricate	:	č ǰ
Nasal	:	m n ŋ m̩ n̩

Table 2.2: Manner assignment to segments used in this research.

2.1.2 Place of Articulation

Place of articulation focuses on the configuration of the vocal tract, in particular the place of constriction, in speech production. In other words, place of articulation focuses on the effects of the vocal tract filter encoded in the speech signal.

Place of articulation was missing in the original Huttenlocher-Zue proposal. This is probably because the energy contours proposed in [52] for acoustic segmentation would perform relatively poorly for place classification. When the acoustic representations are no longer limited to energy contours, place of articulation soon emerges as a major organizational dimension for the speech signal, as exemplified by the cases cited in earlier discussions. Our experiments to be discussed later also suggest data that supports using place of articulation as an organizational dimension.

One of the barriers to using place of articulation features for speech recognition lies in the complexity surrounding it: traditionally two different place systems have been defined for consonants and vowels. For consonants, the place of articulation has been defined to be the location of the maximum constrictions in the vocal tract during pronunciation, such as *palatal*. The place of articulation for vowels has been traditionally defined based on *tongue position* and *lip rounding*. This makes it difficult to define a set of organizational classes that can be used across the full set of phonetic units. We could, for example, simply combine the two systems when modeling place of articulation, as in [22, 59]. However, this would imply a vowel/consonant distinction in modeling place of articulation, which then adds complexity to the model space where manner class dictates place assignment.

As a working hypothesis for simplifying the modeling requirements, we decided to group all sounds into the *same* set of place-based features. Intuitively, /i^{y}/ and /y/ are so similar that a "palatal" place for /i^{y}/ is well-motivated. Similar relationships hold between /ɝ/ and /r/, /u^{w}/ and /w/, and, arguably, between /ɔ/ and /l/. A place assignment for other vowels is less clear, but in the interest of simplicity, we have coerced all vowels and diphthongs to be organized into the *same* place of articulation classes as the consonants. We are using eight distinct place class assignments, as listed in Table 2.3. We realize that our choices cannot be fully justified on the basis of linguistic theory, but we have nonetheless adopted the position that this is a reasonable first step, and that empirical results will ultimately guide us to further refinement.

Diphthongs are slightly more complex because they involve a transition between two different places. Our place assignments for diphthongs are given in Table 2.4.

Place		**Phones**
Alveolar	:	ɨ ɪ n̩ n s z t t̚ d d̚ ɾ
Dental	:	ð θ
Open	:	ɑ ʌ ə æ h ʔ
Labial	:	ʊ uʷ m m̩ f v w p p̚ b b̚
Lateral	:	ɔ l l̩
Palatal	:	iʲ j š ǰ ž č
Retroflex	:	ɝ ɚ r
Velar	:	ɛ ŋ k k̚ g g̚

Table 2.3: Place of articulation assignments adopted for this research.

Phone		**Place**
ɑʲ	:	Open → Alveolar
eʲ	:	Velar → Alveolar
ɑʷ	:	Open → Labial
oʷ	:	Lateral → Labial
ɔʲ	;	Lateral → Alveolar

Table 2.4: Place transition for diphthongs as defined in this research.

2.1.3 Modeling

We implement the first-stage recognizer, using the broad manner and place classes, using the SUMMIT [37]speech recognizer. SUMMIT provides a probabilistic framework to decode graph-based observations and uses a landmark-based modeling scheme. There are two types of landmarks: those corresponding to segment transitions and those that are segment-internal. The segments are traditionally phonetic units. Figure 2-3 illustrates how landmarks are modeled in SUMMIT [1].

In our experiments, the lack of a feature-based corpus forces us to seek ways to model feature-based landmarks using the available phonetically-transcribed data. We dictate manner and place assignments to phonetic units, as outlined in Table 2.2 and Table 2.3. Figure 2-4 illustrates how the feature-based landmarks are derived from the corresponding phone-based landmarks. Feature vectors are constructed around these landmarks and then used to train Gaussian Mixture Models (GMM) for acoustic

[1] The author thanks Jim Glass and T. J. Hazen for providing this graph.

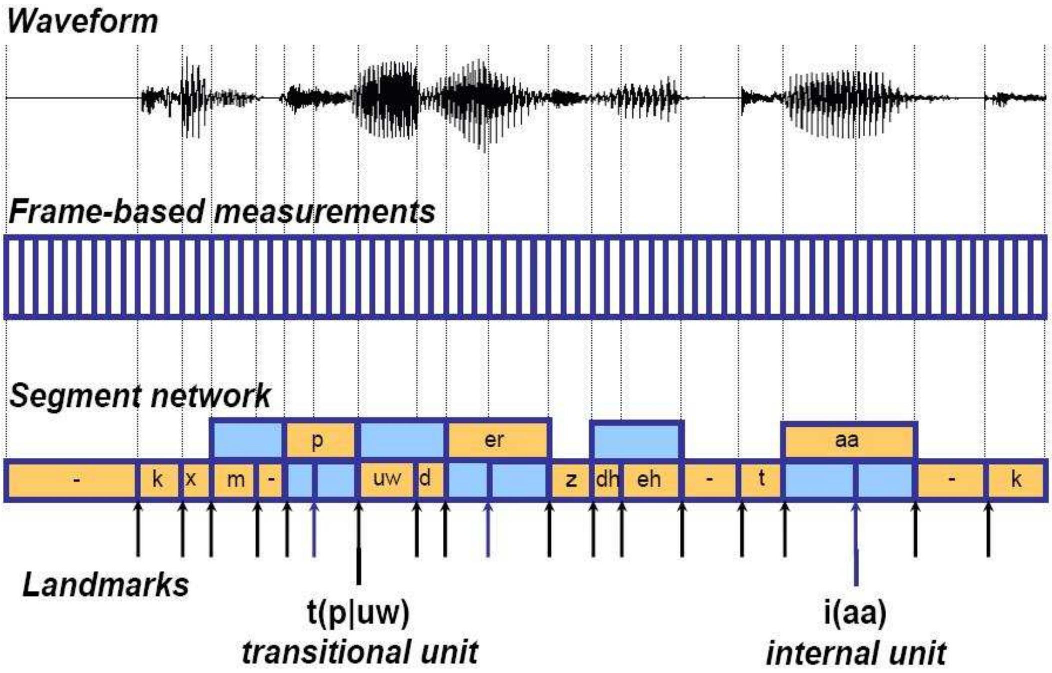


Figure 2-3: In SUMMIT, Mel coefficients are first calculated for short-duration frames at a fixed frame-rate (the middle panel). Next a landmark detector searches for significant acoustic movements. Feature vectors constructed from the Mel coefficients of surrounding frames are modeled by Gaussian Mixtures and a segment network is hypothesized (the bottom panel).

modeling of the manner and place-based landmarks. Phonological rules in SUMMIT are left in intact in this process and compiled into the recognizer as usual. These rules are mainly driven by manner and place spreading and assimilation. As we model manner and place as the basic acoustic units, these phonological rules, which govern the context-dependent realizations of phonemes, are important to our modeling.

In a phone-based SUMMIT recognizer, transitional landmarks and internal landmarks, based on phones, are the basic units for acoustic models. In our experiments, we simply group those phone-based landmarks based on their feature identities. For example, we can place all internal landmarks of schwas into one group and all transition landmarks between alveolar and labial into another group.

We in turn collect all the acoustic observations from a training database and divide them into groups corresponding to the feature-based landmark groups we have. We train acoustic models, using the standard SUMMIT training procedure, for each

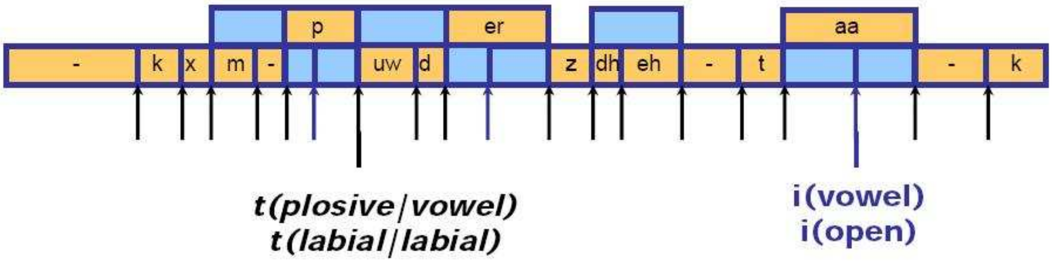


Figure 2-4: Feature-based landmarks are easily derived by mapping phone-based landmarks.

feature-based landmark group. In this way we construct feature-based acoustic models for our experiments.

2.2 Information Fusion

As we are modeling the speech signal along the two organizational dimensions of both manner and place, we now face a challenge not anticipated in the original Huttenlocher-Zue proposal, namely how to combine information from the two feature tiers of manner and place. As manner and place features each describe one aspect of the speech signal, the synergy between these two features provides more constraints for the lexical access procedure. We use information fusion techniques for this task. Information fusion techniques have been widely used for multi-band or multi-stream speech recognition to improve noise-robustness [7, 85] and for audio-visual speech processing [93].

In this thesis, we study three different information fusion schemes. We consider two parallel classification processes based on manner information and place information, respectively. Each process can run independently, performing the speech recognition task by itself, albeit with limited information. We investigate three alternative information fusion schemes: early, late and intermediate. The three schemes differ on when the information from these two channels is combined in the decision-making process, as discussed in the following sections.

2.2.1 Early Fusion

In the early fusion scheme, information from the manner and place dimensions are combined at each landmark. In terms of decision-making, the identity of each landmark is determined by considering both its manner property and its place property. Figure 2-4 illustrates the early fusion scheme in acoustic modeling. Two set of acoustic models, based on manner and place features respectively, are constructed. For each landmark, two acoustic model scores are computed using manner and place-based acoustic models. The two scores are averaged as the final acoustic model score for the landmark.

As discussed earlier, in SUMMIT all knowledge sources are pre-compiled into a single finite state transducer (FST) [86]. The FST is the search space of the recognizer during decoding time. The decoding process finds the most viable path in the FST. In the early fusion scheme, the recognizer searches a single FST and returns a single decoding path. The early fusion scheme enforces all constraints at the earliest possible time. This fusion scheme is very effective in reducing the search space. However, asynchrony between the manner and place channels is not allowed in the early fusion scheme.

2.2.2 Late Fusion

Late fusion, also known as hypothesis fusion, is a widely-used technique to reduce word error rate using multiple recognizers [27]. In our experiments, we build two recognizers along the manner and place dimensions. Each recognizer runs standalone and generates its own hypothesis in the form of an N-best list. The N-best lists generated individually are then integrated using a simple voting scheme.

Figure 2-5 illustrates the late fusion scheme. Two recognizers, based on manner and place respectively, run in parallel. Each recognizer has its own FST and acoustic models. During decoding, each recognizer searches within its own FST and hypothesizes an N-best list.

The late information fusion scheme allows greatest asynchrony between the man-

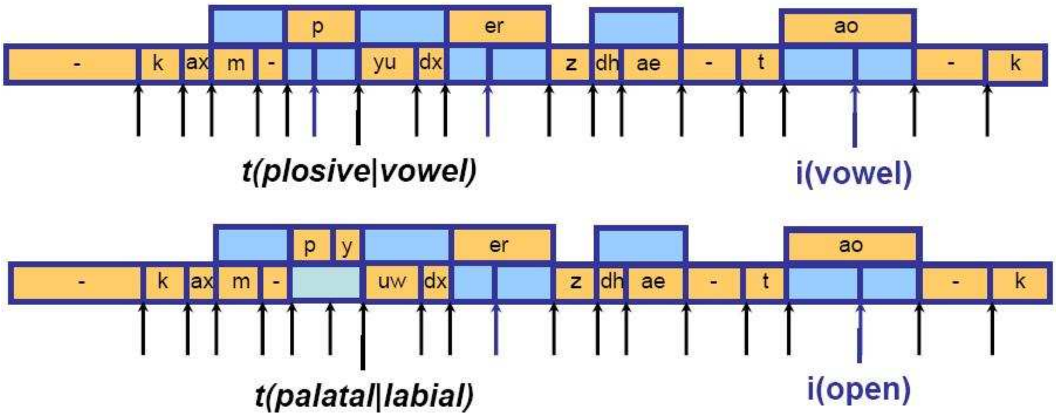


Figure 2-5: In the late fusion scheme, two parallel recognizers search in the manner and place FSTs respectively. Constraints from the two feature channels are combined at the word level.

ner and place channels. The hypotheses from the manner- and place-based recognizer can differ. Even when the two recognizers propose the same word candidate, the underlying segmentation and feature realization can also differ. Thus the two recognizers are able to find the most robust acoustic evidence in decoding, and overall the system has greater flexibility in describing speech variability. On the other hand, as each recognizer itself models only one aspect of the speech signal (manner or place), the discriminating power of each recognizer is relatively limited.

2.2.3 Intermediate Fusion

An intermediate fusion attempts to combine manner and place information in a slightly different way. The intermediate fusion scheme is proposed based on the following observations. First, manner-based broad classes are very effective in describing the steady-state portion of the speech signal. The place-based broad classes are associated with the maximum constriction points of the vocal tract during speech production. These constrictions often result in characteristic patterns of energy distribution at different frequency ranges. As such, the place of articulation is often associated with dynamic movements of the spectral peaks, when energy redistributes as a result of the constrictions.

We are encouraged to propose an intermediate fusion scheme in which we model segment-internal landmarks only by their manner properties and the segment-transition landmarks only by their place properties. The acoustic measurement of the internal landmarks corresponds better with the steady-state behavior of the speech signal and is thus a good candidate for manner identification. The acoustic measurement of the transitional landmarks better captures the spectral dynamics between different segments and is a better candidate for place identification. The way the FST is pre-compiled requires that the internal and transitional landmarks match for each decoding path, thus effectively enforcing both the place- and manner-based constraints to be incorporated into decoding. Figure 2-6 illustrates the intermediate fusion scheme.

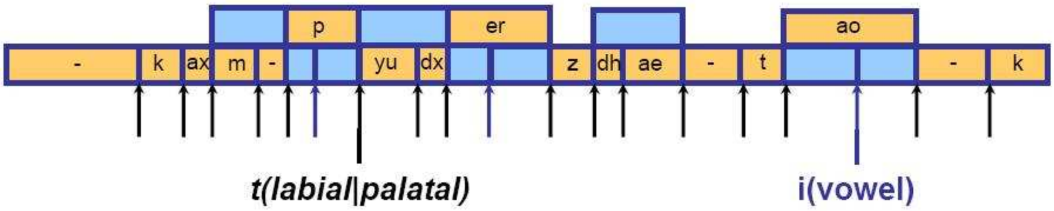


Figure 2-6: In the intermediate fusion scheme, we model internal landmarks with manner-based models and transitional landmarks with place-based landmarks. The decoder searches in only one FST. The topology of the FST ensures that constraints from both the manner features and the place features are utilized during search.

The intermediate fusion scheme does not use the two full sets of manner- and place-based acoustic models. For each landmark, only one acoustic score is computed. It also incorporates constraints from both the manner and place channels into the same decoding paths. Still, it allows flexibility in assigning the manner and place features to segments. In comparison, both the early fusion and late fusion schemes make full use of the two sets of acoustic models, and, for each landmark, both manner- and place-based acoustic scores are computed.

2.3 Experimental Results

In this section we first discuss a comparison between the manner- and/or place-based broad classes and broad phonetic clusters discovered through data-driven approaches. Empirical evidence suggests that broad classes inspired from linguistic knowledge are preferred over clusters derived merely through data-driven approaches. We then discuss a multi-stage speech recognizer in the spirit of the Huttenlocher-Zue proposal. When tested on an isolated-word task, the multi-stage recognizer achieves improved performance over a state-of-the-art phone-based recognizer.

2.3.1 Broad Class Analysis

In Section 2.1, we discussed the design of manner and place-based broad phonetic classes. These classes are primarily motivated by linguistic knowledge, although considerable effort has been given to ensure consistency with real speech data. We believe that a good design of broad classes should meet two criteria. First, within each class, the acoustic instances should be highly homogeneous, while acoustic instances between different classes should be sufficiently heterogeneous. This criterion ensures that the classes are well supported by data. The second criterion requires that the broad classes have strong discriminative power to distinguish lexical items, so that the lexical space can be sufficiently reduced once the broad classes are identified.

In reality, we can evaluate the "goodness" of a broad class design with two measurements: the size of the final acoustic models and the recognition accuracy on a test set. In SUMMIT, the acoustic models are based on Gaussian Mixtures. Intra-class homogeneity translates directly into a small size of the acoustic models. On the other hand, the word recognition accuracy is a good proxy of discriminative power.

In this experiment, we compare the manner and place-based broad classes with phonetic clusters derived completely using data-driven approach [38], as listed in Table 2.1. As we can not directly control the size of the resulting acoustic models, we calibrate the number of clusters so that the final size of the acoustic models, based on these clusters, is only slightly larger than the acoustic models from manner- or place-based broad classes. The recognition experiments are carried out on the Jupiter [117] and Mercury [97] information domain, which will be introduced and studied in greater detail in later chapters. For now, it suffices to know that these are two medium-sized, telephone-quality continuous speech tasks.

Results are listed in Table 2.5, where we compare (1) a manner-based system, (2) a place-based system, and (4) a system integrating both manner and place information as in Figure 2-6. We compare these three systems with (3), a system in which phones are clustered through a data-driven, bottom-up process. We control the number of clusters in (3) such that the overall acoustic models size is roughly the same as those of systems (1), (2) and (4). Our results indicate that the cluster-based system performs better than either the manner- or place-based system operating alone. In part, this may be simply because there were more clusters than in the other models, and thus the cluster-based models can fit the data better. However, with the knowledge-based feature classes and a simple but efficient information fusion scheme (cf. System 4), significantly better performance is achieved than using the data-driven approaches.

System	**Model Size**	**Jupiter**	**Mercury**
(1) Manner	1.78M	30.8	33.1
(2) Place	1.52M	29.5	30.5
(3) Cluster	1.83M	27.9	29.1
(4) Manner + Place	*1.56M*	**25.4**	**25.3**

Table 2.5: Word error rates (WER) of broad class-based models, in the Jupiter and Mercury domains. Manner- and place-based models perform least satisfactorily because of the incomplete information they provide. However, by integration of both manner and place information, the best performance is achieved.

We conclude that the linguistically motivated feature sets we use perform better than the data-driven clusters, which supposedly best fit the data, with a significant margin in our experiments. The output of a clustering algorithm is sensitive to the acoustic measurement used, while, with knowledge-driven methods, we can explicitly define multiple feature dimensions (in our case, manner and place dimensions) and reuse the training data along these parallel dimensions. The robustness of the overall system can be improved by maximizing the orthogonality of the different feature dimensions, for various front ends and acoustic measurements [45]. For these reasons, we consider that knowledge-driven feature sets are a reasonable starting point in our research.

2.3.2 The Phonebook Recognition Task

The Phonebook was collected around 1995 in an effort to build a phonetically rich telephone-quality isolated word database. The database contains about 8000 unique words, which are carefully selected to provide a complete and balanced coverage of triphones, syllables, lexical stress and non-adjacent phoneme co-articulation effects. The complete database contains about 92,000 utterances collected from 1300 native speakers of American English [87]. These utterances are divided into a training set of about 80,000 utterances, an independent test set, and an independent development set of about 6,000 utterances each.

Traditionally researchers have mostly focused on a small vocabulary task defined on the Phonebook database. In this task, the training set contains about 20,000 utterances. An independent test set contains 6,598 utterances from a vocabulary of 600 unique words. In the small vocabulary task, recognizers decode only within this 600-word test lexicon [91, 71, 20, 54]. The best results of past efforts (to our knowledge) are listed in Table 2.6, from System 1 to System 3.

	WER
1. Hybrid HMM/ANN [20]	5.3
2. DBN [54]	5.6
3. HMM+HAMM [91]	4.2
4. Phone-based Landmark Models	3.6
5. ***Transitional Manner Landmarks***	4.5
6. ***Transitional Place Landmarks***	4.5
7. ***5 and 6 N-best Fusion***	3.0

Table 2.6: Phonebook Small Vocabulary Results for various systems. System 4 is our baseline system.

In System 4, we re-implemented a phone-based recognizer with context-dependent duration modeling [71], a system which was originally developed for the more challenging, large-vocabulary recognition task on Phonebook, and which will be used extensively in the experiments described below. On this small vocabulary task, this system achieves a WER of 3.6%. We replaced the phone-based transitional landmark models with manner-based models (System 5) and place-based models (System 6). Results show that, on this small vocabulary, the loss of discriminant power is not serious when we replace the phone-based landmarks with broad classes. Best results are achieved when we combine the outputs of these two recognizers, using the weighted sum of the N-best list scores, adopting the late fusion technique described earlier [27]. In case 7 of Table 2.6, the recognition error is 30% better than the nearest competitor reported in the literature (4.2% vs 3.0%)[2].

In the remainder of this section, we focus on the large vocabulary task of Phonebook, first proposed by Livescu et al. [71]. In this experiment, the training process makes full use of the 80,000 training utterances in Phonebook. The decoder searches on the full 8,000-word vocabulary (compared to the 600-word test vocabulary). At the time our experiments were carried out, the best result to our knowledge in the literature was achieved by the phone-based landmarks + context-dependent duration modeling reported in [71]. We rebuilt the system described in [71] and used this recognizer as the *verifier* in our two-stage recognition system[3] as well as the baseline system in the Phonebook large vocabulary experiments discussed below.

2.3.3 Two-Stage Recognizer Configuration

We built a two-stage speech recognition system using the SUMMIT framework. The first stage corresponds to the *lexical access* module in the Huttenlocher-Zue model. We construct the first stage by building a SUMMIT recognizer using feature-based acoustic models. For example, a manner-based first-stage will use acoustic models trained upon broad manner classes. In 2.1.3 we discussed how these feature-based acoustic models are trained.

The FST of the first-stage recognizer is compiled using the standard SUMMIT

[2] Although only 17% better than the result we achieve using our baseline system (3.6% vs 3.0%).

[3] The author thanks Karen Livescu for her help with these experiments.

procedure. In fact, the first-stage recognizer FST is equivalent to the FST used in the baseline recognizer. During recognition, the feature-based acoustic models of the first-stage recognizer calculate for each acoustic measurement the probability of being a particular feature-based landmark. By checking the phone and feature mapping, this probability is translated into a probability of one or more phone-based landmarks. Based on this information, SUMMIT searches the FST and hypothesizes words. In this way the first-stage recognizer hypothesizes words using a set of feature-based acoustic models. In our experiments, we keep a 50-best list (the cohort) of the first-stage recognizer output for further analysis in the second stage.

The second stage corresponds to the *verifier* module in the Huttenlocher-Zue model. It takes the form of a phone-based SUMMIT recognizer, using a set of detailed, phone-based acoustic models. In this thesis, the second stage recognizer is also used as our baseline recognizer, but with a major difference: whereas the baseline recognizer decodes on the *default lexicon*, the second-stage recognizer decodes only on the *cohort*.

In Table 2.7, we outline the major configuration differences among the first-stage recognizer, the second-stage recognizer, and the baseline system, in terms of the acoustic models and the search space. The second-stage recognizer and the baseline recognizer share the same phone-based, detailed acoustic models, whereas the first-stage recognizer uses a feature-based recognizer. The first-stage recognizer and the baseline system both decode on the default lexicon. The second-stage recognizer decodes on a reduced cohort space. In SUMMIT, all knowledge sources – language modeling, lexicon, phonological rules – are pre-compiled into a finite state transducer. Knowledge sources other than the lexicon, i.e., language modeling and phonological rules, are identical across the three different configurations. However, language modeling and phonological rules in the FST of the second-stage recognizer may be different due to reductions in the lexicon. For example, phonological rules are defined as context-dependent rewrite rules in SUMMIT. With a reduced lexicon (cohort), the number of phonetic contexts one can observe from the cohort is much smaller. Rules that are based upon phonetic contexts beyond the cohort will be pruned in optimization of the second-stage finite state transducer. This is a desirable outcome as the second-stage recognizer will then focus on aspects that distinguish candidates within a cohort.

	Two-Stage Recognizers		Baseline Recognizer
	First Stage	Second Stage	
Acoustic Models	Feature-based	Phone-based	Phone-based
Search Space	Default	Cohort	Default

Table 2.7: Configurations of the first- and second-stage recognizers in our two-stage framework and configuration of the baseline recognizer. The second-stage recognizer and the baseline recognizer share the same phone-based, detailed acoustic models whereas the first-stage recognizer uses a feature-based recognizer. The first-stage recognizer and the baseline system both decode on the default lexicon. The second-stage recognizer decodes on a reduced cohort space. Unless otherwise noted, other knowledge sources, e.g., language models, phonological rules, etc., remain the same for all three recognizers.

The feature-based acoustic models are more robust and compact, albeit less discriminative when the vocabulary is large. In this situation, it is ideal to use these feature-based models as a "filter" to limit the search space to a high-quality cohort so that context-dependent language modeling and/or acoustic-phonetic analysis techniques can be effectively applied [51].

Our implementation of the first-stage recognizer differs from the original Huttenlocher-Zue model in that we built into the first stage not only constraints from broad phonetic classes or manner and place features. Rather, we also introduce other knowledge sources into the first-stage recognizer. In fact, the first-stage recognizer is equipped with the same set of phonological rules and language modeling constraints as the second-stage recognizer. Our implementation is consistent with interactive lexical access models in that higher level knowledge, e.g. semantics, syntactics, contextual constraints, etc., are allowed to interact with acoustic-phonetic processors during the lexical access process [81, 78, 23, 77]. In later experiments on continuous speech, the ability to incorporate higher level knowledge even at the first-stage recognizer becomes very important.

The second difference of our first-stage recognizer is that, while we carefully design and calibrate the manner and place classes based on linguistic knowledge, we rely on the data-driven device provided by SUMMIT to *model* the actual instantiations of these broad classes observed in a large corpus. This approach is general enough that it can easily be enhanced with more sophisticated signal processing algorithms [45] or data-analysis algorithms [83], or be replaced with other acoustic modeling systems [8]. Figure 2.3.3 shows the configuration of our two-stage recognizer.

Our approach allows for fast prototyping and testing. Many speech recognition systems support as a format of output, N-best lists. The N-best list representation is very concise and can be very efficiently generated [99, 48]. As the *de facto* standard output of a speech recognizer, they are a convenient representation as the protocol between multiple stages. For the isolated-word task at hand, the N-best list usually contains a list of words, ranked by the associated likelihood scores, which can be used directly as cohorts. Because of the rather standard output format of N-best lists, we have great flexibility in our choice of the first-stage recognizer so long as the N-best output satisfies our needs.

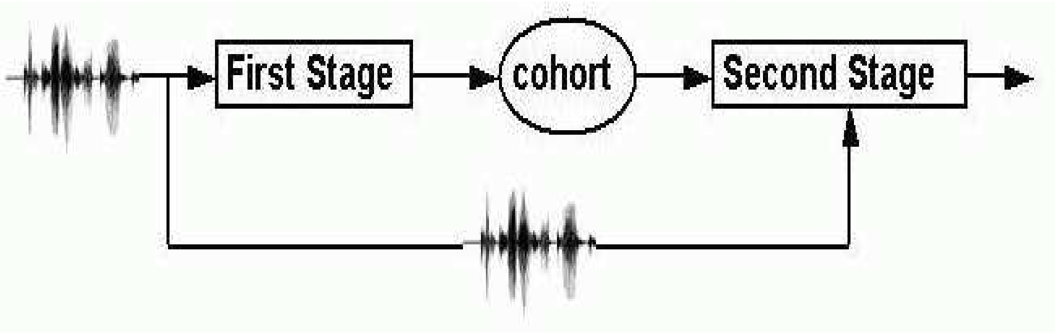


Figure 2-7: A two-stage speech recognition system. Feature-based models in the first stage generate a high-quality cohort for detailed analysis in the second stage.

2.3.4 Phonebook Large Vocabulary Results

Cohort Performance

The goal of the first-stage recognizer is to build feature-based models and apply them to prune the search space at an initial recognition stage. With a reduced search space, we can afford computation-intensive algorithms, such as context-dependent language understanding and/or acoustic-phonetic analysis, in later recognition/understanding stages. We are interested in the cohort quality, i.e., the "missing rates" – the percentage of words that fall outside the cohort – for a given cohort size, of different feature-based models and different fusion schemes. In Figure 2-8, we plot the cohort missing rates of five systems against the size of the cohort. In two of the systems, the first-stage recognizer uses either manner features or place features alone. In three other systems, the first-stage recognizer employs one of the three information fusion techniques discussed earlier when generating the cohort. As shown in the figure, the cohort missing rate drops dramatically when proper information fusion techniques are applied. In particular, the early fusion scheme performs best when the cohort size is small (<50). The late fusion scheme performs extremely well when the cohort size is medium or large.

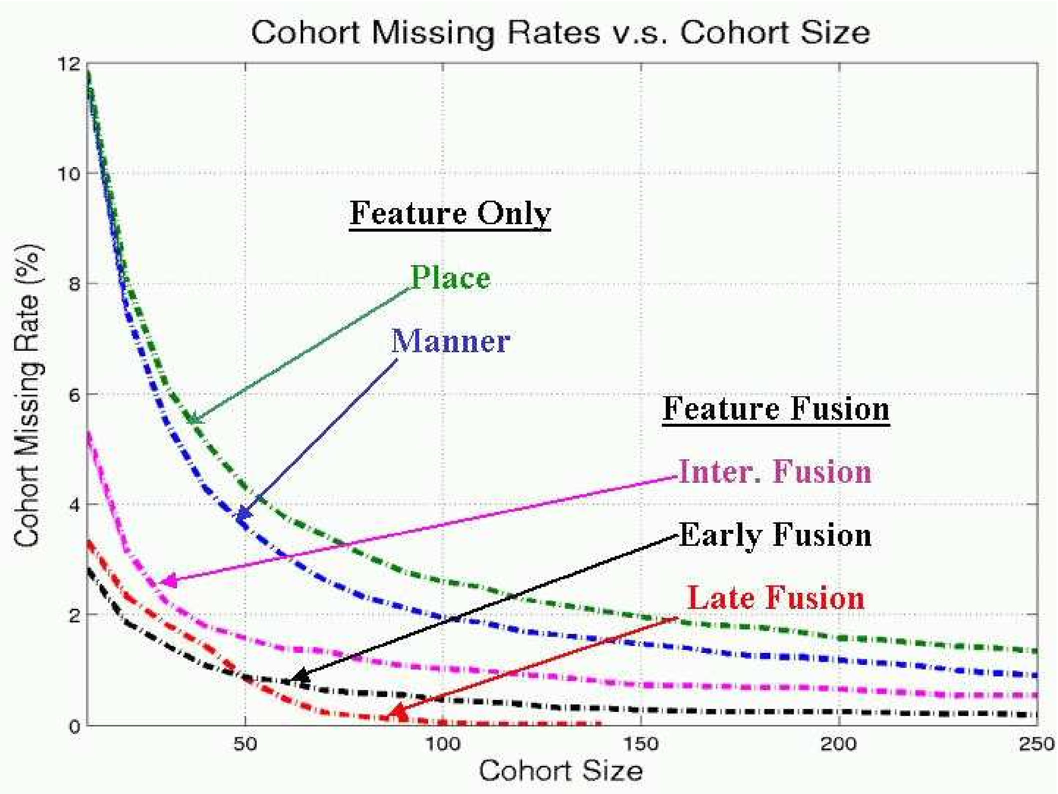


Figure 2-8: The cohort missing rates as a function of cohort size, for various conditions.

It is important that we keep in mind the associated computational costs when we compare the cohort performance of various first-stage recognizers. Table 2.8 tabulates the size of the acoustic models in different configurations of the first stage. The early and late fusion systems, because they use both manner- and place-based models, have more parameters in the acoustic models. While they would not cause data-sparseness problems during training, since the training data can be re-used when training the manner- and place-based acoustic models, they would result in more computational load at the decoding time. Comparably, the intermediate fusion scheme is more attractive because of its performance and its computational profile.

	Size of Acoustic Models
Place Model	496k
Manner Model	676k
Intermediate Fusion	521k
Early Fusion	1.17M
Late Fusion	1.17M

Table 2.8: The size of the acoustic models for different configurations of the first-stage recognizer.

Recognition Accuracy

A fixed cohort size of 50 was chosen in the final experiments. This corresponds to a reduction by a factor of 160 in terms of vocabulary size. The cohort is rescored using the baseline recognizer described in [71]. The WER of the second stage is reduced to 8.4% compared to the 8.7% of the baseline recognizer.

	WER
Context-dependent Duration Modeling [71]	8.7%
Two Stage System	8.4%

Table 2.9: Phonebook Large Vocabulary Results.

When we rescore the cohort, we can also keep the recognizer score from the first stage. This way we preserve in the second-stage more information from the first-stage than the mere reduction of lexical size. Further improvements are observed for all three information fusion schemes employed in the first stage, as shown in Table 2.10. The improvement in performance in Table 2.9 is not statistically significant. But the performance improvements in Table 2.10 are statistically significant at p = 0.05 level. This result suggests that models based on linguistic features provide helpful information.

We also explored a three-stage system: a 300-word cohort generated by the manner-based models is re-scored by the place-based models in the second stage, and a reduced 50-word cohort from the second stage is scored by the phone-based models in the final stage. Although the final result of this three-stage system is similar to that of the other two-stage systems, it is computationally more efficient, mainly because the first stage, by virtue of modeling *only* the manner class dimension, has a significantly reduced search space.

	WER
Early Integration	7.9%
Intermediate Integration	8.0%
Late Integration	8.0%
Three Stage System	7.9

Table 2.10: Recognition results for the 8,000 vocabulary experiments under different integration schemes. These results are significantly better than than the 8.7% WER achieved by the baseline system.

2.4 Discussions

In this chapter, we first discussed the modeling of linguistic constraints in the Huttenlocher-Zue framework. Our methods differ significantly from the original Huttenlocher-Zue proposal in the following aspects:

First, we found both theoretical and empirical evidence suggesting that, in addition to the manner features in the Huttenlocher-Zue formulation, place features are also a significant source of linguistic constraints. We embrace the place features in our methods by organizing speech units along the parallel dimensions of manner *and* place of articulation.

Secondly, unlike the original Huttenlocher-Zue model, which used a discriminative, heuristic-based algorithm for broad classification, we delegate this task to the acoustic modeling device in the SUMMIT framework, which uses a data-driven, Gaussian mixture methodology. We believe this data-driven approach is preferable in dealing with large quantities of data. Our treatment of place of articulation, i.e., using only the traditional place system of consonants for all speech units, greatly simplifies the acoustic modeling procedure. We invested a lot of effort in the design and calibration of the broad phonetic classes along the manner and place dimensions to best match the data. We also choose to model feature-based landmarks as the basic units for acoustic modeling, as landmarks better preserve the manner and place properties.

Thirdly, because we model both manner and place features in our system, we also explored different information fusion techniques to combine constraints from the two feature channels. The three information fusion schemes differ in terms of when constraints from the two feature channels are combined. As a result, the three schemes have different flexibility in describing speech phenomena as well as different computational profiles.

We have tested the effectiveness of the manner and place classes we designed through speech recognition experiments. We compared our broad classes with a broad phonetic class derived through data-driven, bottom-up clustering. We also used transitional models based on the manner or place features in the Phonebook small vocabulary task. In both experiments, the broad manner and place classes perform satisfactorily. When the two feature channels are combined through information fusion techniques, a significant performance improvement is observed over either one modeled independently.

In this chapter we demonstrated our two-stage speech recognition framework. Compared with the Huttenlocher-Zue proposal, our implementation differs in that we use the SUMMIT landmark-based recognizer for both lexical access and verification. Thus, we implement in the first stage a more interactive lexical access module than originally proposed in Huttenlocher-Zue. The N-best lists output from the first-stage recognizer are a natural representation of cohorts.

We tested this two-stage recognition framework on the Phonebook database. Our experiments support our view that manner and place features provide useful constraints for speech recognition. What is more encouraging, however, is that information fusion methods that can combine the two feature channels provide much more constraint for lexical access. In particular, the intermediate fusion scheme requires much less computation without sacrificing much of the discriminant power.

Our two-stage speech recognition system, on the 8000-word Phonebook recognition task, achieved a recognition WER that is 10% better than that achieved in the best results reported in the literature.

Finally, we close this chapter by discussing briefly the distribution of computational load between the two stages. Table 2.11 shows 1) the number of parameters of the acoustic models; and 2) the size of the lexicon of the two stages and compares these numbers with those of the baseline. The computational load is distributed between the two stages. The size of the acoustic models can be used as a proxy for the computational cost associated with acoustic modeling. As we see from Table 2.11, the size of the acoustic models in the first stage is roughly one third that of the second stage. The feature-based acoustic models have a smaller footprint in terms of storage requirements and computation associated with it. It can be installed in devices with limited memory and/or computation power, e.g. hand-held devices. The more detailed acoustic models in the second stage are larger in size, which implies higher computation, however, the search space in the second stage is already effectively pruned. The two-stage framework provides flexibility to balance the computational load between the two stages. The two-stage framework also provides a parsimonious solution to incorporate speech recognition capabilities into hand-held devices, where memory and computation power on hand-heled devices are limited and a distribution of computation between the devices and a back-end server is preferred.

	Size of the Acoustic Models	Size of the Lexicon
First Stage	521K	7,982
Second Stage	1.6M	$\leq$50
Baseline	1.6M	7,982

Table 2.11: The distribution of computational load between the two stages. The size of the acoustic models can be used as a proxy for the computational cost associated with acoustic modeling. The first stage recognizer has a relatively small computational cost when it searches in the full lexicon. In the second stage, when the computational cost is high, the search space is already effectively pruned. The two-stage framework provides flexibility to balance the computational load between the two stages.

Chapter 3

Two-Stage Recognition of Continuous Speech

In Chapter 2, we have demonstrated a two-stage speech recognition system based on the Huttenlocher-Zue model. We have examined the effectiveness of the linguistic features as constraints for lexical access. Our two-stage recognition system achieves more than 10% improvement in the Phonebook recognition experiments over the baseline, a state-of-the-art phone-based recognizer.

In this chapter, we consider extending the two-stage recognition framework to handle continuous speech. Although it suggests that the same principles can be applied to the recognition of continuous speech, the original Huttenlocher-Zue model did not provide any workable plans [52]. In one of the earliest attempts to apply this model to continuous digit strings [10], difficulties already arose as to how to hypothesize digit strings from broad class segmentation graphs. In efforts to develop a two-stage recognizer for Italian [28, 29], the problem was solved by running a dynamic programming procedure on the broad class segmentation. This early effort also met with only limited success, although its tight coupling of the two-stages is rather efficient.

There can be multiple reasons why these past efforts have not been successful. First, they all acknowledge the potential imperfection of the segmentation but fail to address the problem rigorously. In [28, 29], the researchers basically assume one segmentation path of the speech stream, whereas in [10], a small segmentation graph is assumed. Both acknowledge that the segmentation can be imperfect. In [28, 29], the authors employ a dynamic programming procedure for added robustness. In [10], the researchers enumerate possible word candidates at *any* position in the speech stream, a scheme that essentially bears much resemblance to dynamic programming. However, both methods operate only *after* the segmentation is done. Although allowing segment substitution, insertion and deletion does increase robustness, the dynamic programming procedure is effective only when the segmentation paths are of relatively high quality. In our opinion, the classification of short frames in [28, 29], which eventually led to a segmentation, can be very sensitive to noise.

Next, in [28, 29] the lexical access procedure was based on broad classification alone – no higher level information was utilized. We suspect that the lack of higher level information significantly limits the performance of the lexical access module, as both theoretical and empirical evidence suggests that high level constraints are important to the recognition of continuous speech [41, 80].

Thirdly, there lacks a mechanism to compensate for the errors made in the lexical access step. As a result, the authors pointed out in [28, 29] that errors are propagated to the second stage.

In our experiments, the first stage recognizer is constructed as a recognizer with feature-based acoustic models. This implementation addresses some of the above problems. First, the broad classification and segmentation process is now performed by SUMMIT, which during decoding constructs very sophisticated segmentation graphs. This addresses the issue of segmentation graph quality, although we admit that perfect segmentation is still very difficult to achieve. The probabilistic search method in SUMMIT allows segment substitution, insertion and deletion and provides greater flexibility than a simple dynamic programming scheme. Also, very importantly, the first stage recognizer builds in higher level constraints, including language modeling, phonological rules, etc.

3.1 Cohort Representation

In the previous chapter, we suggested that the N-best list is a logical choice to represent the cohort. The N-best list representation is very concise and can be very efficiently generated [99, 48]. As the *de facto* standard output of a speech recognizer, it is a convenient representation as the protocol between stages.

The challenge lies in the fact that the cohort space needs to be as constrained as possible so that recognition in later stages is efficient, while at the same time it needs to be general enough so that the correct answer is indeed included. With the isolated word task where the hypothesis space is limited (by the lexicon), the N-best representation is adequate. With the infinite hypothesis space of a continuous speech task, the N-best space is too restricted and the correct answer is often inappropriately pruned. Empirically, if we only re-score the N-best paths from the first stage, we observe a significant performance drop, similar to the results reported in [28, 29].

Representation efficiency is another consideration. For the isolated-word case, the candidates in the N-best list are unique words. Redundancy at the word level is minimal. (One could argue, though, that other units such as syllables or metrical feet are more efficient representations.) The N-best list for continuous speech, on the other hand, is quite redundant at the word level. Table 3.1 and Table 3.2 contain sample N-best lists from the Phonebook isolated-word task and the Jupiter continuous speech task. The sample N-best list of continuous speech shows great similarity among the various N-best list entries.

Another limitation, if we are to keep the N-best list as the cohort for continuous speech, is the loss of generality in language modeling. SUMMIT, similar to many state-of-the-art speech recognition systems, uses n-gram statistics for language modeling. If we are to use only the N-best entries, the second stage recognizer will lose the ability to generate novel word sequences.

To generalize to continuous speech, we decided to consider a "cohort" to be the set of *words* induced from the N-best output of a feature-based first stage. This approach is not difficult to understand conceptually. Consider Morton's logogen model [81],

109.8592	kennebunk
97.8020	cannonball
82.8171	diamondback
78.4902	cameraman
78.2203	annenberg
74.3227	cadillac
73.7883	carpetbag
72.7965	chatterbox
72.2273	catalog
71.0903	antibody

Table 3.1: A sample N-best list from the Phonebook domain. The numbers are a combination of acoustic model scores and language model scores. The candidates in the N-best list are unique words.

121.1377	uh what is a rainy city in america
120.9076	uh what is the rainy city in america
119.7078	uh what is the rain in cities in america
119.5466	uh what is a rainy city in in america
119.3165	uh what is the rainy city in in america
117.3398	uh what is the rain in city in america
117.0397	uh what is the rain in unknown city in america
116.8669	uh what is the rainy in cities in america
116.7751	uh what is the a rainy city in america
116.5495	what is a rainy city in america

Table 3.2: A sample N-best list from the Jupiter domain. The numbers are a combination of acoustic model scores and language model scores. The N-best list entries are very similar to one another.

word candidates are activated by various input features, which are generated as the speech stream progresses. If we keep track of all the logogens that have a certain degree of activation, we may end up with a bag of words which are the union of all the cohorts to individual words. Admittedly, to disentangle this bag of words and determine the cohort for individual words is very complicated. Yet, for all practical purposes, our approach returns a cohort for the entire utterance. In effect, we restrict the lexicon for the second stage to be only the words that have appeared in the N-best list of the first stage. On the isolated word task, this generalization gives us the same cohort as before. With continuous speech, this generalization allows the second stage to hypothesize novel word sequences unseen in the N-best list, thus providing the capability to recover from mistakes committed in the first stage. Note that the language model for the first and second stages remains the same.

To gain some insight into the effectiveness of this generalization, we computed the word level cohort coverage rate as a function of N, the depth of the N-best list. A word is considered a "hit" if it appears in the N-best list. A word in the transcription but missing from the N-best list will inevitably lead to an error. Figure 3-1 shows the word level cohort coverage rate as the N-best depth increases in the Jupiter [117] domain, which will be described in detail soon. The lower curve shows that roughly 92% of the reference words are covered with a 50-best list, which is encouraging, especially since the OOV rate of this data set accounts for about 2.3% of the words. We also plot, as the upper curve in Figure 3-1, the coverage rate of words that can be correctly recognized by a state-of-the-art recognizer. With a 50-best list, about 98% of such words are covered. We also notice that both curves level off at a very modest N-best depth.

Since the cohort space is now the original search space restricted to the N-best lexicon, the reduction in search space can be approximately estimated by the reduction in lexicon size. The upper curve in Figure 3-2 shows the average N-best lexicon size as N grows. We see that the induced vocabulary is significantly smaller than the original "full" vocabulary. With an N-best depth of 50, the average vocabulary size is only 17.5, less than 1% of the original, 1924-word vocabulary. The lower curve in Figure 3-2 shows the average number of *correct* words in the induced vocabulary, as the N-best depth grows. With N equal to 50, the N-best vocabulary contains 3.93 correct words on average, which is very close to the average sentence length of 4.28 of this data set, as shown by the solid horizontal line of this figure.

These results suggest that using an N-best list from the feature-based first stage could significantly trim down the search space, as evidenced by the reduction in the size of the N-best lexicon. The amount of "useful" information contained in the N-best list, as shown by Figure 3-2 and by the lower curve in Figure 3-1, saturates rapidly as the N-best depth grows, which indicates that only a modest-depth N-best

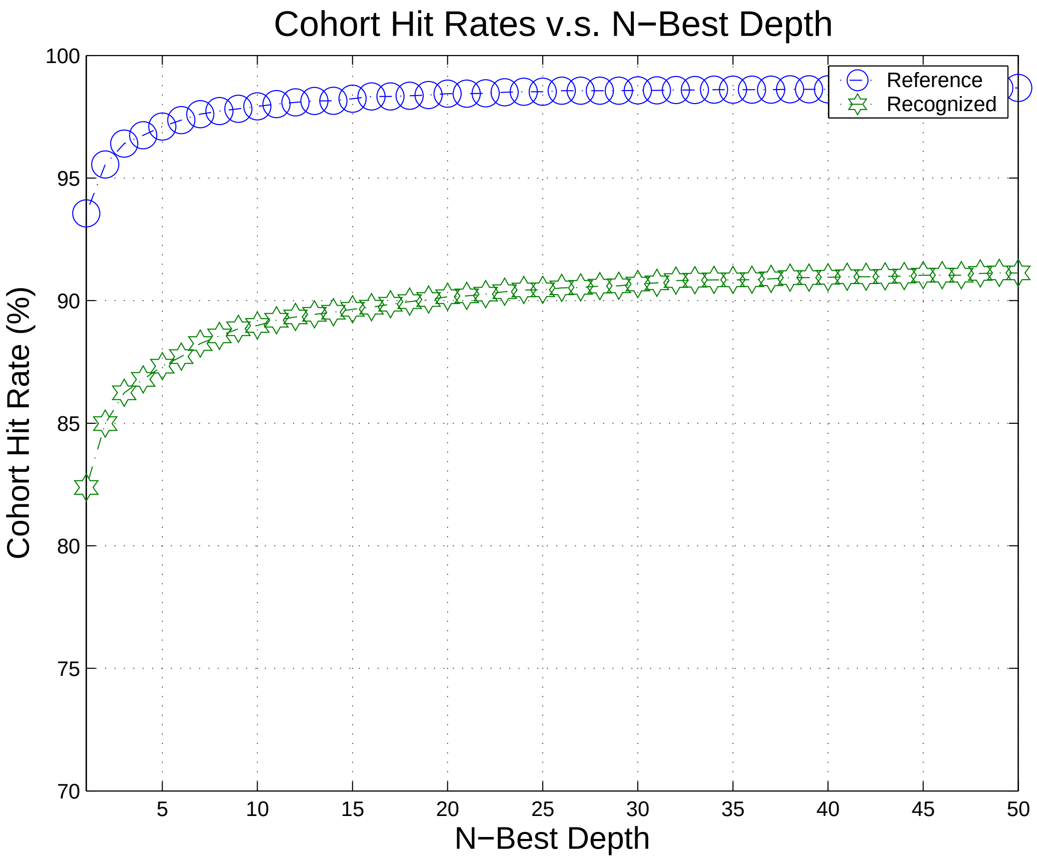


Figure 3-1: The cohort hit rates as a function of N-best depth. Above is the hit rate for words that are correctly recognized by a state-of-the-art phone-based recognizer. Below is the hit rate for the correct transcriptions. In both cases, the hit rates level off at a very modest N-best depth. The difference between the two curves is the words which are missing in the cohort but are not correctly recognized by the baseline recognizer either.

list is necessary for our purposes. Instead of generating deeper N-best lists, some other mechanism is necessary for the system to recover the search space that has been inappropriately pruned.

3.2 Error Recovery Mechanism

We carry out speech recognition experiments using a two-stage speech recognition system. In the first stage, we use the intermediate-fusion scheme to combine manner and place constraints in the first, feature-based stage. In the second stage, we use

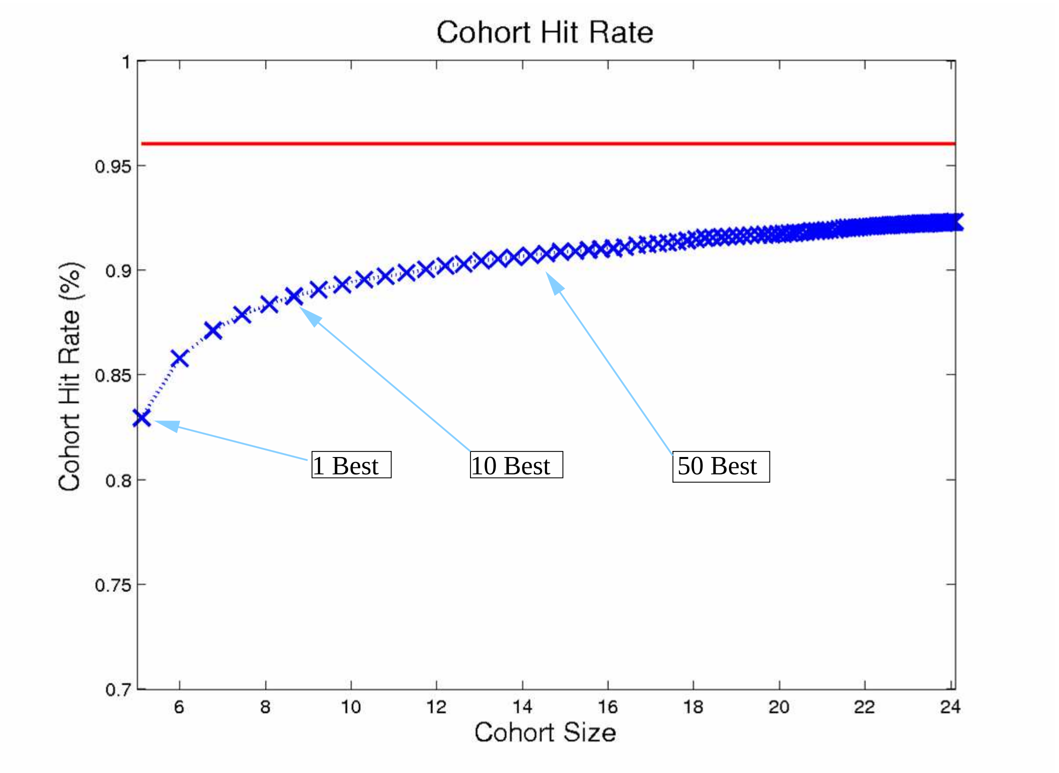


Figure 3-2: The size of the vocabulary induced from the first-stage N-best list. With a 50-best list, the average lexicon size is about 17.5, less than 1% of the original, 1924-word vocabulary. The lower curve shows the average number of correct words contained in the N-best lexicon. With N equals 50, the average number of correct words, 3.93, is very close to the average sentence length of 4.28, as shown by the solid horizontal line. Most notably, the lower curve saturates extremely fast, indicating that a shallow N-best list would contain most of the “correct” information.

a state-of-the-art SUMMIT landmark-based recognizer for more detailed acoustic phonetic analysis. The second stage recognizer uses a language model casted only to the set of words derived from a 50-best list output of the first stage.

Initial experiments showed a degradation in performance, even though we were rescoring a more general, restricted language model rather than exact sentences from the first-stage output. This is not too surprising, as Figure 3-1 shows only 98% of the words that can be correctly recognized by the baseline make it into the cohort.

Since the amount of useful information contained in the cohort level off rather quickly, there is probably little hope of overcoming this difficulty by increasing the depth of the N-best lists. Instead, we analyzed the cohorts and tried to identify the source of errors.

Analysis of the cohort shows that, in function words such as "I," "you," "yes" and "no," the feature-based models are likely to make errors. Such words are often reduced in their acoustic realization because they contain less information [44]. On the other hand, the feature-based models perform very well on content words, which are information salient and hence acoustically prominent.

We considered two ways to help recover the over-pruned search space. The first method is based on word frequency. We note that the words that are often over-pruned tend to have high frequencies [68]. Words of higher frequency are likely to be reduced acoustically. The Morton model, for example, suggests that higher frequency words have a lower threshold value for activation. A lower threshold permits the deletion of, among other things, certain acoustic-phonetic features, which our first-stage recognizer attempts to model, leading to an over-pruning of the first stage outputs. Thus we can complement the N-best list vocabulary with a small set of most frequently used words to provide the syntactic glue the first-stage recognizer is likely to miss.

An alternative method to systematically enhance the cohort lexicon is to run the first-stage recognizer on a development set to create a complementary vocabulary of words that are missing from the N-best output of the feature-based models. In practice, for each utterance, we compare the transcription with the N-best list from the feature-based first stage. Words from the transcription that are missing from the N-best list are counted. Operating on the entire development set, we create a list of words associated with a count of how often they are missing. This way we empirically discover from the real data the set of words that the first stage recognizer is likely to miss using the development set. Compared to the previous method, this empirical method emphasizes the actual discovery of the relative "weakness" of the first-stage.

3.3 Experimental Results

In this section we discuss results of a two-stage speech recognition on the continuous speech recognition tasks in the Jupiter [117] and Mercury [97] information query domains developed at the Spoken Language Systems group at MIT.

Jupiter [117] is a weather information domain deployed live at the MIT Spoken Language Systems group. It is accessible through a toll-free telephone number within North America and a non-toll-free telephone number around the globe. It provides up-to-date weather forecast information for about 500 cities in the world using data from WSI Intellicast [3]. When users dial in, the system engages the user in a mixed-initiative dialog and provides weather forecast information upon request.

Mercury [97] is a conversational interface that provides users with flight and pricing information, using data from SABRE [2]. It enables users to query, plan and schedule multi-legged travel itineraries to about 500 cities, almost half of which are within the US. For registered users, it can also help with the booking and finalizing the itineraries. Mercury is also accessible via a toll-free telephone number.

3.3.1 Experiment Setup

As speech recognition tasks, Jupiter and Mercury are both telephone-quality, multi-speaker and spontaneous continuous speech recognition tasks. For the experiments described in this paper, the Jupiter weather domain has a lexicon of 1924 words and a test set of 1888 utterances. The Mercury domain has a lexicon of 1568 words and a test set of 2049 utterances. Each test set contains a “clean” subset which is free of artifacts or out-of-vocabulary (OOV) words. A fixed training set, which contains over 140k utterances from both domains, is used throughout our experiments. This training data is used to train both the feature-based acoustic models used in the first stage, and the phone-based acoustic models used in the second stage and in the baseline system. We use as our baseline the same state-of-the-art phone-based landmark models in both domains. In our experiments, the forward search uses a bi-gram and the backward search uses a tri-gram language model.

In our experiments, the first stage recognizer still uses feature-based acoustic models. In these experiments, we also adopted the intermediate information fusion scheme in the first stage recognizer. The FST of the first-stage recognizer is equivalent to the FST of the baseline. The first stage recognizer, relying on feature-based acoustic models, generates a 50-best list. We collect all the words from the 50-best list as cohorts for analysis in the second stage.

We thus report our experiments on the Jupiter weather information domain and the Mercury air travel planning domain. The first-stage recognizer uses the feature-based models as illustrated in Figure 2-6 on the basis of their compactness and good performance. It uses a "full" FST created with the entire lexicon. A 50-best list is generated for each utterance. Independently, a vocabulary of the 200 most frequent words in each domain, and a vocabulary of the 100 words the first-stage recognizer is most likely to miss, are created. The second stage lexicon is the N-best vocabulary augmented with one of these complementary sets. On average this translates to roughly a ten-fold reduction in vocabulary size for the second stage recognizer. In the second stage, the "full" FST is pruned to eliminate all arcs outputting words not licensed by the reduced vocabulary. We use as a baseline a state-of-the-art phone-based SUMMIT recognizer in the second stage for both domains. Results are reported on both the clean subset and the full test set.

3.3.2 Recognition Results

The speech recognition results are listed in Table 3.3. When we use the top 200 most frequent words to compensate for the over-pruned search space (System I), the two stage system performs slightly worse than the baseline, for the Jupiter domain. Using the alternative set of words selected from a development set on the basis of their absence from the first stage output (System II), the two-stage system outperforms the baseline, for both Jupiter and Mercury. In hindsight, these results are not surprising, since the scheme for System II specifically focuses on words known to present problems in the first-stage recognizer. Figure 3-3 further illustrates the performance dynamics of System II as the number of compensative words varies. When no compensative words are incorporated, the system performs worse than the baseline, as the words missing from the first stage can not be recovered. The performance improves as we add more compensative words until it saturates with about 100 words. After this point, the performance slowly decreases and converges to that of the baseline as more compensative words are added. We believe the improved performance is due to the improved robustness from representing the lexical entries with broad linguistic features in the first stage.

	Jupiter		Mercury	
	C-Set	F-Set	C-Set	F-Set
Baseline	11.6	18.4	12.7	22.1
Two-stage System I	11.9	18.6	N/A	N/A
Two-stage System II	**11.0**	**17.9**	**12.4**	**21.7**

Table 3.3: WER's on Jupiter and Mercury. Two-stage System I uses the 200 most frequent words to enhance the second stage vocabulary. Two-stage system II uses a complementary set of words that the first stage tends to make mistakes in. The second two-stage system improves the final performance on both the clean data (C-Set) and the full test data (F-Set), for both domains. We did not run Two-stage System I for Mercury, as results from the Jupiter domain suggest relatively inferior results.

Figure 3-4 gives an example where the two-stage approach performs better. The utterance is, "update on tornado warnings". The phone-based models recognized this as "date on tornado warnings", as shown in the lower panel. This is a common type of mistake where confusion between utterance-onset noise and the first word arises at the beginning of an utterance, although the first word is prominently articulated. In this particular example, the fact that the /p̚/ (/`pcl`/ in the figure) in "`update`" is noisy might also contribute to the error. In the two-stage framework, the first stage rules out the erroneous candidate "date", and enables the second stage to produce the correct result, as show in the upper panel. The feature-based models, probably because they use broad classes and are more robust, are less sensitive to noise and perform well in the presence of reliable acoustic evidence. For this reason, we hypothesize that the feature-based models are able to provide complementary information to a phone-based system. A McNemar test on the full test set of Jupiter shows that the reduction

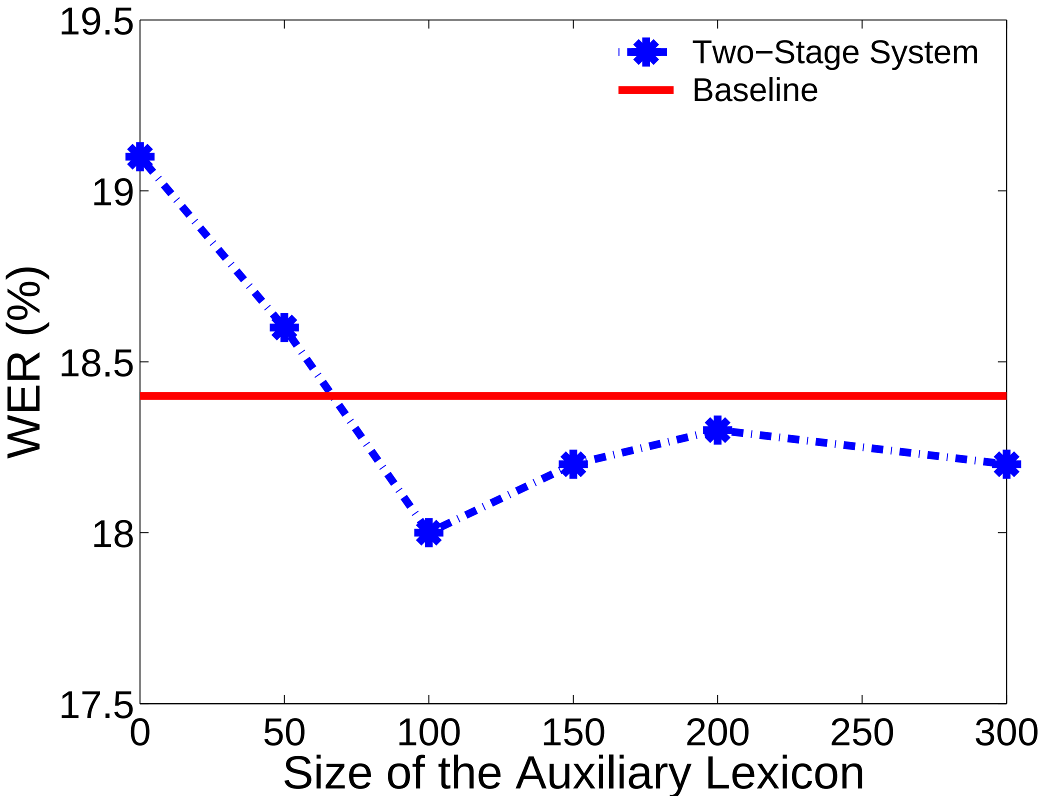


Figure 3-3: Performance dynamics of System II as we increase the number of compensative words.

in error is significant at the $p = 0.05$ level.

3.4 Discussions

This chapter extends our research to build multi-stage recognizers using linguistic features to continuous speech. The N-best list, at the utterance level, is no longer an appropriate representation as a cohort. The reason is that the first stage recognizer, equipped with feature-based acoustic models, has less discriminant power. In the isolated word tasks, we can significantly increase the coverage of the search space by increasing the cohort size. The cohort sub-space provides complementary information and adds robustness. In the continuous speech case, as empirical evidence suggests, re-scoring of the N-best lists only propagates errors from the first stage.

At the word level, if we consider the vocabulary induced from an N-best list, it is concise and contains most of the useful information. The notion of cohort is generalized to continuous speech, where the search space is restricted to the N-best lexicon. This approach allows us to circumvent the problem of finding cohorts for individual words, which is difficult due to the ambiguity of word boundary determination.

Still, we find the quality of the cohort is unsatisfactory due to over-pruning. This exemplifies another important difference between isolated word recognition and continuous speech recognition. In continuous speech, words have different levels of acoustic-phonetic prominence, due to differences in their information saliency. Most notably, frequent words tend to be reduced acoustically in continuous speech, as also noted by psycholinguistic studies. We conducted empirical studies of the over-pruning errors of the feature-based first stage. We found that the cohort is effective when augmented with a small set of complementary words.

Our experiments on two continuous speech recognition tasks confirm that a second stage recognizer achieves statistically significant improvements in recognition accuracy through searching the reduced space. Improvements are observed when the test sets are clean as well as when they contain artifacts.

We again close this chapter with a discussion on the distribution of computation between the two stages in our framework. The two-stage framework still shows the nice property of delaying detailed acoustic-phonetic analysis to a much smaller space in the second stage.

	Size of the Acoustic Models	Size of the Lexicon
First Stage	1.5M	1,924
Second Stage	4.2M	$\leq$117.5
Baseline	4.2M	1,924

Table 3.4: The distribution of computational load between the two stages in the Jupiter domain.

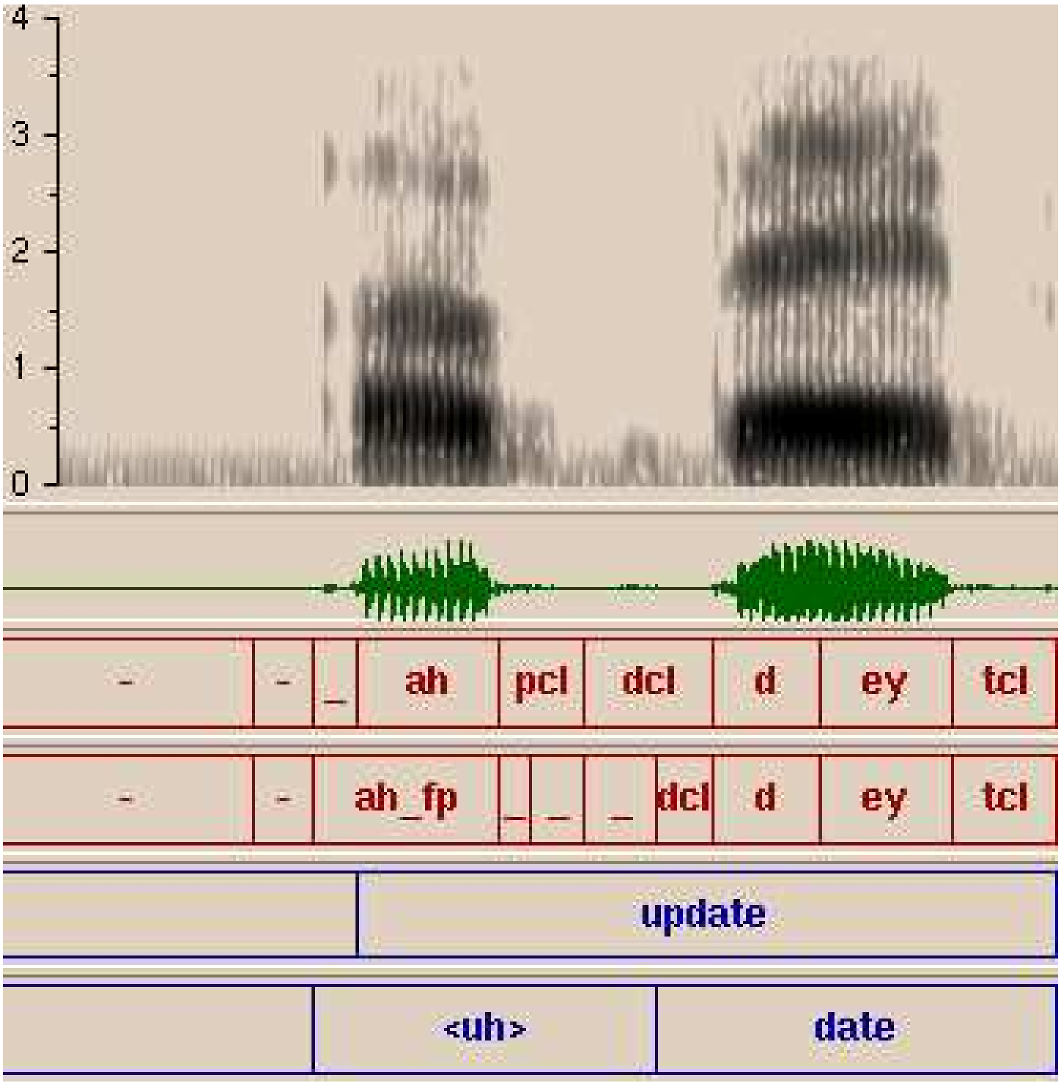


Figure 3-4: The output of the two-stage recognizer (upper panel) and that of the phone-based baseline (lower panel) for the utterance ``update on tornado warnings''. The first stage rules out ``date'' and hence enables the second stage to choose the correct answer.

Chapter 4

Open Vocabulary Recognition

In the previous chapters, we have discussed the implementation of a two-stage speech recognition framework using linguistic features as constraints in the first stage. We have also discussed the application of this framework to an isolated word speech recognition task as well as to a medium-sized vocabulary continuous speech recognition task. In this chapter, we discuss the application of a two-stage framework to the more difficult task of recognizing continuous speech from an open vocabulary.

The challenge of open vocabulary recognition is best explained through an example. The Jupiter domain, discussed in the previous chapter, supports a vocabulary of 1,924 words. Within this vocabulary, it supports about 500 cities and other locations around the world. Within the 1,924 words selected into the Jupiter lexicon, more than half of them (1,011 words) are from a list of 500-plus city names which can be recognized in Jupiter. At the time when the Jupiter system was first designed and deployed, this list included all the cities for which Jupiter had weather forecast information.

Recently, Jupiter has gained access to weather forecast information of more than 30,000-plus US cities and 8,000 international cities provided by WSI [3]. In our discussion in this chapter, we will focus solely on the 30,000-plus US cities to ensure that our results are comparable to recent reported results [50]. However, the methodology we develop in this chapter, as we will see, has a much broader appeal.

One way to support this extended list of US cities is to incorporate the list into the Jupiter lexicon. However, the lexicon associated with this extended city list is overwhelmingly larger than the original Jupiter lexicon. Consequently the search space of the recognizer, due to the expansion in lexicon, will increase dramatically. In this case, the size of the US city-name lexicon (15,606 words) is about eight times the size of the original Jupiter recognizer lexicon (1,924 words). The size of the lexicon, 15,606 words, is smaller than the number of city names (30,000-plus) because words are shared among different city names, such as in *springfield* and *west springfield.*

The complexity also arises as the words in the extended list of city names have a highly skewed distribution. Even a large corpus contains only a small fraction of the city names from the extended list. This makes it very difficult to reliably estimate language model probabilities for city names from the extended list.

In this chapter, we discuss how the two-stage framework we developed so far might provide an effective solution to this type of problem, and how the two-stage system can be enhanced to handle the new challenge. We will discuss how to find the cohort for a word, from within a list of words that are semantically homogeneous to the word in question. This technique is an important extension to the general cohort construction mechanism discussed in Chapter 3.

In this chapter, we will first discuss what we term as *open vocabulary challenges* – the presence of a large and data-sparse vocabulary for speech recognition systems. We then review some recent efforts in dealing with the open vocabulary challenges, as well as the dynamic vocabulary facilities of SUMMIT. Finally, we propose a two-stage approach towards the open vocabulary challenge, in which we use the output of a feature-based first stage to gauge the acoustic similarity of open vocabulary entities. We present experimental results at the end of this chapter.

4.1 The Open Vocabulary Recognition Challenge

We consider the *open vocabulary* problem to be the situation when the speech recognition system is faced with a large lexicon of sparsely-distributed entries. In the Jupiter case, the lexicon associated with the extended US city name list is about eight times the size of the original Jupiter lexicon, as shown in Table 4.1.

	Number of Unique Words	Supported Cities
Jupiter vocabulary	1,924	500+
Extended US city names	15,606	30,000+

Table 4.1: The Jupiter vocabulary and the location vocabulary.

In particular, we are interested in the open vocabulary problem in conversational information interfaces. In general, nouns, verbs and adjectives are major open vocabulary classes. For task-oriented spoken English, noun (especially proper nouns) is the primary and most important open class [18, 68].

We want to analyze how the words from the open vocabulary are distributed in data. As most state-of-the-art speech recognition systems, SUMMIT included, are data-driven, the distribution has important implications for both acoustic modeling and language modeling. For these purposes, we study a one-million word corpus collected at the Spoken Language Systems group through the Jupiter and Mercury interfaces. The corpus contains about 200,000 sentences of real conversation, from users calling our systems. The majority of the corpus is Jupiter sentences. To analyze the distribution of open vocabulary words, we use the extended US city name list as a reference and mark all the instances that appear in the corpus.

Table 4.2 shows that there are 8,159 unique words in the one-million-word corpus. 25% of the lexicon, or 1,885 words, are words related to city names. This empirical evidence supports the commonly-held view that the proper nouns are a major open class and that, in task-oriented spoken English, noun is also one of the most frequent word classes [18, 68].

Entire Corpus	8,159
City Names	1,885

Table 4.2: Number of unique words from the corpus.

Table 4.2 also shows that the one-million-word corpus contains only a small fraction of the entries from the extended US city name list. The 1,885 city name-related words comprises only 12.1% of the 15,606-word lexicon associated with the extended list of US city names.

The reason is because word frequency distribution of open vocabulary words is highly skewed. Again we use the one-million-word corpus in our analysis. There are 8,159 location-related instances we identify from this corpus, containing 1,885 unique words. In Figure 4-1, we show the histogram of log word counts of the 1,885 words. It demonstrates that even within the 1,885-word lexicon, which is a subset of the open vocabulary, the empirical distribution of word counts is highly skewed. Table 4.3 provides some of the descriptive statistics of the empirical word counts. Among the 1,885 words, 457 or 24.2% of them appear only once in this corpus. Because of the skewness, it is impractical to collect enough data so that we can cover the entire open vocabulary and estimate language model probabilities reliably for entries in the open vocabulary.

Mean	73.13
Median	4.5
Max.	9,065
Min.	1

Table 4.3: The descriptive statistics of the empirical counts on open vocabulary words appearing in the one million word corpus.

Lastly, we are interested in the portion of user inquiries that are related to entries in the extended city name list. We count the number of instances of city names that are in the extended US city list but are not in the original 500-plus city list. Table 4.4 shows that the original 500-plus city list covers 95% of all instances of cities in this one-million-word corpus. Admittedly this estimation may have over-stated the frequency of the original 500-plus city list for several reasons. The transcription of the corpus is surely biased towards the original 500-plus city list, particulary since Jupiter can inform the users of what cities it knows in a given region. Still, the message is clear: the expansion from 500-plus cities to 30,000-plus cities affects only a small fraction of user inquiries.

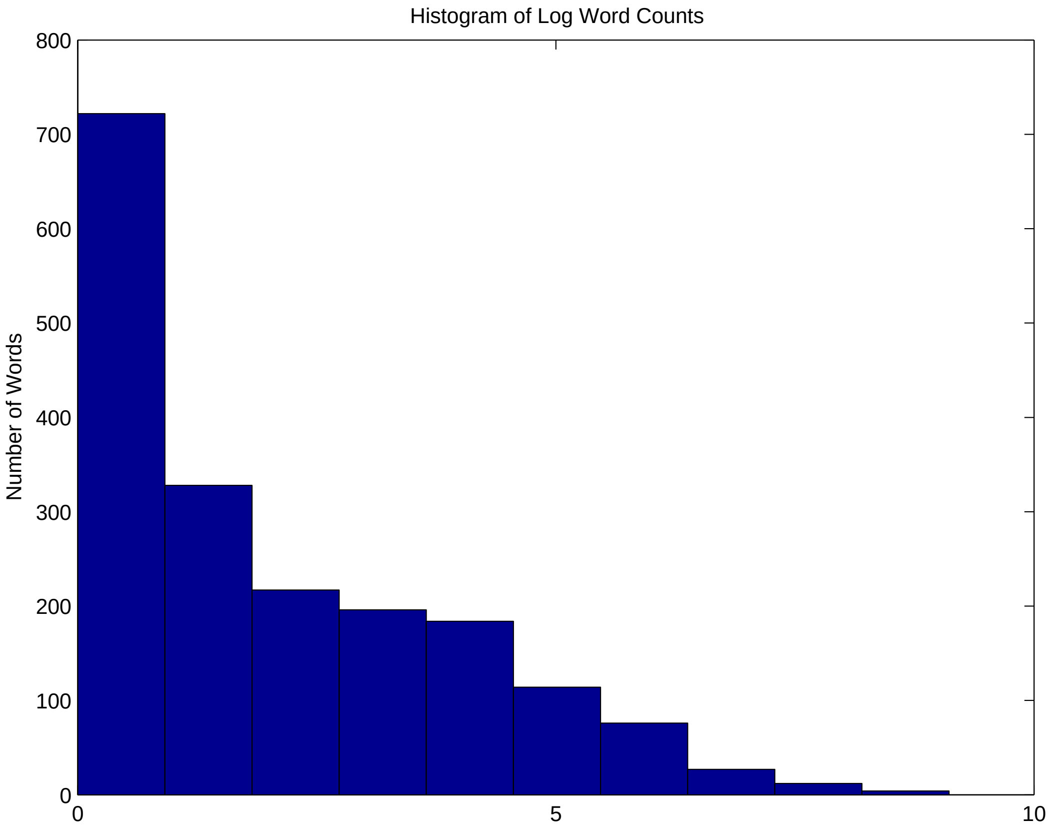


Figure 4-1: Histogram of log word counts of the 1,885 words from the open vocabulary which appear in a one million word corpus in Jupiter. The distribution of word frequency is highly skewed.

4.1.1 Properties of Open Vocabulary

We can sum up the properties of the open vocabulary challenge as follows:

- For task-oriented spoken English, open vocabulary usually stems from semantically homogeneous classes of proper nouns.

- The size of the open vocabulary significantly overwhelms the size of a lexicon that encompasses everything else.

- The word frequency distribution of entries from an open vocabulary is highly

	WER
Locations	4.1%
Words	6.4%

Table 4.4: If we use the original Jupiter lexicon without any support for the open vocabulary, 4.1% of the 8,159 location instances (6.4% if measured by words) from the one-million-word corpus will be treated as OOV words and won't be recognized.

skewed, making it impractical to collect a corpus to cover the open vocabulary.

- Also due to the skewness in distribution, low-frequency words from an open vocabulary affect only a small portion of user inquiries.

4.1.2 Implication for Speech Recognition

A straightforward method to deal with the open vocabulary problem is to add the extended list of city names to the original Jupiter lexicon. By simply adding the extended list to the Jupiter lexicon, we enhance Jupiter's ability to recognize many more cities. It will help to fulfill more of the users' inquiries, as 4.1% of the inquiries involve the open vocabulary. On the other hand, the search space will be increased quite dramatically due to the expansion in the lexicon. Given that inquiries involving the open vocabulary are infrequent, it is hardly economical to increase the lexicon by eight times to handle an extra 4.1% of inquiries (cf. Table 4.4). Due to the skewness and data scarcity, the language model probabilities of most of the words in the open vocabulary will have to be assigned back-off probabilities [55, 92, 104]. For example, we actually use the one-million-word corpus to build language model for Jupiter [117]. As a result, 88% of the words in the open vocabulary will only have a back-off language model probability, making it even less appealing to incorporate them directly into the lexicon.

On the other hand, it is equally undesirable not to support the open vocabulary. We will fail by default 4.1% of all user inquiries. Moreover, in the event when an open vocabulary word is involved, it may pose further difficulty for the recognizer. It has long been observed that out-of-vocabulary words often induce segmentation errors in surrounding words in automatic speech recognition. SUMMIT, for example, uses out-of-vocabulary (OOV) models [6] as a sink for all words that are beyond the vocabulary. Still, the OOV models tend to compete for acoustic observations with in-vocabulary words at decoding time.

4.2 A Two-Stage Approach towards the Open Vocabulary Problem

The two-stage speech recognition framework developed in this thesis provides a promising solution to the challenge of an open vocabulary.

The open vocabulary provides information without which it is impossible to accomplish certain tasks that a conversational interface is designed for. In our example, the task is to provide weather forecast information for various cities. The open vocabulary provides the list of cities to the recognizer, which enables the recognizer to understand the inquiries from users.

What comes with this information is a much expanded search space, and the computational cost of searching in this expanded space at decoding time. Most speech recognition systems adopt the Viterbi decoding algorithm [89, 109]. For practical purposes, a beam search rather than a full Viterbi search is used to manage the computational complexity [46]. Another approach is to use a tree or a stack to speed up the search process [84, 42].

When the open vocabulary is introduced into the recognizer, the search space expands as the size of the open vocabulary increases and, in our example, is overwhelmingly larger than the original search space. In our two-stage speech recognition framework, the first stage uses broad linguistic feature-based acoustic models. Because of this broad representation, the first stage recognizer in our model has a more compact representation of the expanded search space. The result from the first stage recognizer – a cohort – will be transferred for further analysis in the second stage.

The second stage recognizer uses phone-based acoustic models and carries out a detailed acoustic-phonetic analysis in the cohort. The second stage recognizer also benefits from the feature-constraints as an extra source of information. The modern probabilistic speech recognition framework seeks to find the word sequence W that maximizes the conditional probability of $P(W|A)$, where A is the acoustic observations, as in Equation 4.1.

$$\mathbf{W} = \arg\max_{W} P(W|A) \tag{4.1}$$

In our two-stage framework, the second stage searches within the cohort space. One way to characterize this is to consider, that in the second stage, we are looking for the word sequence that maximizes a different conditional probability:

$$\mathbf{W} = \arg\max_{W} P(W|A, F) \tag{4.2}$$

where F represents the constraints from the feature-based first-stage.

Equation 4.2 is important to understand our two-stage framework. The improved performance in a two-stage framework comes directly from the information in **F**, or, to put it another way, by transforming the maximization problem in Equation 4.1 from an unconstrained space to a constrained space in Equation 4.2.

Consider an extreme case where both the first and second stage use the same acoustic models. The second stage recognizer searches within a cohort space generated by the first stage. In this case, F=A. The maximization process in Equation 4.2 is no different from that in Equation 4.1. However, empirically the second stage might have a slightly different result from the first stage recognizer. This is because the beam search scheme is a trade-off between computation and performance and can potentially benefit from a reduced search space.

4.2.1 SUMMIT's Dynamic Vocabulary Facility

In our research on open vocabulary recognition, we utilize the dynamic vocabulary facility recently developed for the SUMMIT recognition system. This technology was initially developed to address the need of dynamically changing grammars in the SUMMIT recognizer while preserving the cross-word phonological constraints [12, 95]. With this facility, we can easily define one or more dynamic classes in the class n-gram.

In SUMMIT, all prior knowledge sources are compiled into a finite state transducer. With the presence of dynamic classes, the FST is broken down into two parts: a static FST and a set of dynamic FSTs, each corresponding to a dynamic class. The static FST contains place-holders for entry into the dynamic FSTs at run time.

4.3 Cohorts for Open Vocabulary Entries

In this chapter we study a particular open vocabulary task on the Jupiter domain. The open vocabulary contains about 30,000 US cities and about 300 cities around the world. As city phrases often contain country names or state names, as in `Boston, Massachusetts` or `Paris, France`, the open vocabulary contains country names, state names and province names as well. In our analysis, we collect all cities, countries, states and provinces, as well as any meaningful combinations of them, and collectively refer to them as the **locations**.

In the previous chapter we extended the two-stage framework for continuous speech recognition tasks. We used the bag of words from the first-stage N-best, augmented with an auxiliary lexicon, as the cohort. This is, in a sense, an aggregate of cohorts for individual words of the underlying utterance. The language model of the second stage is limited to this cohort, which is significantly reduced from the original search space. This is a general method ideal for a medium-sized vocabulary task such as Jupiter or Mercury. However, as one can imagine, this method will have much less impact on the open vocabulary task we are dealing with. This is because the auxiliary lexicon is determined by a data-driven process on the development set.

As discussed earlier, the auxiliary list contains mostly non-content common words that are of less acoustic significance. One could imagine that most open vocabulary entries, due to their infrequent word counts in the data, will not be captured in this process.

In this chapter, we develop a different method to find a cohort *only* for words in the open vocabulary. The auxiliary lexicon method developed in the previous chapter can be used as a complement to the algorithm we discuss here. For simplicity, we will focus only on the open vocabulary words in our discussion in this chapter.

4.3.1 Basic Cohort from the N-best List

We still use a data-driven approach aimed to generate cohorts for words in the open vocabulary. The algorithm has two properties: 1) the cohorts directly reflect acoustic similarities between words in the open vocabulary; 2) across the development set, the cohorts in aggregate minimize expansion of the second stage recognizer. The algorithm runs as follows:

First, we run the first stage recognizer on a development set of utterances. The first stage recognizer has feature-based acoustic models and supports the entire open vocabulary. For each utterance in the recognizer, the first stage recognizer generates an N-best list.

Next we identify all locations in the development set utterances (transcriptions) as well as in the N-best lists generated by the first-stage recognizer. We align the location phrase from the transcription with hypothesized location phrases from the N-best list. If a transcription or an N-best sentence contains no location phrases, we use `NULL` as a place holder.

Note that the recognition by the first stage is done on the entire utterance. However, by limiting the alignments only to the location phrases, we build a cohort for each open vocabulary word we observe in the development set from the N-best list.

4.3.2 Error Correction Mechanism

The N-best list of location phrases naturally form the basic cohort for the location phrase in the transcription. As in the previous chapter, we are interested in an error correction mechanism that can compensate for any errors committed in the first stage. We use alignment to achieve this goal.

We run an alignment algorithm [1] to align the location phrases in the transcription and the N-best lists. Each location phrase in the transcription is aligned against its n counter-parts in an N-best list. The alignment will give us matched word pairs as well as substitutions, insertions and deletions.

For every utterance, we collect the list of open vocabulary words that are in the transcription but are missing from the N-best list. Because we align each location phrase in the transcription with the N hypothesized location phrases in the N-best list, each missing word may have multiple alignment pairs. In particular, in the case of a substitution error, the missing words are aligned with a mistaken word. In the case of a deletion error, the missing words are aligned with `NULL`. In the case of an insertion error, inserted words from the N-best lists are aligned with `NULL`.

For example, an utterance contains the location phrase `MADRID SPAIN`. The first-stage recognizer hypothesizes the following in the N-best list:
MADRID
MADRID SPAIN
MADRID BEIJING
MADRID MAINE
RICHARD SPAIN

The alignment program aligns `MADRID SPAIN` (from the transcript) with the five phrases from the hypothesis. This example gives the following confusion pairs:
{MADRID, RICHARD}
{SPAIN, BEIJING}
{SPAIN, MAINE}

{SPAIN, NULL}

Similar confusion pairs are collected for all the utterances from the development set.

If a word from the transcription is aligned only with `NULL`, it will be included in an auxiliary list. Similar to the auxiliary list discussed in the previous chapter, this list will be used to enhance all the cohorts in the second stage recognizer.

We are left with word substitution errors and insertion errors. We select the substitution pairs in which the hypothesized word appears least frequently among all the substitutions and insertions over the development set. A list of { W_{truth}, $W_{hypothesis}$ } pairs is collected through this process. W_{truth} is the word from the transcription and $W_{hypothesis}$ is the word that the first stage recognizer hypothesizes.

4.3.3 Putting it Together: Cohort for Open Vocabulary words

After the first stage feature-based recognizer generates an N-best list, we construct the cohort for the second stage through the following steps:

- Collect all the open vocabulary words in the N-best list.
- Enhance the list with an auxiliary lexicon developed as part of the error correction mechanism.
- Scan the entire list of { W_{truth}, $W_{hypothesis}$ } pairs also collected as part of the error correction mechanism. If a word in the N-best list matches a $W_{hypothesis}$ from a { W_{truth}, $W_{hypothesis}$ } pair, the W_{truth} in this pair is added to the cohort.

This scheme will ensure that, for the development set, the cohort of the second stage recognizer contains all the open vocabulary words in the transcription, with a minimal expansion of the cohort space. Further, the cohort is now word-specific, compared with the utterance-level aggregated cohort discussed in the previous chapter. The alignment procedure ensures acoustic similarity, as measured by broad linguistic features, of candidates within a cohort.

4.4 Experimental Results

We experiment on our two-stage speech recognition framework again in the Jupiter domain. The test set used in these experiments is the same, standard test set used in Chapter 3. However, the baseline systems used in this chapter differ slightly from the baseline system used in Chapter 3. The new baseline systems support dynamic word classes and use a slightly different set of phonological rules. These new baseline systems have a slightly better performance, measured by word error rates, on the test set. Our experiments in this chapter are based on these new baseline systems.

In the original Jupiter recognition task (without an open vocabulary), the vocabulary is 1,924 words and supports about 500 cities world-wide. Jupiter employs a class n-gram for language modeling. In particular, it contains seven classes that are relevant to locations: `CITY, CITY_STATE, CITY_PROVINCE, PROVINCE, STATE, COUNTRY, CITY_COUNTRY`.

In our two-stage framework, the first-stage recognizer uses feature-based acoustic models. Instead of using seven location-related classes, the seven classes are collected into a new `LOCATION` class. This new `LOCATION` class is also expanded to include the open vocabulary of city-state pairs from the WSI database. The vocabulary of the first stage recognizer is expanded accordingly to support the open vocabulary. This location class undergoes the cohort analysis process in the previous section, for a development set. For the test set, cohorts will be constructed for instances of the location classes as well.

The reason to collapse the seven location-related classes into one `LOCATION` class is because words in these seven classes overlap significantly. Modeling them in one class simplifies the cohort analysis procedure, which is conducted only within the same semantic class. These seven classes are also semantically homogeneous enough to justify a uniform treatment in the first-stage recognizer.

In the second stage, we instead keep the seven location-related classes which are modeled as dynamic classes. The reason is mainly to keep our results comparable to the only other experiment we know that aims to meet the open vocabulary challenge using a multi-pass approach [50]. As the second stage recognizer has an overall much smaller search space, we believe these more refined classes have been developed to improve performance.

Once a cohort is obtained from the first-stage recognition output, we use the cohort to prune the `LOCATION` class. This location cohort is then partitioned into the seven more refined classes in the second stage. Dynamic FSTs are built for each class for each utterance for the second stage recognition.

Our second stage recognizer uses the same phone-based acoustic models as used in baseline systems I and II discussed below.

4.4.1 Baseline System I

The baseline system I is a simple speech recognition system using the 1,924-word lexicon. It does not support the open vocabulary.

4.4.2 Baseline System II

We use as baseline system II a multi-pass system developed by I. L. Hetherington [50]. The baseline system has two stages. It uses the same phone-based acoustic models in both stages. The first stage tries to identify state names from an utterance. All known city-state pairs from the WSI database that match the state(s) identified are included in the second stage. This baseline system differs from our approach in three important ways:

First, this baseline system uses the same acoustic models in the two stages. From our early discussion, the two-stage system may benefit from dividing computational load into two stages. But it will differ significantly from a one-stage system that supports the entire open vocabulary. The results of [50] seem to confirm our analysis. In [50] the two-stage system was compared with a one-stage, static system that incorporated the open vocabulary explicitly and statically. The two systems achieve the same performance, measured by WER, on the standard test set.

Probably more importantly, the baseline system focuses only on the CITY_STATE class in the class n-gram. The operations in the baseline system do not involve any other location-related classes. The baseline system uses very specific semantic constraints – city and state relationships – to control the vocabulary in the second stage recognizer. Such a semantic relationship may be hard to develop or utilize for other tasks.

4.4.3 Initial Experimental Results

To assess the system performance, we compare both the location-related n-gram rules surviving in the second stage as well as the final recognition performance, which are summarized in Table 4.5.

	Our System	Baseline I	Baseline II
Number of rules in the 2nd Stage	797	1,753	2,427
WER	16.8%	17.2%	17.1%

Table 4.5: Results on the open-vocabulary experiments. A rule is a class n-gram rule used in both the baseline and our recognizer, for example, CITY_STATE → boston_massachusetts. We use the number of rules as a proxy to measure the size of the search space in the second stage recognizer.

Baseline system I has altogether 1,753 location-related n-gram rules, including 441 rules of the category CITY_STATE. Each rule is pertinent to a specific CITY_STATE pair. In baseline system II, these 441 CITY_STATE pairs are replaced by, on average, 1,115 CITY_STATE pairs per utterances. These rules are included based on the state(s) identified in the first-stage recognizer of baseline system II. Overall, in the second stage of baseline system II, 2,427 location-related n-gram rules are kept, on average. Comparatively, only 797 such rules remain in the second stage of our system.

When measured by WER, our system also clearly outperforms the two baseline systems. The WER for our system is 16.8%, compared to 17.2% of baseline system I and 17.1% of baseline system II.

We also want to note that the result of Baseline system II was reported in a real-time test and a lot of efforts were invested in Baseline system II to couple the two stages so that the overall system could run in real time. In our implementation and experiments, we were not focusing on the real-time issue so far. If we compare the first-stage recognizer in our system with that of Baseline system II, ours has a smaller search space (due to the broad-class representation of the lexicon) and more compact acoustic models. The second-stage recognizer of our system uses the same set of acoustic models as that of Baseline system II but searches within a smaller search space (cf. Table 4.5). Hence we are confident that the better performance we reported in Table 4.5 will persist if we run a real-time test in the future.

4.5 Discussions

In this chapter, we first analyze the open vocabulary problem that is often encountered in a conversational information system interface. The source of open vocabulary words can be attributed to one or more semantically homogeneous proper noun classes, e.g. city names, people's names, etc. The open vocabulary is important for a task-oriented conversational interface, especially to complete information inquiries. The challenges of an open vocabulary are two-fold. The open vocabulary is often much larger than the remainder of the vocabulary that we need to support in a speech recognizer. It will lead to significant increases in the search space of the speech recognizer. Also the word frequency distribution of the open vocabulary is highly skewed, making it difficult to collect reliable language model probabilities for open vocabulary words, even from a large corpus.

Our two-stage framework is ideal to handle the open vocabulary challenge. With a broad linguistic feature representation, the first stage recognizer search space is much more compact when we incorporate the open vocabulary, as compared with a recognizer armed with more detailed phone-based acoustic models. Moreover, we show that the constraints from the feature-based first stage contribute to increased robustness in a phone-based second stage, which solves a maximization problem on a conditional space: $\mathbf{W} = \arg\max_W P(W|A, F)$.

The cohort construction methods developed in the previous chapter will have little effect on the open vocabulary portion of this task. In fact, the auxiliary lexicon used to enhance the cohort in the previous chapter contains mainly common words that are of relatively less acoustic prominence but of high frequency. Most open vocabulary words, on the other hand, have a very low frequency.

For words in the open vocabulary, a different strategy to find a cohort is proposed. The cohort is still based upon the N-best hypotheses from the feature-based first stage recognizer. However, a confusion analysis is performed *only* for words within the open vocabulary, using alignments from a development set. In this approach, the cohort lexicon is enhance with an auxiliary lexicon based on the acoustic similarities of words *within* the open vocabulary. The fact that the open vocabulary can be defined as a single semantic class in the class n-gram facilitates the cohort analysis process. We tested this method on the Jupiter domain for location-related words. However, it is generally applicable to other potentially open vocabulary classes. We believe this method is more general than methods proposed in [50].

Our results are presented on an open vocabulary task in the Jupiter domain, in which we try to recognize a 30k US city list. Our approach achieves a smaller search space compared to our baseline systems. Our system also achieves a better accuracy, with a WER of 16.8%, compared to the best previously achieved WER of 17.1%.

Chapter 5

Conclusions and Future Work

5.1 Main Results

In this thesis, we study a two-stage approach to automatic speech recognition. The first stage of the recognition system uses linguistic features as constraints and performs coarse recognition on the input speech. The coarse recognition significantly reduces the ambiguity of the speech signal, which is consistent with psycho-linguistic theories. We term the result of the coarse recognition a *cohort*. The second stage of the recognizer performs detailed acoustic-phonetic analysis within the cohort space.

To review our results from a slightly different angle, our studies in this thesis have focused on three aspects, as discussed below.

5.1.1 Linguistic Features as Basic Recognition Units

We model linguistic features as basic speech recognition units, in substitution for the phone unit usually used in a standard SUMMIT recognizer configuration. We define two broad linguistic feature dimensions: manner of articulation and place of articulation.

In our approach, we define linguistic features as broad classes. We calibrate these classes so that speech segments from within the same broad class demonstrate strong acoustic similarity. For example, we divide the traditional vowel classes into three more cohesive sub-classes, *vowel*, *schwa* and *diphthong*. We also treat place of articulation in a somewhat non-traditional way in order to simplify the modeling aspects: we define place of articulation for *all* speech segments using the traditional consonant place system.

We also investigate methods to integrate constraints from the two feature tiers we choose to model. We analyze information fusion schemes that combine the constraints from the manner and place channel at different points in the speech recognition process.

Our approach leads to a very compact representation of the feature-based constraints, compared to models such as the HAMMs or DBNs [54, 100, 120, 72, 114]. Our analysis also shows that the linguistic features we define have clear advantages over phonetic clusters discovered through data-driven methods.

5.1.2 Construction of Cohorts

A cohort is a set of words of a certain degree of acoustic similarity. Words in a cohort can share the same onset and are activated at the same time as in the Cohort model, or they can simply share a particular feature activation, as in the logogen model. For practical purposes for speech recognition, these words are acoustically similar.

While it is relatively easy to find a cohort on an isolated word task, it is much more difficult to determine a cohort for *each word* in continuous speech. For continuous speech, the brain accesses in parallel multiple possible parses of speech. Subtle acoustic characteristics of word onset articulation, which are important for lexical segmentation, are not yet explicitly modeled in current speech recognition systems [56].

The fact that words in continuous speech can exhibit different levels of acoustic prominence, in part due to their frequencies of use, makes the task of finding cohorts for each word in continuous speech a very difficult task. To construct cohorts effectively, we need to distinguish the roles and properties of different word classes. Frequent, non-content words are often less prominent acoustically. As a result, the first stage recognizer often makes mistakes on such words. We have shown that such errors can be overcome by enhancing the utterance cohort with a set of words – common mistakes – identified by data-driven methods. In our study, we also distinguish a particular class of words: the open vocabulary words. For task-oriented conversational interface, proper nouns are important to accomplish the tasks. The number of proper nouns, however, is often significantly larger than that of words that are otherwise needed in a task-oriented conversational interface, and the distribution of word frequency among the proper nouns is highly skewed. For these reasons, they are considered as *open vocabulary* in our study. We analyze a specific *open vocabulary* problem in the Jupiter domain: the recognition of city names. For words in the open vocabulary, we propose a data-driven method to construct cohort, which focuses on the acoustic similarities among words exclusively within the open vocabulary.

For continuous speech, we aggregate the words from the N-best list of an entire utterance as a basic cohort for the *utterance*. We avoid the problem of lexical segmentation in this approach. For open vocabulary words, we model them with special semantic classes in a class n-gram. The cohorts of open vocabulary words are constructed differently. This approach of combining an utterance-level cohort with word-level cohorts, when applicable, proves effective in our experiments.

Error recovery is important to the performance of a two-stage system, especially for continuous speech. The challenge is to prevent the cohort from being over-pruned. The error recovery mechanism also ties with the cohort construction problem.

The design of a sentence-level aggregated cohort provides the first level of error recovery. It generalizes the hypotheses of the first stage to avoid the propagation of certain errors to the second stage.

All our experiments are carried out within SUMMIT, a graph-based probabilistic speech recognition framework. Our experiments, in particular, the construction of feature-based acoustic models and recognizers, are greatly facilitated by the infrastructure SUMMIT provides. On the other hand, it also limits our ability to, for example, explore alternative search schemes based on features.

5.2 Future Research

There are many unanswered questions that call for further investigation in this two-stage framework:

5.2.1 Sub-lexical Modeling

All our analyses of cohorts and two-stage recognition frameworks are carried out on words. In future research, we will examine whether sub-lexical units, such as syllables or the metrical foot, are better candidates for a two-stage framework.

The syllable structure is the basic prosodic unit. Syllables have better invariability compared to phones [33, 34, 43, 76]. There are two types of syllables: weak syllables and strong syllables. The contrast between weak and strong syllables makes the basic metrical tree structure: the metrical foot. The weak/strong property of the syllable is associated with the information contained in it. It has been hypothesized that strong and weak syllables play different roles in lexical access. Strong syllables are engaged in a more complex process that activates and searches the lexicon. This process also provides information for the processing of the surrounding weak syllables, which can be more straightforward pattern matching with help from phonotactic and morphophonemic knowledge [44].

Linguistic features are directly related to syllable structure. For example, the nucleus corresponds to a peak in sonority within a syllable and is usually vocalic. The onset and coda can be optional and are usually consonantal. The structural constraint can easily be expressed with a context free grammar [13, 90]. Here are two examples:

$$\sigma \rightarrow (Onset)\ Rhyme$$
$$Rhyme \rightarrow Nucleus\ (Coda)$$

where σ denotes the syllable, and () denotes optional constituents.

Sub-lexical modeling [96] at the first stage will introduce further constraints which are directly related to the basic linguistic features. It is an important topic to study how to further improve the two-stage framework along these lines.

5.2.2 Effective Implementation

Our study in this thesis can almost be considered as a viability study. We have shown that a two-stage system can achieve very competitive performance in terms of recognition accuracy. However, we have not fully addressed the efficiency issue, although we have shown that the two-stage system has a balanced distribution of computational loads between the two-stages. In our experiments, the detailed, phone-based acoustic models used in the second stage recognizer as well as in the baseline system, are usually 2 or 3 times larger than the feature-based acoustic models in the first stage. The cohort space output by the first-stage is usually 20 to 160 times smaller than the original search space, measured by the size of the lexicon.

For effective implementation, the following topics can be addressed in the future:

Feature-Dependent Acoustic Measurements

In our study, we use the standard landmark-based acoustic measurements for feature-based acoustic modeling. As we perform broad classification along each dimension, we can probably reduce the computational load by using feature-dependent, lower dimensional acoustic representations. For example, place of articulation can be classified using very low dimensional features [82, 108].

Online Processing

In our experiments, we take an off-line approach towards two-stage recognition. We process the N-best list output of the first-stage and create the cohort space for the second stage off-line. For efficiency, we should adopt an online processing strategy in which the second-stage will be tightly coupled with the first-stage in that it searches directly the lattice output by the first-stage.

5.2.3 Continuous Refining Scheme

In essence, the two-stage framework is a continuous refining model for lexical access. At different stages, different information (knowledge sources, constraints) is utilized. In the two-stage framework, we focus on constraints from broad linguistic features in the first stage and more detailed acoustic phonetic information in the second stage.

When we extended the two-stage framework to continuous speech recognition, we designed two different error recovery schemes. A more general, frequency-based scheme helped to recover errors usually associated with function words. We also demonstrated an error-recover scheme based on acoustic similarity, for location-related proper nouns. In this latter process, word-class information and language model constraints are implicitly utilized.

A future research direction is thus to study whether the continuous refining scheme is, for example, dependent on word-classes; or more generally, whether we can build a multi-stage framework where acoustic-phonetic information and higher level information are incorporated in a highly interactive fashion, for example, similar to the logogen model. This highly interactive continuous refining process may also imply a search scheme quite different from the left-to-right search scheme we inherited from the SUMMIT system in our experiments. For example, word candidates can be proposed, based on presence of prominent features *anywhere* in the speech stream, instead of sequentially from left to right.

A related topic is to examine different recognition configurations at different stages of lexical access. In our experiments, the underlying acoustic measurement (MFCC feature vectors) as well as the overall recognizer configuration, for both stages, are built using SUMMIT and are hence, very similar. In the future, we could also explore different venues, for example signal processing algorithms or recognizer configurations (landmark-based or HMM or DBN) at different stages. For instance, we can adapt the signal processing algorithms at the first stage to better match the classification tasks associated with the broad linguistic features.

5.2.4 Integration with Conversational Interface

Another research topic is to integrate the two-stage *recognition* framework with the *conversational* interface. We envisage an open-vocabulary, broader domain conversational interface with which users can interact freely and obtain information from various supported sub-domains.

Instead of deploying multiple speech recognizers, one for each domain, and deciding beforehand which domain the user is going to interact with, we can have the first-stage recognizer search within a general language model which contains keywords from all underlying domains and a small set of common words. By analyzing the cohort from the first-stage recognizer, the second stage can adapt to a domain-dependent language model and better understand the user's inquiry. While the recognition is done in two stages, the dialog itself takes only one turn. As the conversation goes on, the search space in the second stage can be further constrained for improved accuracy and efficiency, by incorporating more and more dialog context. The first-stage recognizer can adapt itself to the dialog context, but can also maintain the general domain-independent grammar. In this way we can achieve the goal of seamless domain switching and provide users with greater flexibility and an interactive experience.

Bibliography

[1] http://www.nist.gov/speech/tools/index.htm.

[2] http://www.sabre.com.

[3] http://www.wsi.com.

[4] D. Archangeli. Aspects of underspecification theory. *Phonology*, (5):183–207, 1988.

[5] J. A. Barnett, M. I. Bernstein, R. A. Gillman, and I. M. Kameny. The SDC speech understanding system. In W.A. Lea, editor, *Trends in Speech Recognition*, pages 272–293. Prentice-Hall, Englewood Cliffs, N.J., 1980.

[6] I. Bazzi. *Modelling Out-of-Vocabulary Words for Robust Speech Recognition.* PhD thesis, Massachusetts Institute of Technology, 2002.

[7] H. Bourlard and S. Dupont. A new asr approach based on independent processing and recombination of partial frequency bands. In *Proc. ICSLP '96*, volume 1, pages 426–429, Philadelphia, PA, October 1996.

[8] H. Bourlard and N. Morgan. A continuous speech recognition system embedding mlp into hmm. In *Advances in Neural Information Processing Systems*, 1989.

[9] H. Bourlard and N. Morgan. Hybrid connectionist models for continuous speech recognition. In C.-H. Lee, F. K. Soong, and K. K. Paliwal, editors, *Automatic Speech and Speaker Recognition.* Kluwer Academic Publishers, Boston, 1996.

[10] F. Chen and V. W. Zue. Application of allophonic and lexical constraints in continuous digit recognition. In *Proc. IEEE Int. Conf. ASSP*, San Diego, CA, March 1984.

[11] N. Chomsky and M. Halle. *The Sound Pattern of English.* Harper & Row, New York, NY, 1968. republished in paperback, Cambridge, MA: MIT Press, 1991.

[12] G. Chung, S. Seneff, C. Wang, and L. Hetherington. A dynamic vocabulary spoken dialogue interface. In *Proceedings of Interspeech 2004*, pages 327 – 330, 2004.

[13] K. W. Church. *Phrase-Sturcutre Parsing: A Method for Taking Advantage of Allophonic Constraints.* PhD thesis, Massachusetts Institute of Technology, 1983.

[14] K.W. Church. Phonological parsing and lexical retrieval. In U.H. Frauenfelder and L.K. Tyler, editors, *Spoken Word Recognition*, chapter 3, pages 53–69. The MIT Press, Cambridge, MA, 1987. The book is a reprint of *Cognition* vol. 25.

[15] G.N. Clements. The geometry of phonological features. *Phonology Yearbook*, 2:225–252, 1985.

[16] R. Cole and others. Speech as patterns on paper. In *Perception and Production of Fluent Speech*, pages 3–50. Erlbaum Ass, 1980.

[17] R. A. Cole and V. W. Zue. Speech as eyes see it. In R. S. Nickerson, editor, *Attention and Performance*, volume VIII, pages 475–494. 1980.

[18] H. Dahl. *Word frequencies of spoken American English.* Gale Group, 1979.

[19] P. Denes and E. Pinson. *The Speech Chain.* Worth Publishers, second edition, 1993.

[20] S. Dupont, H. Bourlard, O. Deroo, V. Fontaine, and J. M. Boite. Hybrid HMM/ANN systems for training independent tasks: Experiments on Phonebook and related improvements. In *Proc. IEEE Int. Conf. ASSP*, 1997.

[21] J. Durand. *Generative and non-linear Phonology*. Longman Group UK Limited., 1990.

[22] E. Eide. Distinctive features for use in an automatic speech recognition system. In *Proc. European Conference on Speech Communication and Technology*, 2001.

[23] J. Elman and J. McClelland. Exploiting lawful variability in the speech wave. In J.S. Perkell and D.H. Klatt, editors, *Invariance and Variability in Speech Processes*, chapter 17, pages 360–380. Lawrence Erlbaum, Hillsdale, N.J., 1986.

[24] L.D. Erman and V.R. Lesser. The Hearsay-II speech understanding system: A tutorial. In W.A. Lea, editor, *Trends in Speech Recognition*, pages 361–381. Prentice-Hall, Englewood Cliffs, N.J., 1980.

[25] L.D. Erman and V.R. Lesser. The HWIM understanding system: A tutorial. In W.A. Lea, editor, *Trends in Speech Recognition*, pages 316–339. Prentice-Hall, Englewood Cliffs, N.J., 1980.

[26] Victor Zue et al. Lecture Notes for 6.345: Automatic Speech Recognition, Spoken Language Systems Group, MIT.

[27] J.G. Fiscus. A post-processing system to yield reduced word error rates: Recognizer output voting error reduction (ROVER). In *Proc. of IEEE Workshop on Automatic Speech Recognition and Understanding*, Santa Barbara, California, 1997.

[28] L. Fissore, E. Giachin, P. Laface, G. Micca, R. Pieraccini, and C. Rullent. Experimental results on large vocabulary speech recognition and understanding. In *Proc. ICASSP '88*, pages 414–417, New York, NY, 1988.

[29] L. Fissore, P. Laface, G. Micca, and R. Pieraccini. Interaction between fast lexical access and word verification in large vocabulary continuous speech recognition. In *Proc. ICASSP '88*, pages 279–282, New York, NY, 1988.

[30] L. Fissore, P. Laface, G. Micca, and R. Pieraccini. Very large vocabulary isolated utterance recognition: A comparison between one pass and two pass strategies. In *Proc. ICASSP '88*, pages 203–206, New York, NY, 1988.

[31] J. Frankel and S. King. ASR - articulatory speech recognition. In *Proc. Eurospeech*, pages 599–602, Aalborg, Denmark, September 2001.

[32] U. H. Frauenfelder and L. K. Tyler, editors. *Spoken Word Recognition.* MIT Press, 1986.

[33] A. Ganapathiraju, V. Goel, J. Picone, A. Corrada, G. Doddington, K. Kirchhoff, M. Ordowski, and B. Wheatley. Syllable - A promising recognition unit for LVCSR. *IEEE Automatic Speech Recognition and Understanding Workshop*, December 1997.

[34] A. Ganapathiraju, J. Hamaker, J. Picone, M. Ordowski, and G. Doddington. Syllable-based large vocablary continuous speech recognition. *IEEE Transactions on Speech and Audio Processing*, 9(4), May 2001.

[35] J. S. Garofolo. *Getting Started with the DARPA TIMIT CD-ROM: An Acoustic Phonetic Continuous Speech Database.* National Institute of Standards and Technology (NIST), Gaithersburgh, MD, 1988.

[36] Heinz J. Giegerich. *English Phonology: An Introduction.* Cambridge University Press, 1992.

[37] J. Glass. A probabilistic framework for segment-based speech recognition. *Computer Speech and Language*, pages 137–152, 2003.

[38] J. R. Glass. *Finding Acoustic Regularities in Speech: Applications to Phonetic Recognition.* Ph. D. thesis, Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, Cambridge, 1988.

[39] James Glass, Jane Chang, and Michael McCandless. A probabilistic framework for feature-based speech recognition. In *Proc. Intl. Conf. on Spoken Language Processing*, pages 2277–2280, Philadelphia, October 1996.

[40] J. A. Goldsmith. *Autosegmental and Metrical Phonology.* Basil Blackwell, 1989.

[41] J. T. Goodman. The state of the art of language modeling. In *HLT-NAACL 2003*, 2003.

[42] P. Gopalakrishnan, L. Bahl, and R. Mercer. A tree search strategy for large-vocabulary continuous speech recognition. In *Proc. ICASSP '95*, pages 572–575, Detroit, MI, May 1995.

[43] S. Greenberg. Speaking in shorthand – A syllable-centric perspective for understanding pronunciation variation. *Speech Communication*, 29:159–176, 1999.

[44] F. Grosjean and J. P. Gee. Prosodic structure and spoken word recognition. In Frauenfelder and Tyler [32].

[45] A. Halberstadt. *Heterogeneous Measurements and Multiple Classifiers for Speech Recognition.* PhD thesis, MIT, November 1998.

[46] N.-Y. Han, H.-R. Kim, K.-W. Hwang, Y.-M. Ahn, and J.-H. Ryoo. A continuous speech recognition system using finite state network and viterbi beam search for the automatic interpretation. In *Proc. ICASSP '95*, pages 117–120, Detroit, MI, May 1995.

[47] J. Harris. *English Sound Structure.* 1994.

[48] I. L. Hetherington. *The Problem of New, Out-of-Vocabulary Words in Spoken Language Systems.* PhD thesis, Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, Cambridge, MA, October 1994.

[49] I. L. Hetherington. An efficient implementation of phonological rules using finite-state transducers. In *Proc. Eurospeech*, pages 1599–1602, Aalborg, September 2001.

[50] I. L. Hetherington. Multi-pass, dynamic-vocabulary automatic speech recognitioin, 2004.

[51] D. P. Huttenlocher. Acoustic-phonetic and lexical constraints in word recognition: lexical access using partial information. Master's thesis, Massachusetts Institute of Technology, 1984.

[52] D. P. Huttenlocher and V. W. Zue. A model of lexical access from partial phonetic information. In *Proc. IEEE Int. Conf. ASSP*, San Diego, CA, March 1984.

[53] D.P. Huttenlocher and V.W. Zue. Phonotactic and lexical constraints in speech recognition. In *Proceedings of the AAAI Conference 1983*, pages 172–176. AAAI, 1983.

[54] J.A.Bilmes. Dynamic Bayesian multinets. In *Proc. 16 Conf. on Uncertainty in Artificial Intelligence*, 2000.

[55] Frederick Jelinek. *Statistical Methods for Speech Recognition.* MIT Press, 1997.

[56] D. W. Gow Jr. and P. C. Gordon. Lexical and prelexical influences on word segmentation: evidence from priming. 21(2), 1995.

[57] Simon King, Todd Stephenson, Stephen Isard, Paul Taylor, and Alex Strachan. Speech recognition via phonetically featured syllables. In *Proc. ICSLP '98*, pages 1031–1034, Sydney, Australia, December 1998.

[58] Simon King and Paul Taylor. Detection of phonological features in contnuous speech using neural networks. *Computer Speech and Language*, 14(4):333–353, 2000.

[59] K. Kirchhoff. *Robust Speech Recognition Using Articulatory Information.* PhD thesis, Der Technischen Fakultät der Universität Bielefeld, 1999.

[60] K. Kirchhoff. Conversational speech recognition using acoustic and articulatory input. In *Proc. IEEE Int. Conf. ASSP*, 2000.

[61] D. H. Klatt. Word verification in a speech understanding system. In D. R. Reddy, editor, *Speech recognition: invited papers presented at the 1974 IEEE symposium*. 1975.

[62] D. H. Klatt and K. N. Stevens. Sentence recognition from visual examination of spectrograms and machine-aided lexical searching. In *Proceedings 1972 Conference on Speech Communication and Processing*, pages 315–318, Bedford, USA, 1972.

[63] D. H. Klatt and K. N. Stevens. On the automatic recognition of continuous speech: Implications from spectrogram-reading experiments. AU 21(3):210–217, 1973.

[64] A. Kornai. *Formal Phonology*. Garland Publishing, 1995.

[65] A. Lahiri and W. Marslen-Wilson. The mental representation of lexical form: a phonological approach to the recognition lexicon. *Cognition*, 38:245–294, 1991.

[66] Wayne A. Lea. Prosodic aids to speech recognition. In Wayne A. Lea, editor, *Trends in Speech Recognition*, pages 166–205. Prentice-hall, Inc., Englewood Cliffs, New Jersey, 1980.

[67] K.-F. Lee. *Automatic Speech Recognition: The Development of the SPHINX System*. Kluwer Academic Publishers, Boston, 1989.

[68] Geoffrey N. Leech. *Word frequencies in written and spoken English*. Longman, 2001.

[69] V. Lesser, R. Fennell, L. Erman, and R. Reddy. The hearsay II speech understanding system. *IEEE Trans. Acoustics, Speech and Signal Processing*, ASSP-23(1):11–24, February 1975.

[70] C. Liegeois-Chauvel, J. B. de Graaf, V. Laguitton, and P. Chauvel. Specialization of left auditory cortex for speech perception in man depends on temporal coding. *Cerebral Cortex*, 9:484–496, 1999.

[71] K. Livescu and J. R. Glass. Segment-based recognition on the Phonebook task: Initial results and observations on duration modeling. In *Proc. European Conference on Speech Communication and Technology*, 2001.

[72] K. Livescue, J. Glass, and J. Bilmes. Hidden feature models for speech recognition using dynamic bayesian networks. In *Proc. European Conference on Speech Communication and Technology*, 2003.

[73] B.T. Lowerre and R. Reddy. The HARPY speech understanding system. In W.A. Lea, editor, *Trends in Speech Recognition*, chapter 15. Prentice-Hall, Englewood Cliffs, N.J., 1980.

[74] W.D. Marslen-Wilson and L.K. Taylor. The temporal structure of spoken language understanding. *Cognition*, 8:1–71, 1980.

[75] W.D. Marslen-Wilson and A. Welsh. Processing interactions and lexical access during word recognition in continuous speech. *Cognitive Psychology*, 10:29–63, 1978.

[76] D. McAllaster, L. Gillick, F. Scattone, and M. Newman. Explorations with fabricated data. In *Proceedings of the DARPA Workshop on Conversational Speech Recognition, Hub-5*, 1998.

[77] J. L. McClelland and J. L. Elman. Interactive processes in speech perception: The TRACE model. In D. E. Rumelhart and J. L. McClelland, editors, *Parallel Distributed Processing: Explorations in the Microstructure of Cognition. Vol. I: Foundations.*, chapter 15. Bradford Books/MIT Press, Cambridge, MA, 1986.

[78] J. L. McClelland and J. L. Elman. The TRACE model of speech perception. *Cognitive Psychology*, 18:1–86, 1986.

[79] H. McGurk and J. W. McDonald. Hearing lips and seeing voices. *Nature*, 264:746–748, 1976.

[80] M.Hwang, R. Rosenfeld, E. Thayer, R. Mosur, L. Chase, R. Weide, X. Huang, , and F. Alleva. Improving speech-recognition performance via phone-dependent vq codebooks and adaptive language models in sphinx-ii. In *Proc. ICASSP '94*, pages I–549 – I–552, Adelaide, Austrailia, April 1994.

[81] J. Morton. Interaction of information in word recognition. *Psychological Review*, 76:165–178, 1969.

[82] P. Niyogi and M. M. Sondhi. Detecting stop consonants in continuous speech. *Journal of Acoustical Society of America*, 111(2):1063–1076, February 2002.

[83] Toshihiko Okada and Shingo Tomita. An optimal orthonormal system for discriminant analysis. *Pattern Recognition*, 18(2):139–144, 1985.

[84] D. B. Paul. An efficient A^* stack decoder algorithm for continuous speech recognition with a stochastic language model. In *Proc. ICASSP '92*, pages 25–28, San Francisco, CA, March 1992.

[85] M. Pavel and H. Hermansky. Information fusion by human and machines. In *Proceedings of the first European conference on signal analysis and prediction*, Prague, Czech Republic, 1997.

[86] Fernando Pereira and Michael Riley. Speech recognition by composition of weighted finite automata. In Emmanueal Roche and Yves Schabes, editors, *Finite-State Language Processing*, pages 431–453. MIT Press, Cambridge, MA, 1997.

[87] J. Pitrelli, C. Fong, S. Wong, J. Splitz, and H. Leung. Phonebook: A phonetically-rich isolated-word telephone-speech database. In *Proc. IEEE Int. Conf. ASSP*, pages 101–104, 1995.

[88] R. Potter, G. Kopp, and H. Green. *Visible Speech.* 1947.

[89] L. R. Rabiner. *A Tutorial on HMM and selected Applications in Speech Recognition*, chapter 6.1, pages 267–295. Morgan Kaufmann, 1988.

[90] M. A. Randolph. *Syllable-based Constraints on Properites of English Sounds.* PhD thesis, Massachusetts Institute of Technology, 1989.

[91] M. Richardson, J. Bilmes, and C. Diorio. Hidden-articulator Markov models: Performance improvements and robustness to noise. In *Proc. Intl. Conf. on Spoken Language Processing*, 2000.

[92] R. Rosenfeld. Two decades of statistical language modeling: Where do we go from here. 2000.

[93] et al S. Basu. Audio-visual large vocabulary continuous speech recognition in the broadcast domian. In *IEEE Workshop on Multimedia Signal Processing*, 1999.

[94] E. C. Sagey. *The Representation of Features and Relations in Non-Linear Phonolgy.* PhD thesis, Massachusetts Institute of Technology, 1982.

[95] J. Schalkwyk, I. Lee Hetherington, and E. Story. Speech recognition with dynamic grammars using finite-state transducers. In *Proc. Eurospeech 2003*, pages 1969 – 1972, 2003.

[96] S. Seneff. The use of linguistic hierarchies in speech understanding. In *Proc. ICSLP*, Sydney, August 1998.

[97] S. Seneff, R. Lau, J. Glass, and J. Polifroni. The MERCURY system for flight browsing and pricing. *MIT Spoken Language System Group Annual Progress Report*, pages 23–28, 1999.

[98] D. W. Shipman and V. W. Zue. Properties of large lexicons: Implications for advanced isolated word recognition systems. In *Proc. IEEE Int. Conf. ASSP*, pages 546–549, 1982.

[99] F. K. Soong and E.-F. Huang. A tree-trellis based fast search for finding the n best sentence hypotheses in continuous speech recognition. In *Proc. ICASSP '91*, pages 705–708, Toronto, Canada, May 1991.

[100] T. A. Stephenson, H. Bourlard, S. Bengio, and A. C. Morris. Automatic speech recognition using dynamic Bayesian netowrks with both acoustic and articulatory variables. In *Proc. Intl. Conf. on Spoken Language Processing*, 2000.

[101] K. N. Stevens. Constraints imposed by the auditory system on the properties used to classify speech sounds: data from phonology, acoustics and psychoacoustics. In T. Myers, editor, *The Cognitive Representation of Speech*. Elviser Science Publishing Ltd., 1982.

[102] K. N. Stevens. *Acoustic Phonetics*. MIT Press, 1998.

[103] K. N. Stevens. Toward a model for lexical access based on acoustic landmarks and distinctive features. *Journal of Acoustical Society of America*, 111(4):1872–1891, April 2002.

[104] A. Stolcke. Entropy-based pruning of backoff language models. 1998.

[105] J. Sun and L. Deng. An overlapping-feature-based phonological model incorporating linguistic constraints: Application to speech recognition. *Journal of Acoustical Society of America*, 111(2), Feburary 2002.

[106] M. Tang, S. Seneff, and V. W. Zue. Modeling linguistic features in speech recognition. In *Proc. European Conference on Speech Communication and Technology*, 2003.

[107] M. Tang, S. Seneff, and V. W. Zue. Two-stage continuous speech recognition using feature-based models: A premliminary study. In *Proc. IEEE Automatic Speech Recognition and Understanding Workshop*, 2003.

[108] Min Tang. Identify stop consonants with highly redundant signal. Technical report, 2002.

[109] A. Viterbi. Error bounds for convolutional codes and an asymptotically optimal decoding algorithm. *IEEE Trans. Information Theory*, IT-13:260–269, April 1967.

[110] A. Waibel and K.-F. Lee. *Readings in Speech Recognition.* Morgan Kaufman Publishers, Inc., San Mateo, California, 1990.

[111] D. E. Walker. Sri research on speech recognition. In W.A. Lea, editor, *Trends in Speech Recognition*, pages 294–315. Prentice-Hall, Englewood Cliffs, N.J., 1980.

[112] C. Wang, S. Cyphers, X. Mou, J. Polifroni, S. Seneff, J. Yi, and V. W. Zue. Muxing: A telephone-access Mandarin conversational system. In *Proc. ICSLP*, Beijing, P.R.China, October 2000.

[113] C. J. Weinstein, S. S. McCandless, L. F. Mondshein, and V. W. Zue. A system for acoustic-phonetic analysis of continuous speech. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 23(1):54–67, Feburary 1975.

[114] M. Wester, J. Frankel, and S. King. Asynchronous articulatory feature recognition using dynamic Bayesian networks. In *Proc. IEICI Beyond HMM Workshop*, Kyoto, December 2004.

[115] S. J. Young, J. J. Odell, and P. C. Woodland. Tree-based state typing for high accuracy acoustic modelling. In *Proceedings ARPA Workshop on Human Language Technology*, pages 307–312, 1994.

[116] V. Zue. The use of speech knowledge in automatic speech recognition. *Proceedings of the IEEE*, 73(11):1602–1615, November 1985.

[117] V. Zue, S. Seneff, J. Glass, J. Polifroni, C. Pao, T. Hazen, and I. L. Hetherington. Jupiter: A telephone-based conversational interface for weather information. *IEEE Transactions on Speech and Audio Processing*, 8(1):100–112, 2000.

[118] V. W. Zue and R. A. Cole. Experiments on spectrogram reading. In *Proc. IEEE Int. Conf. ASSP*, pages 116–119, 1979.

[119] V. W. Zue and D. W. Shipman. Properties of large lexicons: Implications for advanced isolated word recognition systems. In *Proc. 103rd Meetings of the Acoustic Society of Amrica*, Chicago, IL, April 1982.

[120] G. Zweig and S. J. Russell. Speech recognition with dynamic Bayesian networks. In *Proc. AAAI*, 1996.