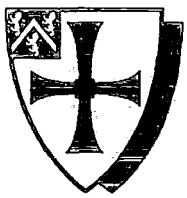
Towards an automatic speech recognition system for use by deaf students in lectures

Collingham, Russell James

University of Durham



Towards an Automatic Speech Recognition System for use by Deaf Students in Lectures

Russell James Collingham

Laboratory for Natural Language Engineering
Department of Computer Science

September 1994

Submitted in partial fulfilment of the requirements for the degree of

Doctor of Philosophy

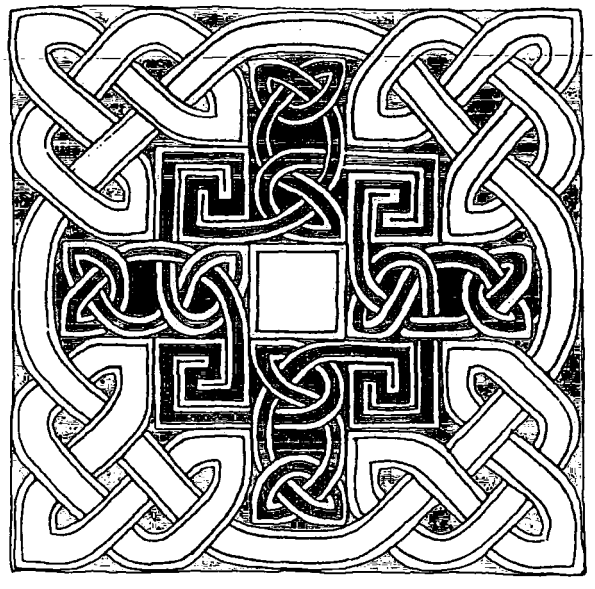
This thesis is dedicated to my family who have given me so much love and support throughout my life...

Yvonne

June and Glyn

Gregg, Janet and Sarah Jane

Carol, and David who is sadly missed



Abstract

According to the Royal National Institute for Deaf people there are nearly 7.5 million hearing-impaired people in Great Britain. Human-operated machine transcription systems, such as Palantype, achieve low word error rates in real-time. The disadvantage is that they are very expensive to use because of the difficulty in training operators, making them impractical for everyday use in higher education. Existing automatic speech recognition systems also achieve low word error rates, the disadvantages being that they work for read speech in a restricted domain. Moving a system to a new domain requires a large amount of relevant data, for training acoustic and language models.

The adopted solution makes use of an existing continuous speech phoneme recognition system as a front-end to a word recognition sub-system. The sub-system generates a lattice of word hypotheses using dynamic programming with robust parameter estimation obtained using evolutionary programming. Sentence hypotheses are obtained by parsing the word lattice using a beam search and contributing knowledge consisting of anti-grammar rules, that check the syntactic incorrectness of word sequences, and word frequency information. On an unseen spontaneous lecture taken from the Lund Corpus and using a dictionary containing 2637 words, the system achieved 83.5% words correct with 15% simulated phoneme error, and 73.1% words correct with 25% simulated phoneme error. The system was also evaluated on 113 Wall Street Journal sentences.

The achievements of the work are a domain independent method, using the anti-grammar, to reduce the word lattice search space whilst allowing normal spontaneous English to be spoken; a system designed to allow integration with new sources of knowledge, such as semantics or prosody, providing a test-bench for determining the impact of different knowledge upon word lattice parsing without the need for the underlying speech recognition hardware; the robustness of the word lattice generation using parameters that withstand changes in vocabulary and domain.

Acknowledgements

There have been many people who have helped me with the work described in this thesis. I would especially like to thank Colin Davies and Margaret Collins from the Durham University Service for Hearing Impaired Students; Richard Wright and Angela King from the Royal National Institute for Deaf people; Roger Moore from the Defence Research Agency; Tony Robinson from CUED; and my colleagues in the Department of Computer Science (past and present), particularly Richard Morgan, Malcolm Munro, Keith Bennett and Barry Cornelius.

Financial assistance with the purchase of equipment has been provided by the University of Durham, Transtech Devices Ltd. and the RNID.

My colleagues in the Laboratory for Natural Language Engineering have provided me with tremendous support, both intellectually and socially and I would like to thank all of them, particularly Kevin Johnson, David Nettleton and Yang Wang. E Luisa, un'amica meravigliosa, grazie mille per la tua amicizia e la felicita che ha dato a me.

My family, especially my wife Yvonne, have had to put up with my many long hours of work, and I thank them for their endless patience, support and love.

Finally, I come to the person to whom I owe the greatest thanks, the leader and main driving force of our research group, Roberto Garigliano. Over the eight years that we have known each other you have been many different things to me, teacher, supervisor and now colleague, but you have always been a good friend, grazie tante.

Declaration

The material contained within this thesis has not previously been submitted for a degree at the University of Durham or any other university. The research reported within this thesis has been conducted by the author unless indicated otherwise.

Contents

List of Tables

List of Figures

Chapter 1

Methodological Introduction

This chapter presents a clarification of some methodological issues in relation to the position of this research in the current field of computer science. This is followed by a discussion of the criteria for success of the research and a description of the logical progression of the thesis.

1.1 Methodological Issues

This section introduces the methodological framework within which the research described in this thesis was undertaken. This work is in the branch of computer science known as artificial intelligence. The particular area of research is in natural language engineering. The discussion of methodological issues is presented in general terms in this introductory chapter. Specific methodological issues that arose during the progress of this work are described in chapters 5 and 6.

1.1.1 Artificial Intelligence

There are many definitions of artificial intelligence (AI). One definition states that AI is

> *... the field of research concerned with making machines perform tasks which are generally thought of as requiring human intelligence.*
>
> [Beardon, 1989]

in other words, simulating human behaviour from an external view. This could be further refined to simulating *successful* human behaviour because it is unlikely that we want, for example, a machine that stutters or makes spelling mistakes. In fact, circumstances do exist when imperfect behaviour might be required, for example when deliberately trying to fool a human into believing that a computer system is another human, as is the goal for the "Turing test" competition.

A distinction has to be made between AI and cognitive science. Cognitive science is the study of the human cognitive process, in other words internal human behaviour, using computer programs as an experimental test-bench. The distinction is that AI aims to achieve a simulation of human behaviour by any available technique, not by only modelling the human cognitive process. For example, an AI approach to computer vision may make use of radar and sonar, whereas the cognitive science approach would model the human vision mechanism. There is, therefore, no obligation to simulate external human behaviour using only human mental techniques, although analysis of the cognitive approach to a particular problem may give a better understanding of that problem, or provide a possible starting point for developing alternative solutions.

More often than not, simulating intelligent human behaviour involves achieving at least as good a performance as a human. In some cases, however, it may be possible for a computer to improve on human performance. For example, a computer vision system may perform better in darkness than the human vision system, or a computer may react to audio stimuli that are outside the range of human hearing.

1.1.2 Natural Language Engineering

The research described in this thesis has been developed according to the principles of Natural Language Engineering (NLE). This is a new approach to natural language processing, with respect to the traditional computational linguistics one, and has been acknowledged by the EEC in their LRE programme as the approach most likely to bring substantial benefits in the medium term to end users.

NLE has been described in the Technical Background Document of the Linguistic Research and Engineering European Programme (LRE) as follows

> *Linguistic Engineering (LE) is an engineering endeavour, which is to combine scientific and technological knowledge in a number of relevant domains (descriptive and computational linguistics, lexicology and terminology, formal languages, computer science, software engineering techniques, etc.). LE can be seen as a rather pragmatic approach to computerised language processing, given the current inadequacies of theoretical computational linguistics.*
>
> [EC, 1991, page 7]

NLE is a pragmatic approach characterised by a readiness to use any means in order to build serious speech and language processing programs: this means taking advantage of existing linguistic and logic theories where they exist and are suitable, and then developing localised theories, using knowledge bases, statistical and adaptive methods, and even *ad hoc* solutions where everything else has failed.

A definition of computational linguistics is as follows:

> *Research in computational linguistics (CL) is concerned with the application of a computational paradigm to the scientific study of human language ...*
>
> [Ballard and Jones, 1990, page 133]:

The traditional computational linguistics approach has been to seek an understanding of the entire process of natural language comprehension and develop a unified

theory of language understanding. A common criticism of applications developed using this approach is the inability to process realistic material:

> *Computational linguistics research in practice tends to revolve round little "toy" subsets of artificially simple linguistic forms, in the hope that systems which succeed in dealing with these may eventually be expanded and linked together until they cover entire languages.*
>
> [Sampson, 1987, page 17]

The goal of NLE is to produce systems which are large in scale, allow easy integration and expansion, are feasible both in terms of speed and of memory, are maintenable, robust and such that the intended users are able and willing to use them.

There is at present a large community, of both academics and people from industry, that is interested in the pragmatic approach of NLE and its potential benefits. Research using the NLE paradigm is also being undertaken at the Universities of Edinburgh and Sheffield. The European Community predicts that the market for NLE products will be 10 million users in the next few years, and has launched several large programmes (EUROTRA, LRE). The American Defence Research Agency, ARPA, is investing heavily in a programme for text scanning (MUC), and several national governments have similar programmes. The commercial market is predicted to grow rapidly [Ovum, 1991] and the traditional engineering and computer science organisations are showing great interest in NLE. A forthcoming conference on Applied Natural Language Processing (ANLP-94), the fourth in a series sponsored by the Association for Computational Linguistics, aims to bring together researchers and developers, who collectively use a wide range of language engineering techniques, to focus on the application of natural language processing to real problems. Cambridge University Press have recently launched the *Journal of Natural Language Engineering*, whose principal aim is to bridge the gap between traditional computational linguistics research and the implementation of practical applications with potential for real world use.

1.1.3 Symbolic and Sub-Symbolic Processing

The traditional approach to artificial intelligence involves the construction of representational formalisms and the development of corresponding search mechanisms. The guiding principle of this representational methodology is the physical symbol hypothesis, which states:

> *A physical symbol system has the necessary and sufficient means for general intelligent action. By "necessary" we mean that any system that exhibits general intelligence will prove upon analysis to be a physical symbol system. By "sufficient" we mean that any physical symbol system of sufficient size can be organized further to exhibit general intelligence. By "general intelligent action" we wish to indicate the same scope of intelligence as we see in human action: that in any real situation behaviour appropriate to the ends of the system and adaptive to the demands of the environment can occur, within some limits of speed and complexity.*
>
> [Newell and Simon, 1976]

The physical symbol hypothesis is only a hypothesis, it cannot be proved or disproved on logical grounds, so it must be subjected to empirical validation. Computers are a perfect medium for this experimentation.

The most significant challenge to the symbolic approach came from the adaptive approach to machine intelligence, initially through parallel distributed processing. The two major branches of adaptive or sub-symbolic processing are the statistical approach, based upon Bayesian statistics, and the machine learning approach, based upon the principle of evolution or the principle of neural processing. A sub-symbolic approach to knowledge representation is one in which the emphasis is not on the use of symbols to represent objects and relations, but instead on the collective behaviour produced by the interaction of a number of simple interacting components.

Evidence supporting the sub-symbolic approach to AI does not necessarily invalidate the symbolic approach — there is often more than one way to accomplish a task. Indeed, under the principles of natural language engineering, a hybrid

solution combining both symbolic and sub-symbolic approaches may be adopted, rather than arguing for one approach over the other in all possible applications:

> *... it is widely believed that there are some activities of intelligence (e.g. recognition of multidimensional patterns) where an approach operating at some lower level than a level of description in symbols is more appropriate than the traditional logical-symbolic approach.*
>
> [Calmet and Campbell, 1993]

To pursue a solution to a problem purposefully using only one approach (symbolic or sub-symbolic) is not really an attempt at finding the best possible solution, rather, it is to research the limits of the approach being used. Having an open mind as to which techniques are better suited to particular problems is the method of investigation adopted in this thesis.

Consider as an example the game of chess. How would we teach a machine to play a good game of chess? The sub-symbolic approach would be to allow the machine to learn how to play well from the experiences of playing many games of chess. The symbolic approach would be to take advantage of the many centuries of experience gained by chess masters over the years, represented by rules: for example, standard opening and endgame scenarios, controlling the centre of the board, and optimum positioning for key pieces. In reality, the leading chess playing computers of today do contain a vast amount of standard knowledge accumulated by chess experts, yet incorporate a certain amount of adaption to the particular characteristics of their opponent.

Another example would be learning to drive a car. When we first learn to drive a car there are a number of rules that must be learnt (symbolic): for example, the highway code and leaving the car in neutral while waiting at traffic lights. However, to develop into a good driver it is necessary to drive in many diverse situations, learning from different sensory experiences, learning to drive in sympathy with the car and the like (sub-symbolic).

The approach that we take is to use the symbolic approach where acceptable behaviour in a reasonable time is achieved, and make use of appropriate sub-symbolic

techniques at other times.

1.2 Criteria for Success

The criteria for the success of the work described in this thesis can be described in terms of the goals of natural language engineering:

Scale : the system should be a large scale system that has a large vocabulary; a large vocabulary is one that contains over 1000 words;

Robustness : the system should be robust enough to handle general spoken English in the form used in university lectures; it should demonstrate domain independence; some preparation is allowed, for example in vocabulary selection;

Integration : the system should allow ease of integration with other sources of knowledge;

Feasibility : hardware requirements should not be too high, execution speed should be acceptable;

Maintenability : the system should be useful over a long period of time, and be flexible to changes in functionality (adaptive maintenance);

Usability : the system should be useful to deaf people studying at university, and achieve an acceptable level of recognition in a reasonable time;

Techniques : the system should use existing theories, or where none exist, use newly developed theories, in addition to any other technique (such as statistical) from the fields of engineering and artificial intelligence.

These criteria are the goals for an ideal NLE system. The work in this thesis does not in any way claim to provide a complete solution to the problems of automatic speech recognition. As such, an improvement, towards the goal of an automatic

speech recognition system that can aid deaf students, in any of these categories over current approaches or current systems can be termed a success. For this work, the three most key criteria are scale, robustness and usability.

1.3 Logical Progression of the Thesis

The thesis is organised according to the following plan.

Chapter 1 address some important methodological issues in relation to the position of the work within the field of computer science; followed by a discussion on the criteria for the success of the research; and finally the organisation of the thesis.

Chapter 2 introduces the concept of human-machine communication and explains the problem being addressed in this thesis: decoding a sequence of phonemes into words, using as little domain specific information and imposing as few restrictions on the speaker as possible; the problem is defined using the taxonomy outlined in chapter 3. The need for a solution and the potential benefits to deaf students are also discussed.

Chapter 3 describes a typical automatic speech recognition system; followed by a discussion of the current trends in the field of automatic speech recognition research; and a taxonomy is introduced for describing speech recognition systems. The chapter concludes with a look at some possible future trends in automatic speech recognition research.

Chapter 4 outlines the capabilities of existing systems for automatic recognition of continuous speech, and of existing systems used by the deaf community for real-time machine transcription of speech.

Chapter 5 outlines the general solution adopted to the problem described in chapter 2: phoneme recognition, word lattice generation and word lattice parsing. The novelty of the solution is also addressed.

Chapter 6 describes in detail the solution outlined in chapter 5: phoneme recognition using a simulation and also the AURIX and CU-CON systems; word lattice generation using dynamic programming with robust parameter estimation obtained using evolutionary programming, and the system dictionary; and word lattice parsing using a beam search and contributing knowledge such as the anti-grammar and word frequency information. A detailed discussion of the anti-grammar is presented. The software engineering aspects of the test-bench are also addressed with reference to integration of new knowledge sources and maintenability of the underlying representations.

Chapter 7 outlines the framework in which the work described in this thesis is evaluated. Addressing in particular: phoneme recognition assessment, word lattice quality, the suitability of the anti-grammar, word recognition assessment, execution times and readability issues. The problem of evaluating spontaneous speech recognisers is also discussed, and a case for developing a new measure for assessing such recognisers is presented. A brief mention is made of the early work in this area.

Chapter 8 gives details of the data that was used for evaluation purposes and presents results for the areas outlined in chapter 7.

Chapter 9 will conclude the thesis by checking if this work has met its criteria for success; discussing future research directions; and describing what this work can offer researchers in the field of automatic speech recognition and also what it can offer the deaf community.

Appendix A lists the anti-grammar rules used as contributing knowledge during word lattice parsing.

Appendix B shows the recognition output of the system on the first 31 sentences of the Lund corpus lecture and the first 40 WSJ sentences that were used for evaluation purposes.

The guiding methodological paradigms have now been described. The progres-

sion of the remainder of the thesis follows the order: problem definition, current state of the art, general solution, detailed solution, evaluation framework, results and conclusions.

Chapter 2

Analysis of the Problem

This chapter introduces the concept of human-machine communication and explains the problem being addressed in this thesis: decoding a sequence of phonemes into words, using as little domain specific information and imposing as few restrictions on the speaker as possible. The need for a solution and the potential benefits to deaf students are also discussed.

2.1 Introduction to Communication

2.1.1 Human Communication

Animals use all five senses (hearing, seeing, smelling, tasting and touching) as well as body language to communicate with each other. This can range from aggression towards an incoming predator, to tenderness during the mating season. For their level of communication needs, however, hearing is no more important than any of the other methods.

In contrast, human communication, which often involves the transfer of very complex information, relies heavily on speech and hearing. Although writing too is

important, and has the advantage of lasting longer (i.e. something written can be repeatedly read at different times), it is not as uniformally used or as immediate as speech. For humans, speech is the output channel that achieves the highest rate of communication, yet hearing is not the best input channel. The best channel for human reception of information is vision.

2.1.2 Human-Machine Communication

Human-machine communication is dominated by typing; not for the reason that producing words by means of fingers is better but because of the inability of machines to understand speech. Three methods of possible human-machine communication are described below:

TYPING: Typing is a very accurate method of communication; errors that occur are caused by the typist. Skilled typists can work at 100–150 words per minute, an unskilled typist can work at 10–25 words per minute. Becoming a skilled typist requires a considerable amount of training. What can be typed is limited by the design of the keyboard. Modern software packages utilise multiple key-presses and mouse control to select certain functions, but these only slow down operating rates.

WRITING: Handwriting is a more universal skill than typing, however it is a slow means of communication with a speed of only about 25 words per minute. Machine recognition of handwriting is complicated by the fact that it is so non-uniform: no two people have the same handwriting. Although suffering from many of the disadvantages associated with speech, it is a much slower method of communication.

SPEAKING: Speaking rates vary from about 120 to 250 words per minute making this potentially the fastest form of human-machine communication. Speech is easily learned as a child and is the most natural form of human communication.

ADVANTAGES	DISADVANTAGES
◦ Most natural form of communication between people — familiar, convenient and spontaneous.	◦ Natural, yet unrecognisable sentences may be spoken.
◦ Requires no training — people can speak, but not in all cases can they type or write efficiently.	◦ Need to constrain utterances to those recognisable by machine — dependent on the application.
◦ Human's highest capacity output channel.	◦ Speaking rate is slowed down by pauses or unfamiliarity.
◦ Allows simultaneous methods of communication — hands and voice, for example.	◦ Could confuse computer by speaking something to another human.
◦ Allows simultaneous communication to humans and machines.	◦ Lack of privacy if other humans are present.
◦ Possible in darkness, around obstacles and for the blind or handicapped.	◦ Sensitive to dialects and differences in pronunciation.
◦ Permits the verification of a speaker's identity.	◦ Interfering "noise" can make accurate recognition difficult.
◦ Requires no panel space, displays or complex apparatus.	◦ Microphone must be worn or held (closely to avoid "noise").
◦ Possible at a distance and at various orientations.	
◦ Permits simultaneous use of hands and eyes for other tasks.	
◦ Permits telephone to serve as a computer terminal.	

Figure 2.1: The Advantages and Disadvantages of Using Speech Recognition for Human-Machine Communication

Speech, therefore, is potentially the best method for a human to communicate to a machine and visual display should be used for a machine to communicate to a human. It is interesting to note that machine-machine communication using speech would be extremely inefficient. The bounds of machine-machine communication are being pushed further and further to their potential maximum limit. Recent advances in optical technology mean that machines can communicate at speeds much faster than those allowable by voice, or even electrical means.

A summary of the advantages and disadvantages of human-machine communication by speech recognition are outlined in Figure 2.1 [Lea, 1980, Page 5].

2.2 The Basic Problem of Automatic Speech Recognition

It is useful to remind ourselves of the complexity of the task by considering our own human performance. We are Olympic standard talkers (never mind the content), and when we need to be, we are expert listeners. We exchange concepts and meanings (semantics) about various topics (pragmatics), using a spoken language which consists of known words in accepted orders (syntaxes). We can break words down into sub-units such as syllables (morphemics). We have a knowledge of the basic sounds of our language, and we can describe or label them (phonetics). We also have knowledge of acoustics — "Madonna has a clear, high pitched voice". In exchanging thoughts, we use all this knowledge at all times, and we need to, since the data at every one of these levels is variable for any concept. We express the same concept in multiple ways, using different sentences of different words. Also, any given word is pronounced differently each time we use it, depending on its place in a sentence and on the speaker, resulting in different acoustic streams for the same word. Spoken language is full of starts and restarts, 'ums' and 'ahs', and incomplete sentences. Yet we are able to decode this single-goal variable data in each instance and can use the variability to identify speakers and styles of speaking. When in doubt, we can ask questions for clarification.

[Fallside, 1989]

There can be no doubt that automatic speech recognition is one of the most difficult "human-impersonation" tasks demanded of a computer. The ideal scenario is of any person, talking about anything, into an unobtrusive microphone, under any conditions (for example over the phone, at a railway station or with a cold), having their exact words immediately recognised. What happens after this step is a further problem, but could include a visual or typed reproduction, or result in some action being taken in response. The latter will involve some understanding of what is spoken.

The *current* reality of existing systems for automatic speech recognition is very different, and this is described in detail in section 4.2.

2.3 Description of the Problem

The problem that we are addressing in this thesis is that of decoding a sequence of phonemes into words, using as little domain specific information and imposing as few restrictions on the speaker as possible. The intention is to create a general purpose sub-system for reducing the large search space involved in automatic speech recognition. This sub-system can be used in isolation or in combination with other knowledge sources (such as semantics) for word recognition.

Research into the phoneme recognition system, used as a front-end to the word decoding sub-system, does not fall within the scope of this thesis. Phoneme recognition results of systems suitable for use as a front-end to this research are described in section 4.2. These results demonstrate the feasibility of this approach — high phoneme recognition rates can be achieved, making the results obtained in this work realistic.

The framework within which this sub-system has been built is that of developing an automatic speech recognition aid for use by deaf students in university lectures. The style of the speech encountered is that of monologue, and although a certain amount of question and answering between a lecturer and the students does occur in lectures, it is beyond the scope of this research.

The problem will be described using the taxonomy developed in section 3.2.

2.3.1 The Speaker

The problem of speaker dependence is the responsibility of the phoneme recogniser. The research described in this thesis makes no assumptions in this area. It is usually the case that a speaker dependent system performs better than a speaker independent one, although the problem of enrolment has led to more speaker independent systems being developed. It would not be too inconvenient if a speaker dependent system were developed in this case, as after a single short enrolment

period, each lecturer would use the system many times.

2.3.2 The Connectedness of Speech

The connectedness of speech used in this research is continuous speech. Within the framework of normal university lectures, a speaker addresses a group of students at the normal rate of human speech. Any system that is to aid deaf students must be usable in normal university lectures, so no impositions can be made upon a lecturer, apart from the wearing of a headset microphone. During the analysis of some lectures within the Durham corpus, it was calculated that the upper level on the average number of words spoken per minute is 100.

2.3.3 The Speaking Style

The speaking style used by lecturers lies somewhere between read and spontaneous speech. It is neither completely spontaneous, because a lecture is a prepared monologue, nor is it completely read, because a lecture although prepared is not scripted word for word.

2.3.4 The Unit of Speech

The unit of speech, in other words the interface between the acoustic-phonetic unit and the word lattice generation unit, is the phoneme. Choosing a lower level unit (such as allophone) would have meant more research into the field of phonology, of which the author has little experience. Choosing a higher level unit, for example words in the form of a word lattice, might have been suitable. This was not adopted for two reasons. Firstly, word lattices were not a common intermediate data structure when this research began. This made it difficult to find a group able to build a suitable system. Secondly, such a system would have been very

inflexible to use because it would have required a huge amount of training data for *each* given domain.

The choice of the phoneme as the unit of speech allows us to change domain and vocabulary easily without the need to retrain the underlying speech recognition hardware.

2.3.5 The Language

The language used by lecturers lies somewhere between restricted and unrestricted (see section 3.2.5), tending more towards unrestricted. In general, fragments of grammatically correct English will be used interspersed, because of the speaking style, with speech repairs. Studies into the nature of university lectures have shown that 32% of spoken sentences contain repair [Johnson *et al.*, 1994a].

This has an effect on the type of grammar that we are able to use during the recognition process. A formal grammar of (written) English is not appropriate because of the unrestricted nature of the speech being recognised in the lecture scenario. Nor is it possible to collect a large amount of lecture data in order to train an n-gram language model, because of the spontaneity of the speech.

2.3.6 The Level of Recognition

The level of recognition required by this research is one of the constraints that has been relaxed in an attempt to obtain a useful and working system. Experience in the development of Palantype (see section 4.3.1 showed that a 75% correct transcription was very useful to well motivated deaf people. The level of recognition that it is hoped will be achieved is at least 75% words correct.

2.3.7 The Vocabulary

The vocabulary size of the current system is approximately 2600 words. This is an arbitrary figure and could be much higher, with a corresponding reduction in performance. During the analysis of some lectures within the Durham corpus, it was calculated that one lecturer used only 1100 unique words during the whole of a two lecture course fragment.

2.3.8 The Speed of Recognition

Clearly, the speed of recognition needs to be on-line so that a deaf student may "keep up" with the topic at any point in a lecture. This is at the cost of a lower than verbatim level of recognition. Should the purpose of transcription be note-taking, it could be possible that a second, off-line, attempt is made at the recognition to try and construct a more accurate record of a lecture. Off-line recognition has not, however, been developed in this thesis.

2.4 The Need for a Solution

According to the Royal National Institute for Deaf people (RNID) there are nearly 7.5 million people in Great Britain with some degree of hearing loss. From this figure it is possible to estimate the number of hearing impaired people who attend universities around the country. A further significant proportion of hearing impaired people are prevented from attending higher education because of a lack of support facilities. Hearing aids are only really effective in quiet environments when used close to the person speaking. A hearing aid cannot replace the damaged ear's ability to discriminate speech and consequently many people with severe and profound losses hear speech but cannot understand it.

The most common form of communication between a hearing person and a deaf

person is lip-reading. Unfortunately this is not possible when more people become involved. It may be possible to employ an interpreter to act as an intermediary between one or more hearing people and several deaf people. But there are several methods of communication employable by an interpreter (British Sign Language, American Sign Language, Sign Supported English, for example), and each interpreter would have their own particular style of signing which would take time to adjust to, possibly causing a deaf person to miss some information. It would also be impossible for a deaf person to make notes on what is being said whilst carefully watching the interpreter.

Machines that can recognise and display speech would be beneficial to deaf people. The profoundly deaf may be interested in such a machine in situations where lip-reading is difficult, for example over the telephone. Other possible uses, which would also benefit the hard of hearing, are at church services, public meetings or lectures. In recent discussions on technology, deaf and hard of hearing people indicated three major areas in which they hope to see automatic speech recognition applied: telephone communication, face-to-face communication, and captioning of television and films [Harkins, 1988]. More than 24 million people in the United States are deaf or hard of hearing. The idea of a "little black box" that will recognise and display all speech, although desirable, is certainly not achievable within the next five years, despite huge technological advances in computing during the past decade.

At a recent symposium [RNID, 1990], Ross Trotter, from the National Association for Deafened People, outlined a deaf user's *ideal* requirements of an automatic speech recognition system. These have been enlarged upon below.

1. **SPEED**

   The system should produce a visual display of what is spoken in real-time. Within five seconds is acceptable, but 15 seconds is too long; by this time the speaker may have changed topic, making any questions by the deaf person out of place, or the deaf person may experience difficulty following displayed slides, or in lip-reading, if the system is not displaying what the speaker is

currently saying.

2. **CLARITY**

   The displayed text should consist of English words with phonetics, or "sound-spelled" words, kept to a minimum. Many born-deaf people do suffer from English comprehension difficulties, and cannot possibly cope with non-English words. The system should thus show a level of approximately 90% word accuracy.

3. **SPEAKER-INDEPENDENCE**

   The system should be as speaker-independent as possible; although for some applications a minimal enrolment period would be acceptable.

4. **OPTIONAL DETAIL**

   The system should include some means to spell a word letter by letter to achieve detail when important, for example, when using proper names.

5. **SPEAKER-LABELLING**

   The system should make some visual distinction between different speakers.

6. **NON-SPECIALIST EQUIPMENT**

   The system should be implemented on an easily obtainable computer system, for example an IBM PC or compatible, and produce its visual output on a standard monitor; although additional viewing facilities, such as a large television screen, or an overhead projector are desirable.

7. **HARD-COPY**

   It should be possible to produce a printout of a transcription. This would be of great benefit to deaf people, who often find note-taking impossible whilst concentrating on a speaker, even more so with the addition of a visual display to watch.

8. **COST**

   The system should be reasonably priced, under £2000, and hopefully around £300 in the future.

Two further problems are more difficult to overcome. A speaker often does not want to see their exact words transcribed, but rather what they *meant* to say, with all pauses, mumbles, stutters, repetitions and examples of bad English removed! Communication is more than the written word; it is very difficult to convey expression and feeling accurately. An automatic speech recognition system would replace face-to-face communication by person-to-machine-to-person communication. In the particular context of a university undergraduate lecture, an automatic speech recognition system would have to transcribe the speech of a single lecturer over a sixty minute period.

The emphasis is clearly on developing usable speech recognition systems that offer some help to deaf people in certain situations.

The domain of this research is university lectures. Experiences in America, where real-time classroom captioning is routinely provided at some institutions, have shown that hearing-impaired students can benefit a great deal from the printed display of speech. Hearing-impaired students at the Rochester Institute of Technology are benefiting from the use of "RapidText", a stenotype-based computer aided transcription system (see section 4.3.1). According to Victor Galloway, director of the National Center on Deafness at California State Northridge

> *This changes the way deaf people will receive information. It helps students in a classroom who are able to lip-read but who may be seated too far away.*
>
> [Mackey, 1989]

Classroom captioning, known as real-time graphic display (RTGD), is routinely provided in some courses. Students reported a higher mean level of understanding of lectures through reading the lectures in real-time on the television screen (RTGD) than from watching the interpreter. Asked which support service they would choose if only one were available, the students responded with the following [Miller, 1990]:

Display on TV	32%
Printout	30%

Interpreter 21%

Note-taking 16%

Tutoring 1%

It is clear then that deaf students at university would benefit from the visual display of speech during a lecture.

> *It is evident that the real-time printed display of speech, together with the printout that also becomes available, have considerable potential for many deaf students and particularly those in mainstreamed programs.*
>
> [Miller, 1990]

As will be seen in the following chapter, human-operated machine transcription systems achieve exceptionally low word error rates in real-time. The disadvantage of such systems is that they are very expensive to use, because of the difficulty in training operators. This high cost makes them impractical for everyday use in higher education, but does allow their use if the teaching of hearing impaired students is centralised (as at Rochester).

Existing large vocabulary automatic speech recognition systems also achieve low word error rates, the disadvantages being that they work only for read speech in a very restricted domain. Moving a particular system to a new domain requires a huge amount of training data relevant to the new domain, both for training acoustic models and language models. This is clearly impractical for use as an aid to deaf students in university lectures.

What is required, therefore, is a domain independent real-time automatic speech recognition system that performs to an acceptable level of recognition.

Chapter 3

Trends in Automatic Speech Recognition

This chapter contains an overview of each of the main stages of a typical automatic speech recognition system and a description of the current trends in speech recognition research. Many of these trends are independent and can be described as the *dimensions* of speech recognition, and form a taxonomy for describing automatic speech recognition systems. These dimensions almost completely form the design space for automatic speech recognition systems, but other factors do play a part. The dimensions that will be examined are: the speaker; the connectedness of speech; the speaking style; the units of speech; the language; the level of recognition; the vocabulary and the speed of recognition. Other trends that will be examined are: prosodic factors; speech corpora; performance measures and the integration of speech recognition and natural language processing techniques. The chapter will conclude with a look at some possible future trends in automatic speech recognition research.

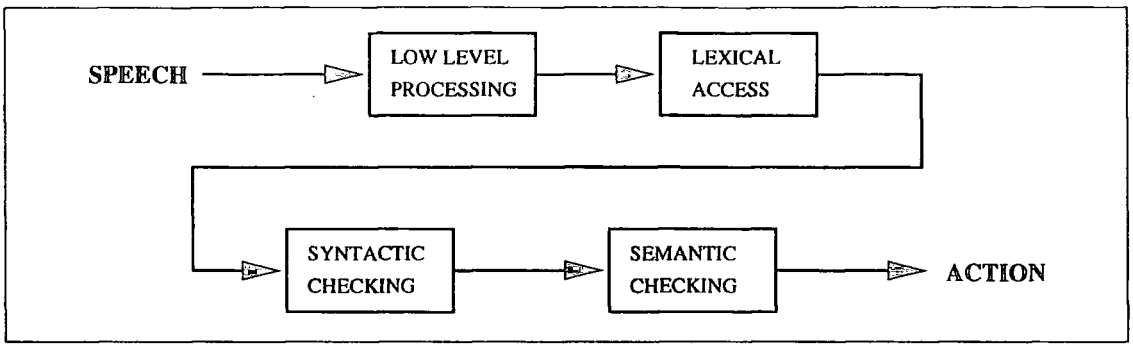


Figure 3.1: Typical Stages in an Automatic Speech Recognition System

3.1 An Overview of Automatic Speech Recognition Research

The stages of a typical automatic speech recognition system are shown in Figure 3.1. Each of these stages is described below.

3.1.1 Low Level Processing

The speech signal must first be filtered, to isolate those frequencies in the range of human hearing, then converted from an analogue to a digital form. It is very important that the information provided in the speech signal is extracted accurately, because errors made at this early stage of processing can easily propagate to other areas of the recognition process. Too much emphasis on high level techniques and poor quality low level (segmenting and then labelling) processing has been blamed for the weak performance of several of the speech recognition systems developed during the mid 1970s as part of the Advanced Research Projects Agency (ARPA) sponsored program of research and development. A good overview of the ARPA speech understanding project may be found in [Klatt, 1977].

The first step, speech analysis, is common to all approaches to automatic speech recognition. The speech analysis stage provides a spectral representation of the

characteristics of the speech signal in the form of a compact set of parameters. The two most common methods of speech analysis are filter bank analysis and linear predictive coding. During filter bank analysis, the speech signal is passed through a bank of several bandpass filters whose coverage spans the frequency range of interest (100–3000Hz for telephone quality speech, 100-8000Hz for broader signals). The individual filters overlap in frequency. Each filter processes the speech signal independently to produce a spectral representation at a particular frequency. Alternatively, during linear predictive coding (LPC), the speech signal is broken into a series of discrete frames. LPC spectral analysis produces a vector of LPC parameters that represent the signal spectrum over the time of the frame of speech. The parameters may then be converted to various other formats depending on the exact information that is required. One important set of parameters that can be derived are known as cepstral coefficients.

The second step, known as feature detection, converts the spectral measurements into a set of features that describe the broad acoustic properties of the different phonetic units. These features would include such things as nasality, frication, formant locations, voiced/unvoiced classification, energy ratios and pitch. A set of feature detectors would work in parallel and make a decision as the presence, absence or value of a particular feature.

The aim of the third step, segmentation and labelling, is to identify stable regions where features change very little over time. These segmented regions are then labelled according to how well the features match those of individual phonetic units. Typically, the result of segmentation and labelling is a sequence of the most likely phonemes. Some systems go further and pass a phoneme lattice on to the lexical access stage. A phoneme lattice is a two-dimensional structure giving the n most likely phonemes at each point in time. Each phoneme in the lattice would have a score associated with it according to to the quality of the matching features within a segment.

3.1.2 Lexical Access

The lexical access stage typically results in a word lattice structure representing the most likely sequences of words given the phoneme output from the previous stage. The speech recogniser would have associated with it a lexicon or vocabulary containing all the words known to the system and their phonetic representation. Other information in the lexicon might include the syntactic part(s) of speech for each word, for example noun, verb, adjective and the like. Each word in the lattice is scored against the phoneme sequences and the best matching ones are recorded in a word lattice.

3.1.3 Syntactic Checking

The role of the syntactic checker is to check the syntax of all of the possible paths through the word lattice. This stage makes use of the parts of speech information contained within the lexicon and any special grammar associated with the task that the speech recogniser is being used for. The lexical access stage and the syntactic checking stage are often combined into a single deterministic stage by only comparing syntactically allowable words against the phoneme sequence, this has the effect of substantially reducing the matching and checking space.

3.1.4 Semantic Checking

Once the search space has been reduced by the syntactic checking component, the semantic checking stage assesses the semantic correctness of the remaining sentence hypotheses. This may also make use of pragmatic knowledge by taking into consideration the particular context of the task the speech recogniser is being used for.

3.1.5 Action

The resulting action of the speech recogniser may be to simply display the best recognised sequence of words onto a visual display unit, or it may be to pass a query onto a database, or it may be to return a suitable response to the speaker and undertake a dialogue, in which case there will be several more components in the system.

3.2 Dimensions of Automatic Speech Recognition

This section introduces the major dimensions along which automatic speech recognition systems vary. Taken collectively, the dimensions form a taxonomy for describing automatic speech recognition systems.

3.2.1 The Speaker: Dependent vs. Independent

Human speech varies not only between speakers, but also within an individual speaker; words can vary in loudness, pitch, stress and pronunciation rate, even different words may sound similar. Automatic speech recognition systems are either speaker dependent (they work best with one particular speaker) or speaker independent (they achieve an acceptable recognition rate with anyone). Speaker dependent systems invariably demand a large amount of training data (in other words, samples of speech) from a speaker. Most training takes the form of repeatedly speaking sentences or words known to the system. The duration of the training is generally proportional to the size of the lexicon, though more recent systems are trained on a set of phonetically "rich and balanced" sentences that are independent of the task domain.

Segmenting the speech signal into word units has several consequences when it

comes to training a system. Each word has to be trained individually and repeatedly. If the lexicon contains a large number of words, this is a great inconvenience for a speaker. Extending the vocabulary would also be difficult. A speaker dependent system is said to be robust if recognition rates for new speakers (who haven't trained the system) are not too poor.

Speaker independent systems are trained on speech collected from a variety of sources. Speaker dependent systems will generally achieve a higher rate of recognition than speaker independent systems, though they are clearly not as versatile. Speaker adaptive systems are trained on speaker independent data, yet require each new speaker to repeat a small number of training samples; providing, in effect, an easier to prepare speaker dependent system.

The success of speaker independent (and speaker adaptive) systems depends on the availability and quality of sampled speech in order to build up an imaginary picture of the "average" speaker. The availability and variety of speech corpora is discussed in section 3.4.

3.2.2 The Connectedness of Speech: Isolated vs. Continuous

One of the biggest problems faced by an automatic speech recogniser is detecting the gaps between the words in a passage of speech. Early systems avoided this by only accepting individual words; these are known as isolated word recognisers. More recent research has concentrated on the problems of continuous speech, spoken at the normal speed of the human speaker. Three of the special problems caused by continuous speech are that:

- word boundaries are not clearly marked;
- words are "shrunk" (reduced) in order to achieve a faster speaking rate. For example, the word "solicitor" is often actually pronounced as "slisster". Short

words such as "of", "the", "in", "a" and the like almost disappear;

- words become assimilated, for example, "did you" is often actually pronounced as "diju".

Speech recognition systems have to overcome the problems of coarticulation when recognising continuous speech. Coarticulation occurs when one spoken sound is affected by either the previous or the following spoken sound. In other words, the context of the spoken sound needs to be taken into account. This happens not only between words, but also within words at the phoneme level. Word pronunciations are different when the words are uttered in isolation from when they are uttered in continuous speech [Giachin *et al.*, 1990]. Many of the algorithms based on pattern matching that were successful for isolated word systems cannot cope with the variations caused by continuous speech.

3.2.3 The Speaking Style: Read vs. Spontaneous

The speaking style used for communicating with a speech recognition system is of vital importance. There is a vast difference between read speech and spontaneous speech, that of disfluency. Disfluencies are irregularities of speech such as filled pauses, repair (including repeated words) and lost sentence structure. Filled pauses are strange sounds (for example "erm" and "err") used to fill silence while a speaker is thinking. Repair takes place when a self-correction occurs in speech and may or may not include cue words, part words and filled pauses, for example:

I want a vanilla no I mean a strawberry ice cream please.
I am so thir hungry.
I think I will have some vege err no some err cheese pie please.

During spontaneous speech, sentence structure often breaks down completely as a speaker tends to ramble on adding more and more information without completing the sentence that they originally started. This is made worse by filled pauses and

repair. Recent research has been undertaken into automatic analysis and correction of repair [O'Shaughnessy, 1992], and also in labelling speech repair [Bear *et al.*, 1993].

3.2.4 The Units of Speech: Whole-Word vs. Sub-Word

After segmentation, each segment must be labelled as some unit. Words would seem a natural choice as a unit of speech: they are the typical outcome of any recognition. Word models can also take within-word pronunciation differences into account. If the unit chosen is the word, then the recognition process simply relies on pattern matching against stored word templates. Although time is saved during labelling (no complex sub-word identification algorithms are required), scanning for templates to find the best match in a large lexicon (allowable words) can be very time consuming. Substantially more training data would be required to train the word models, compared to sub-word models; and extending the system vocabulary would require further training data.

Syllables are not a suitable unit: syllable boundaries in words are difficult to detect and there are so many possible syllables. Diphones are vowel-consonant sequences. They contain a lot of acoustic information as the diphone is taken across two sounds, so it contains much of the coarticulation information not present in other units, yet their main disadvantages are their large number, and the difficulties in representing words by sequences of diphones.

Phonemes are used by phoneticians as a convenient unit to represent speech sounds. The letter 'i', for example, is pronounced differently in the word "give" than in the word "hive", this would be reflected in their phonetic transcriptions:

```
give    :    /g I v/
hive    :    /h aI v/
```

Table 3.1 shows three different machine readable phonetic alphabets: ARPA-

bet, an American standard; the representation used in the Oxford Advanced Learner's Dictionary (see section 3.4.11; and SAM-PA, an European standard [Barry *et al.*, 1989]. The relative frequencies of each phoneme are calculated from a recent British English pronunciation dictionary (BEEP), containing 96,279 pronunciations.

Although the set of phonemes use in each language may be different, in practice, most are very similar because all humans share a similar speech apparatus. There are 44 phonemes in the English language. Each phoneme can be represented by several allophones. These classify speech sounds in terms of the way they are produced. Again, there would be many thousands of allophones for any given language. Acoustically defined sub-word units have also been used in speech recognition systems [Blomberg, 1989] [Svendsen *et al.*, 1989]. These units need not have a one-to-one correspondence with existing linguistic units. Segmentation of the speech signal in terms of these units can easily be done using well defined acoustic criteria. The difficulty then lies in generating a word lexicon based upon these acoustically defined sub-words.

One of the most successful methods of phoneme modelling, and the foundation of most recent automatic speech recognition systems, is hidden Markov modelling. A tutorial on hidden Markov models may be found in [Rabiner, 1989]. Hidden Markov models may be used at the segmentation and labelling level (as in the triphone model, for example) or at the syntactic level (known as the language model, or grammar).

Using Bayes' rule,

$$P(Model|Observed\ Features) = \frac{P(Observed\ Features|Model) \cdot P(Model)}{P(Observed\ Features)}$$

but this simplifies to

$$P(Model|Observed\ Features) \propto P(Observed\ Features|Model) \cdot P(Model)$$

ARPA-bet	*OALD*	*SAM-PA*	*Example*	*RelativeFrequency*
p	p	p	put	3.1%
b	b	b	but	2.3%
t	t	t	ten	6.8%
d	d	d	den	4.1%
k	k	k	can	4.7%
m	m	m	man	3.1%
n	n	n	not	6.5%
l	l	l	like	5.5%
r	r	r	run	5.4%
f	f	f	full	1.8%
v	v	v	very	1.2%
s	s	s	some	6.6%
z	z	z	zeal	3.6%
hh	h	h	hat	0.8%
w	w	w	went	0.9%
g	g	g	game	1.3%
ch	tS	tS	chain	0.5%
jh	dZ	dZ	Jane	0.8%
ng	N	N	long	1.6%
th	T	T	thin	0.3%
dh	D	D	then	12.2%
sh	S	S	ship	1.2%
zh	Z	Z	measure	0.1%
y	j	j	yes	0.8%
iy	i	i	bean	1.4%
aa	A	A	barn	0.9%
ao	O	O	born	1.0%
uw	u	u	boon	1.0%
er	3	3	burn	0.7%
ih	I	I	pit	10.0%
eh	e	E	pet	2.4%
ae	&	{	pat	2.5%
ah	V	V	putt	1.5%
oh	0	Q	pot	1.6%
uh	U	U	good	0.4%
ax	@	@	about	7.2%
ey	eI	eI	bay	2.0%
ay	aI	aI	buy	1.6%
oy	oI	oI	boy	0.2%
ow	@U	@U	no	1.5%
aw	aU	aU	now	0.4%
ia	I@	I@	peer	0.7%
ea	e@	e@	pair	0.2%
ua	U@	U@	poor	0.2%

Table 3.1: Three Machine Readable Phonetic Alphabets

as $P(Observed\ Features)$ is constant and independent of any model.

$P(Observed\ Features|Model)$ is known as the acoustic model. Each model is composed of a series of states and arcs between states. Each arc has an associated transition probability; in other words each state t depends on the previous state t - *1*. One model would be used for every word in the lexicon; or in the case of sub-word modelling, for every (context dependent or context independent) phoneme model. Word models can be made from concatenating phoneme models (see Figure 3.2).

A speech recogniser that uses phonemes as the sub-word unit must also take into account silence. For example, "sweet sheep" might be transcribed as

```
/sil s w i t sil S i p sil/
```

but taking left and right context into account, to produce triphones, this would be transcribed as

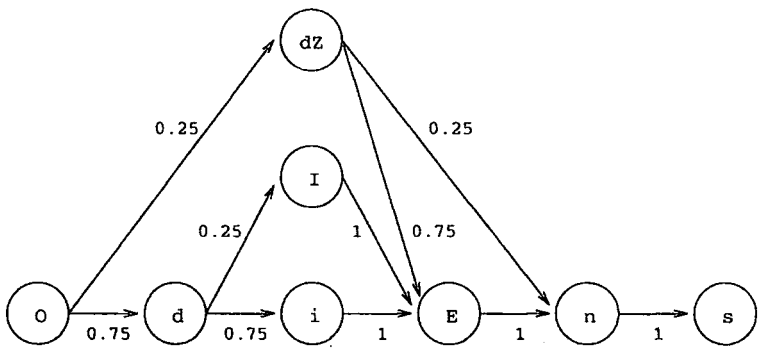
```
/()sil(s)  (sil)s(w)  (s)w(i)  (w)i(t)  (i)t(sil) ...
    ...  (t)sil(S)  (sil)S(i)  (S)i(p)  (i)p(sil)  (p)sil()/
```

The unit `(w)i(t)` is distinct from the unit `(S)i(p)` even though they have the same "main" phoneme. Therefore, instead of 45 context independent phoneme models (including silence) being used during labelling of the speech signal, 45^3 (91125) context dependent triphones would be required. Context dependent phoneme modelling performs better than context independent phoneme modelling because, for example, they model the coarticulatory effects which different contexts have upon the realisation of phonemes [Schwartz *et al.*, 1985]. The models have their associated probabilities calculated from training samples of speech. The more training information that is available, the more accurate will be the models. Sub-word based hidden Markov models can be trained on vocabulary independent and task domain independent samples. Vocabulary independent systems do not require new

Suppose that a speech recognition system has a lexicon containing one word: "audience". Four pronunciations of "audience" could be written (using context independent phonemes) as:

```
/O d i E n s/
/O d I E n s/
/O dZ E n s/
/O dZ n s/
```

This could be represented as a Markov model, each state representing one phoneme, and each arc having a probability associated with it.



Traversing the arcs can only result in the four pronunciations above; multiplying the probabilities along the way gives a resultant probability for each pronunciation occurring.

```
/O d i E n s/     0.75 * 0.75 * 1 * 1 * 1 = 0.5625
/O d I E n s/     0.75 * 0.25 * 1 * 1 * 1 = 0.1875
/O dZ E n s/      0.25 * 0.75 * 1 * 1     = 0.1875
/O dZ n s/        0.25 * 0.25 * 1         = 0.0625
```

Each phoneme (state, in the above model) would be associated with a model of its own in which each state would consist of a vector of numbers representing features of a speech signal. In a more complicated example with many more words, the probabilities associated with each arc would be considerably smaller.

Figure 3.2: An Example of a Word Markov Model

speaker enrolment or training when the recognition task is changed [Hon and Lee, 1991].

Although word based, context independent phoneme based and context dependent phoneme based, speech recognition systems are capable of vocabulary independent recognition, their performance is dependent upon the quality of the available training data. In order to ensure that all the models in a system are fully trained, it is necessary that each occurs frequently and in different contexts in the training data. In practice, many of the models never occur, and for those that do occur it may be difficult to observe enough samples in the training data, even for vocabulary dependent systems, to produce accurate estimates of those speech features present. Much recent research has concentrated on reducing the number of models to a more manageable figure [Rabiner *et al.*, 1989] [Sagayama, 1989]. The Sphinx speech recognition system [Lee, 1988] uses a clustering procedure to combine similar triphone contexts into clusters of generalised triphones. For example, in the ARPA Resource Management database, there are 2381 intra-word triphones, this rises to 7057 triphones when inter-word triphones are also considered. In Sphinx, this figure is reduced from 7057 models to 1,000 generalised triphone models. For vocabulary independent recognition, decision tree techniques have been used to produce a general set of context sensitive models [Hon and Lee, 1990] [Downey and Russell, 1992]. Work has also been undertaken that exploits some important features of speech which are apparent at the sub-phonemic level, by using sub-phonemic units originally developed for speech synthesis [Downey, 1993]. Initial results show that this approach is at least as good as the triphone approach.

The three fundamental units, whole word, phoneme-like and acoustic segment units, have been compared in [Lee *et al.*, 1989a]. The conclusions that were reached are that a hybrid approach based on a combination of both whole word and sub-word models should be used. Whole word models should be used for short function words that are acoustically more variable; phoneme like models are useful for constructing word models not observed during training; and acoustic models maintain

consistency during unit modelling and generation of acoustic representations of words in the lexicon.

Other, knowledge based, approaches have concentrated on simulating the expertise of spectrogram readers [Zue, 1985] [Guzy and Edmonds, 1986] [Hatazaki *et al.*, 1989] [Lamel, 1993]. A speech spectrogram is a graph showing the frequency of the speech signal against time. Trained readers can achieve up to 90% correct phonetic decoding when tested against phonetically labelled sentences. This approach encounters difficulties in translating the visual cues used by the expert readers into rules that can be applied on a numeric representation of the speech signal.

3.2.5 The Language: Restricted vs. Unrestricted

Perplexity is an important measure in specifying the degree of sophistication in a recognition task, it is often called the average word branching factor of the language model [Rabiner and Juang, 1993]. Perplexity is roughly calculated as the average number of allowable words at any given point in the recognition process [Sondhi and Levinson, 1978]. So, clearly, if no language model were to be used, in other words each word in the lexicon is equally likely to occur, then the perplexity would be equal to the number of entries in the lexicon. Conversely, a system that could only recognise the sentence "All sheep are sweet" will have a language model of perplexity one, as only one word is allowable at any given point during the recognition. Perplexity does not take into account any acoustic similarities between words in the lexicon. The Hearsay-II system [Engelmore and Morgan, 1988, Part I] developed at Carnegie Mellon University, for example, used (in common with several other systems) a semantic template grammar to restrict the solution set of possible utterances. A semantic template is a set of semantically type-equivalent phrases. These are stored as the nodes of a network, and any path through the network forms an acceptable sentence. One path might be "tell X about Y" where X and Y are semantic type templates. X might represent the set {me, us, him, her} and Y the set {ships, planes, submarines, helicopters}.

If a recogniser would only accept spoken input in this form, the perplexity of the system would be (1 + 4 + 1 + 4)/4 = 2.5. In this application the semantic template can be used to hypothesise the form of the sentence in a top down manner and other stages in the recognition process, such as acoustic/phonetic, can be used to choose among the alternatives. This is clearly too restrictive.

Most existing speech recognition systems use large amounts of data to train statistical (Markov model) language models. This data can be used to determine probabilities of word sequence likelihoods, known as n-grams, where n is typically two (bigram) or three (trigram). For example, in business correspondence the most likely word to occur after the word "Dear" would be "Sir". Using Markov models for the language model has several disadvantages: each word occurring in the lexicon must also occur frequently in the training samples in order for the language model to be at all complete and accurate; the training samples need to be relevant (in other words, realistically close) to the actual speech likely to be encountered — for example, it would be no use training the language model of an airborne reconnaissance reporting speech recognition system on training samples of people speaking poetry.

3.2.6 The Level of Recognition: Verbatim vs. Meaning

All automatic speech recognition systems can be categorised by the amount of speech understanding that takes place during recognition [Linggard, 1988]. At one end of the scale are the speech transcription systems that attempt to reproduce verbatim (in other words a transcript) what is spoken; at the other end are the speech understanding systems that respond to what is spoken either in the form of an answer or an action. It therefore follows that speech understanding systems can relax the requirement for 100% accurate recognition of speech, if the same meaning can be conveyed with, say, 80%–90% accuracy. Speech understanding involves the integration of speech recognition techniques with the techniques of natural language understanding. The speech recognition component of a system would hypothesise a

set of words or sentences, and the semantic component, incorporating knowledge of natural language understanding, would choose the most likely hypothesis [Makhoul, 1989]. Section 3.6 discusses this in more detail. Methods of evaluating automatic speech recognition systems are described in section 3.5.

3.2.7 The Vocabulary: Small vs. Large

The lexicon, or dictionary, contains the vocabulary of the speech recognition system. Each word contained in the lexicon is represented either as an averaged template estimated from several (different) spoken repetitions for pattern matching systems; or as a sequence of phonemes (or other unit) for sub-word based systems. There may be more than one phonetic representation to account for different pronunciations according to context. The lexicon may also include some syntactic information concerning the type of word, for example, noun.

The size and content of the vocabulary can play a large part in the success of every speech recognition system. For small vocabularies of 100–200 words, it may be possible to reduce potential confusion by deliberately not including similar sounding words. As the size of the vocabulary grows, the possibility of confusion between words grows, increasing the complexity of the task. Medium sized vocabularies have from 300–1,000 words, large vocabularies have between 1,000–5,000 words, and very large vocabularies have over 5,000 words.

3.2.8 The Speed of Recognition: Off-Line vs. On-Line

For some systems speed of recognition is not essential. It is possible to imagine, for example, that a manager dictating a letter into an office speech recognition system might not require the letter until later in the day. On the other hand, a pilot controlling part of an aircraft flight system by voice would need immediate recognition to avoid a potential accident.

3.3 Prosodic Factors

Most automatic speech recognition systems do not take prosodic factors into account even though prosody is critical to human speech perception. Prosodic factors that need to be considered include pitch, intensity, rhythm and duration.

[Waibel, 1988] puts forward the view that prosodic knowledge can contribute at various stages during the speech recognition process. At the lexical level, prosodic knowledge can provide an alternative way of hypothesising words to matching a series of phonemes or matching templates. For example, distinguishing between the words "did" and "deed" is quite difficult as the centre phonemes in each word are very similar (/I/ and /i/), however, the distinction may be made if the *duration* of the centre phoneme is taken into consideration. In the area of repair, work has been undertaken to identify false starts, by using word duration and fundamental frequency [O'Shaughnessy, 1992].

Prosodic knowledge is important and can be used in addition to semantic knowledge to recognise a sentence by understanding its meaning. Compton writes that the

> *... rise and fall of the voice in speech is another important factor in conveying meaning. This again is one of the speaker's unconscious devices for making the sense clear to the listener with the least effort on either's part.*
>
> [Compton, 1947, page 145]

I don't know where he is	(statement of fact)
I don't know where he is	(someone else may)
I don't know where he is	(contradiction)
I don't know where he is	(but I can guess)
I don't know where he is	(he has quite disappeared)
I don't know where he is	(I know where the others are)
I don't know where he is	(I know where he was)
I don't know where he is?	(why, of course I do!)

Figure 3.3: An Example of the Importance of Stress in Speech Comprehension

An example of what he means is given in Figure 3.3; this is taken from [Compton, 1947, page 146].

Some prosodic factors, such as energy (intensity), duration and fundamental frequency (pitch) are measured at the acoustic-phonetic level of processing.

3.4 Speech Corpora

The recent availability of high quality recorded speech corpora has seen a plethora of domain specific, very restricted grammar, high accuracy speech recognition systems. Most of the speech corpora available were initiated by ARPA, and the increasing complexity of the domain they represent, and of the speaking style, is an indication of the increasing performance of speech recognisers. This section will also look at several natural language corpora and dictionaries that have some relevance for speech recognition systems. Further information may be found in [Taylor *et al.*, 1991] and [Souter and Atwell, 1994].

The role of speech corpora has been to provide specific tasks for researchers to assess their speech recognisers. The speech corpora contain a large amount of high quality recorded speech that can be used for training and testing speech recognisers. Sub-word models *and* language models may be trained on a specific portion of the data available. The remaining data within a corpus may be used

for evaluation purposes. The use of recorded data eliminates the variability that may be introduced into the evaluation process: a text-based parser will repeatedly produce the same output for a given sequence of words, but a speech recogniser, on the other hand, is likely to be very sensitive to the smallest of variations in speaking style, background noise and the like, and is unlikely to repeatedly produce the same output for a given input. The use of recorded data also allows many different speech recognisers to be evaluated on the same test data, and the results compared against each other.

One disadvantage of training a speech recogniser on high quality recorded speech, from a speech corpus for example, is that the speech is recorded in an artificial (laboratory) environment. The consequence of this is that the performance of the recogniser in a real environment deteriorates substantially [Spitz, 1991]. The majority of the difference in performance can be accounted for by the acoustic models for silence, in other words the quiet laboratory environment used for collecting the speech data is far from the noisy office environment into which the recogniser is to be released.

3.4.1 TIMIT

The TIMIT corpus was developed to train and evaluate speaker independent phoneme recognition systems [Lamel *et al.*, 1986]. It consists of 630 speakers (441 male), each saying 10 sentences.

3.4.2 RM

The ARPA Resource Management (RM) corpus for continuous speech recognition [Price *et al.*, 1988] was developed under the Strategic Computing Speech Recognition Programme. The corpus represents over 21,000 recorded utterances from 160 speakers with a variety of dialects, and is separated for training and testing purposes. The utterances consist of read speech and are made up of database

queries, for example:

> how many long tons is the average displacement of ships in bering strait
>
> list the cruisers in persian sea that have casualty reports earlier than jarrett's oldest one

The Resource Management corpus uses a vocabulary of 1,000 words and can be used with three different language models, with perplexities 9, 60 and 1,000 (no grammar).

3.4.3 ATIS

The Air Travel Information System (ATIS) corpus serves as one of the common tasks for ARPA spoken language system research and development (see section 4.1.1. The corpus was collected by six different organisations in the United States (Texas Instruments, AT&T, BBN, Carnegie Mellon University, MIT and SRI), and includes over 14,000 utterances from over 430 speakers [Hirschman *et al.*, 1993]. Three types of system tests may be performed: spontaneous speech recognition tests, natural language understanding tests and spoken language understanding tests. Like the RM corpus, the ATIS corpus consists of database queries, for example:

> please list the flights from pittsburgh to baltimore that will be made by six seat airplane on june twentieth
>
> list the number of first class flights available on delta airlines

The perplexity of the language models used in the ATIS corpus ranges from 17 to 35, depending on the query classification.

3.4.4 WSJ

The Wall Street Journal (WSJ) continuous speech recognition (CSR) corpus will improve upon the ATIS domain by providing ARPA with its first very large vocabulary, high perplexity, general purpose natural English corpus. The corpus is available in speech and text forms and will contain 400 hours of speech data and forty seven million words of text data and allows the integration of speech recognition and natural language processing in a highly practical application domain [Phillips *et al.*, 1992]. This corpus is currently under development at several sites in the United States.

The WSJ corpus can be used with variable size large vocabularies (5,000, 20,000 words and larger), variable perplexities (80, 120, 160, 240 and larger), speaker dependent and speaker independent training with variable amounts of data, including equal portions of verbalised and non-verbalised punctuation (to allow dictation and non-dictation applications), variable microphones, variable noise levels and equal numbers of male and female speakers [Paul and Baker, 1992]. The majority of the recorded speech is read speech which is prompted by newspaper text paragraphs, though a small amount of the utterances consist of spontaneous dictation [Bernstein and Danielson, 1992].

3.4.5 SCRIBE

The SCRIBE corpus is the British English speech database. It consists of a variety of phonetically 'compact' and 'rich' sentences, a two minute accent sensitive passage, ten 'free' speech sentences, fifty 'natural' task-specific sentences and fifty 'synthetic' task-specific sentences. Seventy one talkers with four regional accents were used, recording a total of over 10,000 sentences.

3.4.6 Durham

The Durham Lecture corpus contains texts that are prepared but unscripted single speaker oration. The main bulk of the texts are undergraduate lectures recorded and transcribed at the University of Durham, a BBC television lecture is also included. The lectures were on a number of topics, performed by male and female speakers whose age, experience in speaking to an audience, and background all varied, although each has an academic background. The transcriptions were made as accurate as possible by including part words, sentence repair and filled pauses. This corpus is currently under development.

3.4.7 Brown

The Brown corpus was compiled in the early 1960's at Brown University in the United States. It contains 500 text samples of some 2,000 words each, representing fifteen categories of American English texts printed in 1961 [Francis and Kucera, 1979]. The corpus is available in a number of versions, with and without part of speech tagging.

3.4.8 LOB

The Lancaster-Oslo/Bergen (LOB) corpus was compiled in the 1970's at the Universities of Lancaster (England) and Oslo (Norway). It is a British English counterpart of the Brown corpus and contains 500 text samples of some 2,000 words each, representing fifteen categories (identical to the Brown corpus) of British English texts printed in 1961 [Johansson *et al.*, 1978]. The corpus is available in a number of versions, with and without part of speech tagging.

3.4.9 LUND

The London-Lund (LUND) corpus contains 100 spoken English texts of some 5,000 words collected and transcribed at the Survey of English Usage, University College London, and computerised at the University of Lund (Sweden) [Svartvik, 1992]. The texts in the corpus are transcribed orthographically, with detailed prosodic marking. They represent a range of text categories, such as spontaneous conversation, spontaneous commentary, spontaneous and prepared oration, and the like. Five of the Lund texts are prepared but unscripted single speaker oration. It is not clear if any post-processing of the transcriptions has taken place; some of the transcriptions look too "clean" to be actual spontaneous speech. The LUND corpus does not contain any part of speech tagging.

3.4.10 SEC

The Lancaster/IBM Spoken English corpus (SEC) contains approximately 52,000 words, representing eleven categories of contemporary spoken British English [Taylor and Knowles, 1988]. The majority of the texts of the corpus were obtained from the BBC. The material is available in orthographic and prosodic transcription versions, and in two versions with part-of-speech tagging.

3.4.11 OALD

The machine-readable form of the Oxford Advanced Learner's Dictionary (OALD) contains over 70,000 entries and was derived originally from the Oxford Advanced Learner's Dictionary of Current English, third edition, published by the Oxford University Press in 1974. It contains all of the headwords and subentries, including their inflected forms, from the original dictionary, to which were added 2,500 proper names [Mitton, 1992]. Each entry includes an orthographic spelling, a phonetic spelling (pronunciation) with an indication of primary and secondary stress,

possible parts of speech with rarity indicators, the number of syllables, and, for verbs, the sentence structures in which the word can occur.

3.5 Performance Measures

Performance is generally measured in terms of accuracy of recognition, and speed of recognition. These figures, though, must be taken relative to the size and perplexity of the grammar, as well as the level of speaker dependency and the speaking rate [Pallett, 1985] [Hunt, 1988].

In assessing the accuracy of a speech recognition system, care must be taken to examine the kind of word errors: deletions, insertions or substitutions of words. Deletions occur when something was spoken, but nothing recognised; insertions occur when nothing was spoken, but something was recognised; and substitutions occur when a word is recognised in place of another.

Evaluation of automatic speech recognition systems is discussed in detail in chapter 7. Simple metrics include calculating the percentage of correct words and the percentage of substitution, insertion and deletion errors. The existence of speech corpora allows different systems to be evaluated on the same data.

3.6 The Integration of Speech Recognition and Natural Language Processing Techniques

Unfortunately, many of the techniques for parsing text-based (typed) natural language do not adapt well to speech specific problems [Young *et al.*, 1989].

- **Probability Measures**

  Typed input is accurate; whereas the result of each stage in an automatic speech recognition system involves some form of probabilistic estimation.

- **Identifying Words**

  Several words are hypothesised for each actual word.

- **Phonetic Ambiguity of Words**

  Many words sound identical, and can only be correctly identified when context is taken into account.

- **Syllable Omissions**

  Words are often missed out to achieve higher speaking rates, or successive words are assimilated.

- **Missing Information**

  The automatic speech recognition system may completely fail to recognise a correctly spoken word.

- **Ungrammatical Input**

  Whereas natural language understanding systems have to handle mis-typing, speech systems have to cope with mis-spoken words, inserted pauses and noises. Natural speech is also more likely to be ungrammatical.

Rather than use the power of full natural language processing (NLP) systems, speech recognition researchers have only made use of parsing and some semantics in order to achieve their aims. Much research has concentrated on the area of robust or partial parsing [Ward, 1991b] [Stallard and Bobrow, 1992] [Baggia and Rullent, 1993]. This has been combined with frame-based semantics for robust speech processing in the ATIS domain. For example, working in the ATIS domain, a speech recognition system would use partial parsing and semantic frames on the following sentence:

i want a flight uh that arrives in boston let's say at 3pm

to extract the information flight, arrive, Boston, 3pm, and ignore the irrelevant parts of the sentence. These techniques are suited to information retrieval applications but not for the more sophisticated speech understanding tasks.

Sentence	*Log Likelihood of Model*	
	Acoustic	*Language*
THE LOW WAS ELEVEN OH NINE POINT OH EIGHT	-24424.82	-794.52
THE LOW WAS ELEVEN OR NINE POINT OH EIGHT	-24494.42	-733.35
THE LOW WAS ELEVEN OWN NINE POINT OH EIGHT	-24441.68	-786.91
THE LOW WAS ELEVEN OWNED NINE POINT OH EIGHT	-24447.09	-792.71
THE LOW WAS A LITTLE KNOWN NINE POINT OH EIGHT	-24537.12	-706.06
THE LOW WAS ELEVEN O. NINE POINT OH EIGHT	-24424.82	-841.33
THE LAW WAS ELEVEN OH NINE POINT OH EIGHT	-24495.23	-775.93
THE LAW WAS ELEVEN OR NINE POINT OH EIGHT	-24564.83	-714.76
THE LAW WAS ELEVEN OWN NINE POINT OH EIGHT	-24512.09	-768.32
TO THE LOW WAS ELEVEN OH NINE POINT OH EIGHT	-24392.35	-890.53

Table 3.2: The N-Best Output of a Speech Recogniser on a WSJ Sentence

A common interface between speech recognition and natural language processing systems is the n-best sentence list. This represents the most likely sentences according to the speech recognition system, usually taking into account acoustic information and a trigram language model. The role of the NLP system is to then select the most likely sentence from the list, making use of a deeper grammatical analysis in addition to semantics and pragmatics, and perform some action. This is a suitable method for overcoming the "short-sightedness" of the trigram language model. An example of the n-best (n = 10) output of a speech recognition system[1] is shown in Table 3.2.

3.7 Future Trends in Automatic Speech Recognition Research

As the next chapter will show, existing automatic speech recognition systems have achieved very good recognition with very large vocabularies on restricted domain (Wall Street Journal for example) read speech. Substantial amounts of training data are used to train acoustic and language models. These systems cannot be

[1]Generated using The HTK Large Vocabulary Speech Recognition System developed by Steve Young, Phil Woodland, Julian Odell and Valtcho Valtchev from the Speech Vision and Robotics Group, Cambridge University Engineering Department.

improved much more in this kind of scenario. The major challenges facing the speech recognition community in the future are in developing *domain independent* large vocabulary systems, initially for read speech. Subsequent efforts should be aimed at handling spontaneous speech.

More integration will take place with large-scale natural language processing systems, as serious applications beyond word recognition and into spontaneous speech understanding are demanded by users.

Chapter 4

Existing Systems

This chapter describes the most recent ARPA speech recognition evaluations (December 1993) [ARPA, 1994] and outlines the capabilities of existing systems for automatic recognition of continuous speech, and of existing systems used by the deaf community for real-time machine transcription of speech.

4.1 Recent ARPA Speech Recognition Evaluations

Over the years, ARPA (formerly known as DARPA) have organised many competitive evaluations of sites that they support financially, who research into speech and language. More recently, invitations have been extended to several European groups to evaluate their systems for comparison.

4.1.1 ATIS

The Air Travel Information System (ATIS) evaluation assessed speech recognition (SPREC), natural language database query (NL), and their integration, spoken language understanding (SLS). The ATIS corpus is described in section 3.4.3. The task was to solve air travel planning scenarios using a 46-city relational database of air travel planning information.

The ATIS corpus moves on from just evaluating speech recognition performance and reflects the recent advances made in the recognition of spontaneous speech and the importance of actually doing more than recognising words by evaluating any subsequent actions. When the ATIS task was developed in 1990 [Price, 1990], little work had been done on formal evaluation of understanding for natural language interfaces. In the absence of a generally accepted semantic representation, the ARPA spoken language system community focussed instead on generating "correct" database queries. Evaluation was then based upon a comparison between canonical database answers and system answers [Pallett, 1991] [Pallett *et al.*, 1992]. Correct answers are classified as being context independent (A), context dependent (D) and un-evaluable (X). Queries with un-evaluable answers are only used for SPREC evaluation.

The results of the 1993 ARPA evaluation are summarised in Table 4.1. The term %WE refers to the percentage word error made by a system, and the term %UE refers to the percentage utterance (in other words, sentence) error made by a system.

4.1.2 CSR (SPREC)

The ARPA continuous speech recognition evaluation was performed on parts of the Wall Street Journal Corpus (WSJ), described in section 3.4.4. The evaluation consisted of two 'hub' tests, and nine 'spoke' tests, these are shown in Table 4.2. Each test had a primary condition (P0), in which any acoustic data or language

System	*SPREC (%WE)*			*NL (%UE)*		*SLS (%UE)*	
	A	D	X	A	D	A	D
AT&T	8.6%	9.6%	15.5%	7.4%	14.2%	22.1%	28.0%
BBN	3.0%	4.0%	7.2%	9.6%	21.8%	13.8%	22.5%
CMU	3.0%	3.9%	7.3%	6.0%	13.8%	8.9%	19.1%
CRIM	6.3%	7.2%	15.0%	14.7%	29.2%	23.7%	34.5%
MIT-LCS	4.3%	4.9%	10.0%	10.0%	16.0%	11.8%	17.5%
SRI	3.9%	5.5%	8.0%	14.3%	32.3%	16.5%	36.3%
UNISYS	3.6%	4.9%	10.1%	28.6%	63.1%	33.5%	65.2%

Table 4.1: Summary of the ARPA 1993 ATIS Evaluation Results

model may be used, and several contrastive tests (CX), in which some conditions were fixed to allow comparison between systems. All sites were required to evaluate on one of the Hub tests, the spoke tests were optional. The H1-P0 condition was the premier test of the evaluation. The spoke tests advanced research in several directions: adaptation of the language model (S1 and S2); adaptation to the speaker (S3 and S4); compensation for channel variability (S5 and S6); compensation for noise (S7 and S8); and spontaneous dictation (S9). Recognition time was not measured in the ARPA CSR evaluation; indeed some systems took several hours to recognise each sentence.

Results for the H1-P0 (any grammar), H1-C1 (trigram), H2-P0 (any grammar) and H2-C1 (bigram) tests are summarised in Table 4.3. Several groups entered more than one system, differences between systems are described below under each individual group's details. Only one site (BBN) competed in the S9, spontaneous speech recognition, test. On S9 data, their S9 system achieved 19.1% word error and their H1-C1 system achieved 24.7% word error, indicating, as expected, that spontaneous speech is harder to recognise than read speech.

Test	*Description*	*Vocabulary*
H1	Read WSJ Baseline	64K
H2	Read WSJ Baseline	5K
S1	Language Model Adaptation	Unlimited
S2	Domain Independence	Unlimited
S3	SI Recognition Outliers	5K
S4	Incremental Speaker Adaptation	5K
S5	Microphone Independence	5K
S6	Known Alternate Microphone	5K
S7	Noisy Environments	5K
S8	Calibrated Noise Sources	5K
S9	Spontaneous WSJ Dictation	Unlimited

Table 4.2: ARPA 1993 CSR Evaluation Tests

4.2 Existing Systems for Automatic Recognition of Continuous Speech

This section first describes each of the systems that entered the ARPA ATIS/CSR evaluations, and then three other systems of note.

4.2.1 AT&T

The AT&T ATIS system [Bocchieri, 1994] used a natural language understanding system provided by CMU; the interface was the single best sentence provided by the recogniser. In the speech recognition component, 998 context independent phone models were used, and the acoustic model was trained on 14,000 ATIS sentences. The language model used a probabilistic finite state grammar; 18,000 ATIS sentences were used to train a bigram model, with a perplexity of 25. The size of the lexicon was 1433 words, one pronunciation per word. The search algorithm used a standard forward beam search. Recognition took approximately two minutes on an SGI R4000 computer.

System	*H1 - 64K*		*H2 - 5K*	
	P0	C1	P0	C1
BBN	12.2%	14.2%		
BU (1)		15.7%	6.7%	11.6%
BU (2)		14.3%	5.4%	10.3%
BU (3)		14.5%	5.8%	10.8%
CMU (1)		13.6%		
CMU (2)	13.9%			
CU-CON				13.5%
CU-HTK (1)		12.7%		
CU-HTK (2)			4.9%	8.7%
CU-HTK (3)				12.5%
DRAGON		19.0%		
ICSI				17.7%
LIMSI (1)		11.7%		
LIMSI (2)			5.2%	9.3%
MIT-LL	16.8%	18.6%		
PHILIPS (1)			9.2%	12.3%
PHILIPS (2)		14.8%	6.4%	
SRI		14.4%		

Table 4.3: Summary of the ARPA 1993 CSR Evaluation Results

4.2.2 BBN

The BBN Systems and Technology HARC (Hear And Respond to Continuous speech) spoken language system integrates a speech recognition sub-system, Byblos, and a natural language understanding sub-system, Delphi [Stallard, 1994] [Zavaliagkos *et al.*, 1994] [Bates *et al.*, 1993] [Bates *et al.*, 1992] [Kubala *et al.*, 1992] [Bobrow *et al.*, 1992]. Byblos uses a multi-pass search strategy designed to use progressively more detailed models on a correspondingly reduced search space. The output is an n-best list of hypotheses which is then re-ordered by several knowledge sources. The top choice in the list is used for results on Byblos alone; the entire list is passed to the language understanding component for further re-ordering and interpretation. For the ATIS evaluation, the acoustic model was trained on a large number of ATIS sentences; the language model was trained on 30,000 ATIS sentences; the lexicon contained 2600 words. For the CSR evaluation, the acoustic and language models are trained on WSJ data, and the n-best

output of Byblos was re-scored using a segmental neural network. On the CSR spontaneous WSJ dictation (spoke 9) test, BBN used their H1-P0 system with the addition of 1000 new words to the lexicon, and 8000 spontaneous WSJ dictation sentences for language model training.

The natural language component, Delphi, uses an agent-based chart parser, with scheduling based on the measured statistical likelihood of grammatical rules. The system allows for semantic interpretations of input which has no valid global syntactic analysis, by the use of a fallback component in which statistical estimates play an important role.

The basic interface between Byblos and Delphi in HARC is an n-best list. In evaluating HARC on the ATIS test set, n was set to five. Initially Delphi applies the full parsing strategy to each of the sentences in the list passed to it from Byblos. If none of these results are acceptable, Delphi makes a second pass through the hypothesis list using the fallback strategy.

4.2.3 BU

The Boston University ATIS system combined the BBN Byblos speech recogniser with the Boston Stochastic Segment Model (SSM) recogniser [Ostendorf *et al.*, 1994]. The interface between the two systems was an n-best sentence list (n = 100). In Table 4.3, the BU(1) system is the baseline version of Byblos shown for comparison, this is different to the BBN system described above which uses a segmental neural network (SNN) to re-score the n-best output of the Byblos recogniser. The BU(2) system re-scored the n-best output of Byblos using HMM log-likelihood; SSM log-likelihood; SNN log-likelihood; n-gram word sequence probability; language model scores; and phoneme, word and silence counts. The BU(3) system is similar to the BU(2) system but does not make use of the HMM and SNN log-likelihood scores. In each of the systems, the lexicon used is identical to that used by the BBN Byblos system under the corresponding conditions.

4.2.4 CMU

Research at Carnegie Mellon University is focussed around the Sphinx speech recognition system. Sphinx uses phonetic hidden Markov models which are trained on task dependent data [Lee *et al.*, 1989b] [Lee *et al.*, 1990]. Generalised triphones are used to model coarticulatory effects; similar triphones are merged to improve the trainability of the models and the probabilities smoothed to improve robustness. Recently, the performance of the Sphinx system was greatly improved, with the new system being called Sphinx-II [Huang *et al.*, 1993] [Alleva *et al.*, 1992]. These improvements have been made using additional dynamic features, speaker normalised features, semi-continuous hidden Markov models, sub-phonetic modelling, vocabulary independent and adaptive speech recognition, speaker adaption, efficient search and language modelling. A three pass search strategy was used: left-to-right beam search with bigram, right-to-left beam search with bigram, and an A* search with a trigram language model.

For the ATIS task, the Sphinx-II speech recognition system produced a single best hypothesis for the spoken input which is then passed to the Phoenix natural language understanding system which uses flexible parsing to cope with novel phrasings and mis-recognitions [Isaar and Ward, 1994] [Ward *et al.*, 1992] [Ward, 1991a]. The system used a 3207 word lexicon. The acoustic model was trained on 22,000 ATIS sentences, and the language model was trained on 26,000 ATIS sentences.

For the CSR task, the CMU systems used a lexicon of 19,979 words. The CMU(1) system used a trigram language model provided by MIT-LL. The CMU(2) system used an adaptive language model that combined the conventional trigram language model with mutual information models (bigram, trigram and a long distance bigram model), in addition to a "rare words only" unigram cache, and a bigram cache.

4.2.5 CRIM

The Centre de recherche informatique de Montreal (CRIM) ATIS system used a large vocabulary spontaneous speech recogniser to generate a list of sentence hypotheses, the best of which was passed to a natural language component for interpretation [Normandin *et al.*, 1994]. For each sentence, 100 hypotheses were produced for each of two acoustic models (male and female), these are then re-scored using cross-word triphone models, followed by bigram and trigram language models. The perplexity of the language models was 18 (bigram) and 9 (trigram). The final score for each sentence was obtained using a weighted sum of these three scores. N-best lists were then produced using a two-pass beam search and a bigram language model. The recognition dictionary contained 1863 entries. Only one sentence is sent to the natural language module, which makes use of a chart parser and semantic frame classification, for interpretation.

4.2.6 CUED (CU-CON)

Cambridge University Engineering Department's Connectionist group (CU-CON) used a hybrid connectionist-HMM speech recognition system for the ARPA CSR evaluation [Robinson *et al.*, 1994] [Hochberg *et al.*, 1994]. A recurrent net was used to map acoustic vectors to probabilities of phone classes. The maximum likelihood phone or word string is then extracted using Markov models. The acoustic training data consisted of 84 speakers uttering a total of 7200 sentences. The lexicon, provided by Dragon, contained the standard WSJ 5K words. The language model, provided by MIT-LL, was the standard bigram language model.

4.2.7 CUED (CU-HTK)

The second Cambridge University Engineering Department system (CU-HTK) was a large vocabulary continuous speech recogniser built using HTK, an HMM toolkit

[Woodland *et al.*, 1994a] [Woodland *et al.*, 1994b]. HTK has a unique generalised parameter sharing mechanism that allows HMM systems to be constructed that are balanced between acoustic model complexity and parameter estimation accuracy for a given training corpus. The CU-HTK(1) system used gender independent triphone models and was trained on 7193 WSJ utterances (14 hours of speech). Word recognition was performed using a static network decoder with a 5K bigram language model. The CU-HTK(2) system used gender dependent triphone models, and was trained on the same amount of data. Word recognition was performed using a dynamic network decoder and the same 5K bigram language model. The CU-HTK(3) system used gender dependent triphone models and was trained using 36,515 WSJ utterances (66 hours of speech). Word recognition was performed using a dynamic net decoder with 5K bigram and 20K trigram language models. Dynamic network decoding required approximately ten minutes per sentence using a 20K lexicon. CU-HTK systems gave the lowest word error rates in three out of the four ARPA tests entered, and the second lowest word error rate on the fourth test.

4.2.8 DRAGON

The Dragon large vocabulary speech recognition system was an HMM-based system [Scattone *et al.*, 1994] [Roth *et al.*, 1993] [Baker *et al.*, 1992]. It used context dependent, gender dependent, triphone models and was trained on 26,000 WSJ utterances. Gender determination was performed before recognition. Word recognition was performed using a single pass dynamic programming algorithm with the standard trigram language model, provided by MIT-LL.

4.2.9 ICSI

The International Computer Science Institute used a hybrid HMM and multi-layer perceptron system for the ARPA CSR evaluation [Morgan *et al.*, 1994]. This

system was a pilot system scaled up from ICSI's Resource Management system. It made use of context independent, gender independent, phone models, trained on 7200 WSJ utterances. The standard bigram language model and 5K pronunciation lexicon were used.

4.2.10 LIMSI

The LIMSI continuous speech dictation system was an HMM-based system that used context dependent, gender dependent, phone models [Gauvain *et al.*, 1994a] [Gauvain *et al.*, 1994b]. The acoustic model was trained using 37,518 WSJ sentences from 284 speakers. For word recognition, a two pass beam search was used: the first pass used the standard bigram language model to generate a word lattice, and the second pass used a trigram language model to search the word lattice. The LIMSI(1) system used a 20K pronunciation lexicon, and the LIMSI(2) system used a 5K pronunciation lexicon.

4.2.11 MIT (MIT-LCS)

The Massachusetts Institute of Technology Laboratory for Computer Science spoken language system couples the Summit speech recognition system with the Tina natural language understanding system [Zue *et al.*, 1992] [Seneff, 1992]. The system used a lexicon of 2460 words. There are three major components in the Summit system: acoustic-phonetic; pronunciation network; and linguistic decoder [Zue *et al.*, 1990]. The phonetic recognition subsystem of Summit takes as input the speech signal and produces as output a network of phonetic labels with scores indicating the system's confidence in the segments and in the accuracy of its labels. A pronunciation network is established for each entry in the system's vocabulary. This contains the possible different pronunciations for each word, determined by a phonological expansion system, and their associated likelihoods. The linguistic decoder produces an n-best list of candidate word sequences in decreasing order of

total path score. It makes use of an A* search algorithm during alignment of the phonetic network with the lexical word pronunciation network.

The Tina natural language system was developed for applications involving speech recognition tasks [Seneff, 1989]. The parser used a best first strategy, with probabilities obtained automatically from a set of example sentences. The grammar was entered as a set of simple context free rules which are automatically converted to a shared network structure. Tina parsed the n-best word sequence hypotheses provided by Summit, and, for the best parse, generated a set of query functions which were passed to the back end for response generation [Hirschman *et al.*, 1991].

4.2.12 MIT (MIT-LL)

The Massachusetts Institute of Technology Lincoln Laboratory large vocabulary continuous speech recognition system is a stack decoder-based HMM system [Paul, 1994] [Paul and Necioglu, 1993]. The system used gender dependent triphone models, and was trained on 37,000 WSJ utterances (82 hours of speech). A stack decoder is used to control the acoustic and language model search by applying a fast match routine to find a small number of potential words which are then evaluated using a more expensive detailed match [Paul, 1992]. The standard 20K trigram language model was used.

4.2.13 PHILIPS

The Philips large vocabulary continuous speech recognition system is an HMM-based system that uses gender independent triphone models [Aubert *et al.*, 1994]. PHILIPS(1) was trained on 7200 WSJ utterances from a total of 84 speakers, and used the 5K lexicon provided by LIMSI. PHILIPS(2) was trained on 37,200 WSJ utterances from a total of 284 speakers, and uses the 20K lexicon provided by Dragon. For word recognition, both systems formed a word lattice using a left-to-right beam search incorporating a bigram language model. The word lattice was

then re-scored by incorporating an additional trigram language model.

4.2.14 SRI

Decipher is SRI's hidden Markov model based speaker independent continuous speech recognition system [Murveit *et al.*, 1993a] [Digalakis *et al.*, 1994]. The system used gender dependent triphone models. For word recognition, a two pass progressive search strategy was used. The first pass generated a word lattice using a bigram language model, the second pass re-scored the lattice with more complex HMM models to generate an n-best list of sentence hypotheses. For the CSR C1 test, the standard WSJ trigram language model was used to re-score and re-order the n-best list.

A natural language processing system, known as SRI Travelogue, was integrated with Decipher for use with the ATIS corpus [Moore *et al.*, 1994] [Appelt and Jackson, 1992]. The acoustic component of Decipher was trained on 19,854 ATIS utterances. The lexicon consisted of 1665 words. The n-best output of Decipher was re-scored using a parser-based language model. The Travelogue system consists of a template matching sentence analysis mechanism together with a context handling mechanism and a database query generation component.

4.2.15 UNISYS

The Unisys ATIS system consisted of the BBN speech recognition system combined with a natural language processing system [Dahl *et al.*, 1994]. The interface between the two systems was an n-best list of sentence hypotheses. The NL component used robust parsing techniques to re-score the n-best list. The list was then re-ordered and the remaining part of the NL system (using semantics) used to filter out unacceptable hypotheses. This was achieved with varying success: results for SPREC showed a significant increase in sentence error; results for SLS were slightly improved.

4.2.16 DRA

The Armada system was produced as part of the ARM (Airborne Reconnaissance Mission) project being undertaken at the Speech Research Unit at the Defence Research Agency (DRA). Armada was a medium sized vocabulary, speaker dependent, continuous speech recognition system. With a (null) grammar of perplexity 540, Armada achieved a word correct rate of 94.3%, and a word accuracy of 82.0%. With a grammar of perplexity six, the word correct rate was 99.5% and the word accuracy was 99.2%. [Ponting and Russell, 1989] [Parry, 1990] [Russell *et al.*, 1990a]

The texts of the ARM reports were created using an automatic sentence generator based on a finite state syntax and a 497 word vocabulary. This syntax was based on existing airborne reconnaissance reports and had a perplexity of approximately six. Each report was recorded by two male and one female speakers in a sound proof room using a head mounted microphone. Recordings were sampled at 20kHz to produce 100 frames per second. Orthographic annotation was done semi-automatically and then checked manually; some non-speech sounds occurring between sentences were also labelled. A dictionary containing a single phonemic transcription of each word in the ARM vocabulary was created for each speaker. The system was first trained on one or two hand labelled ARM reports at the context independent phoneme level. These models were then optimised using 37 training reports. The context independent phoneme hidden Markov models were then used as initial estimates of the associated triphone model parameters. These were then optimised on the same 37 report set.

Armada was based on sub-word hidden Markov models in which the basic unit was the triphone. Three classes of hidden Markov models were used in the Armada system: triphone models (approximately 1500), in which each triphone was modelled using a three state hidden Markov model; word level models (6), in which short words, such as "air", "at" etc, were modelled explicitly rather then as a sequence of triphones; non-speech models (4), in which non-speech sounds, such as "silence", "short noise" and the like, were represented by single state hidden Markov models.

Two syntaxes have been used to assess the performance of Armada. In the word syntax, triphone sequences were constrained to be consistent in that the centre and right context phonemes of a particular phoneme had to be identical to the left context and centre phonemes of the following triphone, as well as producing a valid word sequence according to the Armada dictionary. The additional restrictions of the full syntax were that the word sequence must be consistent with the ARM syntax.

More recently a speaker independent recogniser has been developed for the ARM task [Russell, 1992b]. This system was trained on three recordings of complete ARM reports from each of 61 male speakers. The assessment of the final system was done on a test set consisting of three reports each from 80 male speakers, none of whom were in the training sets, giving a total of around 13,000 words. Without using any explicit syntactic constraints, the system achieved a word correct rate of 84.1%, and a word accuracy rate of 74.1%.

4.2.17 CSTR

The Centre for Speech Technology Research at Edinburgh University developed a real-time domain dependent, speaker dependent, speech recognition system known as Osprey [Clery, 1989]. It was based on readily available, off the shelf digital technology and plugged into an IBM-AT compatible personal computer. The vocabulary was limited to 300 words. A finite state transition grammar with a typical branching factor (perplexity) of 3 to 5 words was used. Osprey was divided into three layers: the technology platform, the algorithmic layer and the applications layer [Sutherland *et al.*, 1989].

The technology platform described the hardware basis and processing requirements for the system. It was required to be flexible, so that changes to the processing or recognition algorithms caused minimal disturbance to the hardware; available, so that the hardware and software used are easily obtainable by other people; affordable, costs were kept to a minimum in order to increase availability;

and fast, the recognition process had to operate in real-time. Speech was input through a close speaking microphone. After analogue to digital conversion, the signal, sampled at 10kHz, was passed to the digital signal processor board, the output from which was passed onto the Inmos transputer board, containing four transputers, which performs the hidden Markov model processing, lexical access and syntactic processing [Sutherland *et al.*, 1990]. The algorithmic layer handled the division of processes between transputers.

The Osprey design took scalability into account, for example, if a larger vocabulary was employed, additional transputers may be used to handle the data. The system modelled 44 phonemes; first time training involved reading 200 sentences. The flexibility of the system allowed function dependent and context sensitive models to be added. The application layer was airport ground movement control command monitoring, this required a certain amount of speech understanding. A knowledge base contained an up to date state of the airport, in other words the state and locations of the various aircraft. This knowledge base did not play any part in the recognition process; it was only accessed during the intermediary stage between the recognition of a phrase and the reaction of the system.

4.2.18 IBM

IBM are working on automatic speech recognition of continuously read sentences from a naturally occurring corpus: office correspondence. Their recognition system combines features from their previously developed isolated word and continuous speech recognition systems. It consists of an acoustic processor, an acoustic channel model, a language model, and a linguistic decoder. Some new features in the recogniser, relative to the isolated word speech recognition system, include the use of a "fast match" to rapidly prune, to a manageable number, the candidates considered by the detailed match; multiple pronunciations of all function words; and modelling of inter-phoneme coarticulatory behaviour. The test data consisted of 50 sentences from ten male speakers drawn from spontaneously generated memos

covered by a 5000 word vocabulary. The perplexity of the test sentences was calculated to be 93. Preliminary speaker dependent recognition results yielded an average word correct rate of 89.0% [Bahl *et al.*, 1989].

Training was performed by ten male speakers reading training scripts of 2000 sentences fully covered by a 20000 word vocabulary. The first 500 sentences were the same for each speaker, while the remaining 1500 were different from each speaker to speaker. The average sentence length was 16.4 words. It took each speaker approximately one week to record the necessary speech. The acoustic processor extracts a vector of 20 spectral features from the speech signal, and codes each feature vector as one of 200 possible prototype classes. The acoustic channel model describes, in a probabilistic fashion, the way in which words are realised as sequences of prototypes produced by the acoustic processor. The fast match produces a shortlist (thirty on average) of words that match the prototype string.

The language model estimates the probability of the next word in the sentence given the previously hypothesised words. This is the standard IBM trigram model which is based on an interpolation of relative frequencies of trigrams, bigrams and unigrams collected from a 200 million word text database. Each word in the vocabulary has one or more pronunciations associated with it, known as lexemes. Each lexeme is made up of a series of phonemes selected from a phonetic alphabet of size 64. In addition to this phonetic acoustic model, IBM also uses a contextual allophonic acoustic model in which each of the 64 phonemes is realised by a variety of allophones. Each lexeme is then represented by a series of allophones, which in turn are represented by a series of Markov models.

4.3 Existing Systems Used By The Deaf Community For Real-Time Machine Transcription Of Speech

This section outlines those systems, already in use by the deaf community, that provide machine transcription of speech in real-time.

4.3.1 Palantype

The Palantype shorthand machine has been used for some time as a transcription aid for the deaf. Speech is recorded on a 29 key chord (i.e. several key presses are allowed at the same time) keyboard in a special phonetic form, one syllable at a time, and without indicating word boundaries. The original Palantype transcription system was built of standard digital hardware; the output of the system was phonetic codes and hence not very easy to read. Since then, modern computer technology has been used to enhance the system both in terms of the quality of the output and its flexibility. In 1979 the original system was replaced by a microcomputer-based version with a dictionary of approximately 1400 words. The system achieved approximately 70% correct word spelling; those words not appearing in the dictionary were represented by their phonetic spellings.

The system was further improved, with the commercial environment in mind, by increasing the vocabulary size to 10,000 and adding a facility for personalising the dictionary both for individual palantypists and subjects. The applications for the hearing impaired were not forgotten; and care was taken to ensure that the system could work in real-time and thus produce a simultaneous transcript when required. A large screen projection television was added specifically for this use. The improved quality of the output script meant that anyone with normal reading skills could understand the output, and thus a much larger range of hearing impaired people could benefit from the system [Newell and Brooks, 1985]. Subse-

quent use of editing and file handling facilities allowed a perfectly spelt verbatim report of a meeting to be prepared. The current Palantype computer aided transcription system, marketed by Possum Controls Ltd., has a vocabulary of 15,000 words and the performance achievable by a trained operator is normally over 95% words correctly spelt [Newell *et al.*, 1988].

The main drawback of the Palantype system is the length of training required for the stenographers (chord keyboard typists): one to two years. Trained stenographers, though, can achieve transcription rates of up to 200 words per minute. The average delay between a word being spoken and it being displayed on a visual display unit has been measured at 1.9 seconds.

The American Palantype system, Stenotype, uses a 23 key chord keyboard, thus requiring a different phonetic coding method. A slightly simpler transcription system, known as Velotype, produces direct (i.e. not processed by computer) output. A 37 key syllable chord keyboard is used. Proficiency with Velotype can be achieved after six months, with speeds of up to 120 words per minute. [RNID, 1990]

4.3.2 HI-LINC

Hi-Linc is a visual text system for conferences developed at Bristol University. It allows pre-prepared text to be simultaneously displayed with a video image on a television screen as the speaker talks. The speaker can interrupt the system at any time to add additional text. All on screen text is stored and a typed transcript may be subsequently made. Speaker pre-defined abbreviations may be used to further increase the speed of the system. [RNID, 1990]

4.3.3 Speed Typing System

The Speed Typing System developed at the Open University is aimed at providing sufficient accuracy of transcription at sufficient speed. The text that appears on screen is interpreted and not verbatim. In trials, the system has been shown to provide 60%–70% of information items present in the original. A user with only ten minutes training can achieve a high degree of accuracy. This system also directly addresses the requirement of clarity (see page 20): the interpreted summary is in correct English; this may be at a more suitable linguistic level for born-deaf people than a verbatim transcription. [RNID, 1990]

Chapter 5

General Solution

This chapter outlines the general solution adopted to the problem described in chapter 2: phoneme recognition, word lattice generation and word lattice parsing. The novelty of the solution is also addressed. The system that has been developed is known as AURAID. Chapter 6 outlines the detailed solution and gives evidence for the claims made in this chapter.

5.1 Methodology Revisited

Before a description of the general solution is given, it is important to reiterate the methodological approach that has been adopted. The work described in this thesis is guided by the principles of artificial intelligence and natural language engineering. The aim of artificial intelligence research is to simulate successful intelligent human behaviour by any available techniques, not just by modelling human mental behaviour. Natural language engineering is a pragmatic approach to building speech and language processing computer systems. The emphasis is on using current best solutions to solve practical problems. It is desirable for these solutions to be theoretically complete, but it is not essential. Use may be made of local theories, knowledge bases, statistical methods, adaptive methods and even

ad hoc solutions. The goal is to produce practical and usable systems. Should new theories be developed that replace existing solutions, the natural language engineer would take a pragmatic approach and use them where possible, rather than dogmatically holding on to old ideas.

5.2 Phoneme Recognition

Front-end processing of the raw speech signal is to be performed by a continuous speech phoneme recognition system. Research on this has been undertaken in collaboration with two groups, the Defence Research Agency (DRA), and Cambridge University Engineering Department (CUED). In addition, a computer program to simulate the performance of such a front-end has been written. This produces a realistic corruption of a phoneme data stream, to a degree specified as a parameter.

The phoneme was chosen as the interface between the underlying speech recognition hardware and the language processing component for two main reasons. Firstly, it is as high a recognition unit as can possibly be achieved using the least amount of domain dependence. Secondly, it is the most common unit of speech between the acoustic and the word level. Nearly all of the speech recognition systems described in section 4.2 use either context dependent (triphones, for example) or context independent phoneme models. There are only 44 phonemes in English, making it very easy to train a phoneme recogniser using large corpora and still retain domain independence. Using lower units than the phoneme would introduce unnecessary complexity, and reduce the choice of underlying speech recognition hardware. Using higher units than the phoneme would introduce unnecessary domain dependence because of the pre-dominance of statistically trained word recognition systems.

For development purposes, using a simulation is justified because it reduces development time by allowing work on the underlying speech recognition hardware and the word recognition algorithms to be undertaken in parallel at different insti-

tutions. A simulation also provides reproducible input for testing purposes. The question that needs to be asked is does the simulation provide a valid model of a continuous speech phoneme recognition system? We argue that it does because the recognition error probabilities were obtained from an existing continuous speech recognition system; there is a random factor; and the corruption rate is tunable to allow testing of the robustness of the word recognition algorithm to changes in phoneme recognition accuracy. A further reason that demonstrates the independence of the word recognition algorithm from the simulation is that the dynamic programming parameters, used for generating a word lattice, are determined using an adaptive algorithm. A near optimal solution is found automatically for a given set of acoustic-phonetic conditions. This process is described later in this chapter.

5.3 Word Lattice Generation

An appropriate data structure that may be built prior to generating sentence hypotheses is a word lattice [Murveit *et al.*, 1993b] [Baggia *et al.*, 1992] [Ljolje and Riley, 1992]. A word lattice contains the set of word hypotheses produced by the phonemic matching stage. Each word hypothesis is characterised by the start and end points of the spoken utterance portion against which it has been matched, and a score representing its likelihood of occurrence. The word lattice contains many more word hypotheses than the number of actual spoken words and word hypotheses may overlap one another. A simplified example of a word lattice is shown in Table 5.1.

5.3.1 Dynamic Programming

Dynamic time warping is a technique that compensates for variability in the rate at which words are spoken. It is based on a more general computational technique known as dynamic programming. Dynamic programming is used to match each word in the dictionary with a series of phonemes in order to build a lattice of spoken

spoken input	this	course	is	on	software	maintenance
spoken phoneme form	D I s	k O s	I z	Q n	s Q f t w e@ r	m eI n t @ n @ n s

recognised phoneme form	D	s	k	O	I	z	Q	s	Q	f	t	l	e@	r	m	eI	n	t	@	n	n	@	s

word lattice	this	course	is	on	software	maintenance
	earth / ask		us	loss	off / tell	room / an / to / known / as
	these	call			saw / law	may / ten / nice
		carry			soft / air	main
		courses				meant

Table 5.1: A Simplified Example of a Word Lattice

word hypotheses. Dynamic programming is a mathematical concept that has been used for many years for multi-stage optimal decision calculation. In the field of speech recognition it was used initially in isolated word recognition systems for comparison of segments of speech with stored word templates. This was extended to continuous word recognition by storing each template as a series of frames which were then compared to the segments of speech. A detailed description of dynamic programming for speech recognition can be found in [Silverman and Morgan, 1990]. By assuming a continuous stream of phonemes as its input, AURAID does not deal with frames or segments of the speech signal. However, dynamic programming can be used to match stored template words, made up of a series of phonemes, with the input phonemes. Each word is given a score representing its likelihood of matching a particular sequence of input phonemes.

Dynamic programming has become the standard lexical access algorithm for matching dictionary entries against phoneme sequences. Different approaches to using dynamic programming do exist, but they reduce to essentially the same algorithm. One important choice to be made at this stage of processing is whether various sources of knowledge should be incorporated. Many of the systems described in section 4.2 use a multi-pass strategy for word recognition, initially incorporating a cheap (in terms of computational expense) language model, such as a bigram, into

the dynamic programming matching routine to reduce the search space. Further passes bring in more sophisticated, yet expensive, techniques.

The choice that we have made is to separate the dynamic programming algorithm from the contributing knowledge. This has been done for several reasons. Firstly, it is envisaged that many different knowledge sources may be used during the recognition process. Use is already made of syntax and word frequency, additional knowledge could be semantics, prosody and repair. Determining the optimal *serial* combination of these knowledge sources is a very complex task, if achievable at all. It is more likely that they will need to operate in parallel, independently of each other, so that each knowledge source contributes positively and negatively when assessing competing sentence hypotheses. The second reason, therefore, is that knowledge sources may give bonuses as well as penalties when judging the relative merits of different sentence hypotheses. A sentence hypothesis that is penalised by the grammar may be given a bonus by the semantic knowledge source — a balance needs to be achieved between pruning the large search space and ensuring that the correct hypothesis is not eliminated too early. It may be that certain knowledge sources can be brought within the dynamic programming algorithm leading to an improvement in performance.

So, our choice has been to use dynamic programming for building a lattice of word alternatives, and then to use different sources of knowledge during word lattice parsing. The effect of each knowledge source can be clearly identified and various strategies for combining the different knowledge contributions can be developed.

This disadvantage of using dynamic programming for generating a word lattice is that the time involved is proportional to the size of the dictionary being used: each phoneme of each word in the dictionary is matched against the phoneme input. This effect could be reduced by exploiting the fact that this task could be performed in parallel with, for example, a portion of the dictionary on each of several processors. A further problem is that out of vocabulary words are not handled at all, they are simply mis-recognised.

5.3.2 Robust Parameter Estimation

The typical method of determining the likelihood of a word is to collect a corpus of recorded speech for a particular domain and determine *a priori* probabilities for each word or sub-word (i.e. phoneme) pronunciation. The disadvantages of this approach are that the likelihood scores need to be re-calculated for each new domain, this involves collecting a new corpus of recorded speech. In addition, it is unlikely that the acoustic models generated will be robust enough for vocabulary and domain independence.

The approach outlined in this thesis is to use evolutionary algorithms to generate the required parameters for word lattice generation. This involves assessing the quality of a word lattice generated by a given set of parameters. The evolutionary algorithm converges towards a near-optimum parameter solution set for a small set of data (225 words). The advantages of this approach are that the parameters are robust enough to withstand changes in vocabulary and domain. In fact the only dependence is on the performance of the underlying continuous speech phoneme recognition system. Should this be improved, then the evolutionary algorithm may be re-run to automatically generate a new set of parameters.

It is not possible to set these parameters by hand. The use of an adaptive algorithm allows near-optimal values to be determined automatically without the need for a general theory concerning any inter-dependence between the parameters. An adaptive algorithm is the current best solution to this particular problem. Evolutionary programming was chosen as a suitable adaptive algorithm after initial tests showed it out-performed a genetic algorithm.

5.3.3 Dictionary

The dictionary used by AURAID currently has approximately 2600 words. This comprises approximately 1600 words contained within four lectures from the Durham Lecture Corpus, and made up to 2000 words by merging with the most

commonly occurring words from the LOB Corpus that weren't present in the four lectures. For the processing of the LUND Corpus lecture, 600 words were added to the system dictionary. For each word, the system dictionary contains a phoneme pronunciation and one or more syntactic categories, both obtained from the Oxford Advanced Learner's Dictionary.

For practical use, the system dictionary clearly needs to be larger than 2600 words, 5000 words would be a more suitable size, but 2600 words is adequate for development. As an illustration, the first two lectures of a second year course on software engineering contained 1300 unique words, and the first two lectures of a third year course on software engineering contain 1100 unique words. A limitation of current approaches to speech recognition dictionary construction is that they are required to explicitly contain each word that could be recognised. A more sensible approach would be to list only root words, and allow inflected forms and plurals to be generated automatically.

5.4 Word Lattice Parsing

The word lattice generated by the dynamic programming stage contains many paths representing possible interpretations of a spoken sentence. For example, two possible paths through the lattice given in Table 5.1 are

this courses loss off tell air main to known as

these call us on soft law room an ten nice

A beam search is used to expand likely sentence hypotheses from left-to-right across the word lattice; a wider beam, resulting in more expansions, is used initially. The sentence hypotheses are scored using various knowledge sources, and the most promising are expanded by another word. In addition to the phonemic match score determined during word lattice generation, the score of a sentence hypothesis is made up of a grammatical "incorrectness" penalty, word frequency information

and a guess of the remaining penalty likely to be incurred during the expansion of this hypothesis. Other knowledge sources such as semantics, prosody and repair could be included at this stage with little inconvenience.

Rather than use a probabilistic grammar for scoring sentence hypotheses, a set of rules were developed that can be used to check the syntactic *incorrectness* of sequences of words. These rules are collectively known as an "anti-grammar" because the rules are used to penalise certain syntactic constructs rather than identify syntactically correct sequences.

There are two main alternative strategies that could have been chosen at the syntactic checking stage of processing: statistical language models (bigram or trigram), or conventional parsing techniques. In the context of developing a domain independent speech recogniser, we feel that a wholly statistical approach is invalid. It is not possible to build a domain independent n-gram language model simply because, by their inherent nature, statistical language models are only valid in the domain in which they have been trained. It is possible, however, to obtain some domain independent statistical information: the 500 most common domain independent words, for example, could be determined by analysing word frequencies from a variety of different corpora. More detailed information, though, would be too domain specific.

Conventional parsing techniques could not be used in the framework of this research because of the errors contained in spontaneous speech. A parser used for written language processing would not be able to handle repair and filled pauses for example. Approaches to spontaneous speech recognition using partial parsing are suitable for extracting information, such as semantic frame filling in the ATIS domain, but not for word recognition. For recognition of read speech, for example in the WSJ domain, conventional parsing techniques could be used. Conventional parsing is computationally expensive, and so would not be an appropriate technique to use on the large search space contained within a word lattice. The approach to take in this situation would be to generate a word lattice, cut down the search using a statistical language model to generate an n-best list, and then use a full

parse to determine the most likely sentence hypothesis in the list.

The main limitation of using an anti-grammar to reduce the search space is that the correct hypothesis is not always the first choice. Rather than select the most likely hypothesis, the anti-grammar rules out many incorrect hypotheses. Further sources of knowledge may be necessary in order for the actual spoken hypothesis to emerge as the most likely candidate.

Work has recently begun on incorporating a semantic analysis knowledge source into the word lattice parsing stage based on semantic selection [Short *et al.*, 1994a] [Hirst, 1987]. This work is not yet at a level sufficient to be included in this thesis, but is mentioned here for completeness. Semantic selection is the use of the meaning of concepts to prune impossible interpretations of a possibly ambiguous input. Consider, for example, the sentence "green ideas sleep". The adjective "green" cannot be applied to the noun "idea", it is only applicable to concrete concepts and "idea" is abstract. The verb "sleep" requires an animate subject, this is not satisfied by the word "idea". In order to perform full semantic selection, a semantic analyser would first require a full grammatical parse of a sentence in order to build a semantic representation. As was mentioned in section 5.4, conventional parsing is computationally expensive, and is not an appropriate technique to use on the large search space contained within a word lattice.

What is required, therefore, is a form of *weak* semantic selection, in other words, a fast method of partial semantic selection. Two simple observations of English form the basis of this heuristic. Firstly, that adjectives tend to precede the noun to which they are to be applied. Secondly, that the subject and object of a verb tend to be nearby in lexical terms; furthermore, the subject tends to precede the verb and object tends to succeed. Clearly there are exceptions to these observations, but even so, a form of weak semantic selection could be used to penalise certain sentence hypotheses during word lattice parsing.

5.5 Novelty of the Solution

There are several novel aspects to the work described in this thesis. Firstly, the major original contribution in this thesis is that rather than use a probabilistic grammar for scoring sentence hypotheses, anti-grammar rules are used to check the syntactic incorrectness of sequences of words. This has the effect of reducing the large search space, represented as a word lattice, whilst at the same time allowing normal spontaneous English to be spoken. This inverted method of modelling follows naturally from the fact that it makes sense to keep the size of the model to a minimum for efficiency reasons. For a constrained task it is efficient to model the few legal sentences, but once the balance changes, so that there are more legal sentences than illegal ones, it is more efficient to model the smaller set of illegal sentences.

Secondly, the system has been designed to allow ease of integration with new sources of knowledge, such as semantics, prosody or repair, in effect, providing a test-bench for determining the impact of different knowledge upon word lattice parsing.

Thirdly, the use of evolutionary programming to determine near-optimal robust parameters for word lattice creation removes the need for retraining word acoustic models on large corpora of data each time the vocabulary or domain changes. Instead, the only dependence is on the performance of the underlying continuous speech phoneme recognition system; the parameters are robust.

The next chapter gives more detail on the ideas mentioned in this chapter and also provides further evidence for the claims made in this section.

Chapter 6

Detailed Solution

This chapter describes in detail the solution outlined in chapter 5: phoneme recognition using a simulation and also the AURIX and CU-CON systems; word lattice generation using dynamic programming with robust parameter estimation obtained using evolutionary programming, and the system dictionary; and word lattice parsing using a beam search and contributing knowledge such as the anti-grammar and word frequency information. A detailed discussion of the anti-grammar is presented. The software engineering aspects of the test-bench are also addressed with reference to integration of new knowledge sources and maintenability of the underlying representations.

6.1 Phoneme Recognition

The raw speech signal is first processed by a continuous speech phoneme recognition system. The two groups with whom collaborative research has been undertaken are the Defence Research Agency (DRA), and Cambridge University Engineering Department (CUED). Performance details of such systems are mentioned in section 4.2. The phoneme recognition systems under development by these two groups are described below. For the development of the research outlined in this thesis, a

computer program was written to simulate the front-end phoneme recogniser.

6.1.1 The AURIX System (DRA)

AURIX is a speech recognition system configurable for many different applications. In this research, it is used as a real-time continuous speech phoneme recognition system. It is based on work by the DRA as part of their Airborne Reconnaissance Mission (ARM) continuous speech recognition project. The aim of the ARM project is the accurate recognition of continuously spoken airborne reconnaissance reports [Russell *et al.*, 1990b]. The project uses a speech recognition system based on phoneme-level hidden Markov models, and is described in section 4.2.16. A large corpus of speech was collected in order to support future work on task independent and large vocabulary speech recognition [Browning *et al.*, 1991] and this was used as training data for AURIX [Russell, 1992a].

The current version of AURIX yields approximately 40% phoneme recognition accuracy and is not yet suitable for providing a front-end for the remainder of the research described in this thesis. Recognition is performed in real-time, however, and the equipment is in place ready for an improvement in the phoneme modelling software.

6.1.2 The CU-CON System (CUED)

CU-CON is a speaker independent speech recognition system developed at CUED being developed for the ARPA Speech Recognition Evaluations, described in section 4.2.6. Recently, collaborative work has been undertaken between Durham University and CUED on producing a British English pronunciation dictionary (BEEP) for use by CU-CON and other researchers. In addition, CU-CON can be configured to recognise phonemes.

Phoneme recognition performance using British English has not yet been calcu-

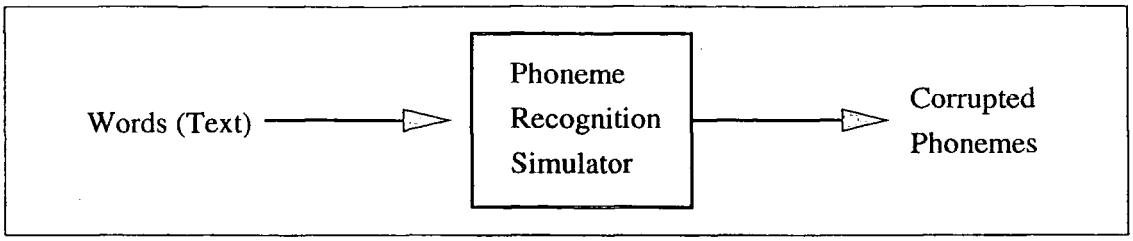


Figure 6.1: The Phoneme Recognition Simulator

lated, however on American English, CU-CON achieved phoneme recognition rates of 73% on the TIMIT acoustic-phonetic continuous speech corpus [Robinson *et al.*, 1994] [Robinson, 1992]. In the intervening time since these results were published, the system has been trained on a large amount of American English speech data for the ARPA evaluations, and also a parallel version is being developed for British English. Phoneme recognition rates for both of these systems are expected to exceed 75%[1]. Direct connection with the CU-CON phoneme recogniser has not yet been attempted because of hardware requirements.

6.1.3 Simulation

In order to develop the word lattice generation and parsing components in isolation from the main phoneme recognition hardware, a program, written in PERL, was constructed for simulation purposes (see Figure 6.1). The purpose of the program is to corrupt a sequence of phonemes to a specified degree. This is an off-line process, and independent of word lattice generation and parsing. Although the corruption is performed with a certain amount of randomness, it is based on the kinds of errors made by existing phoneme recognition systems, in that particular classes of phonemes are easier to recognise than others, and substitution of one phoneme for another usually occurs within classes, in other words vowels are mainly confused for vowels, and plosives for plosives [Browning *et al.*, 1990]. The phoneme classes are shown in Table 6.2. Corruption is evenly spread throughout the phoneme input, and a maximum rate of corruption for a word can be specified.

[1] Personal communication with A.J. Robinson, the primary researcher involved.

An example of the corruption produced by the simulation program is given below.

Words	for	this	lecture	we're	going
Original Phonemes	f O r	D I s	l E k tS @ r	w I@ r	g @U I N
Corrupted Phonemes	f U@ r	D I s	l E k tS r	w U I@ r	d g @U I N

Words	to	be	looking	at	maintenance	models
Original Phonemes	t @	b i	l U k I N	{ t	m eI n t @ n @ n s	m Q d l z
Corrupted Phonemes	t @	b i	l U g I N	{ t	eI n @ n @ m s	m eI d z

Details of the phoneme corruption:

```
Number of phonemes: 44

                NUM  SUBS  DELS  INS
plosives    =     8     1     1    1
affrics     =     1     0     0    0
strfrics    =     3     0     0    0
wkfrics     =     2     0     0    0
liquids     =     7     0     1    0
nasals      =     7     1     1    0
vowels      =    16     2     1    1
-----------------------------------------
TOTALS      =           4     4    2 (10)
TOTALS (%)  =         9.1   9.1  4.5 (22.7)
```

After corruption, the example sentence contains 22.7% phoneme error, consisting of 9.1% substitutions, 9.1% deletions and 4.5% insertions. It must be made clear that although word breaks are used by the phoneme corruption program, they are invisible to the word recognition system which treats the corrupted sequence as a continuous stream of phonemes.

6.2 Word Lattice Generation

A word lattice is a data structure that holds detailed information resulting from the lexical matching (word hypothesis) routine of a speech recognition system (see Figure 6.6). Informally, a set of words are each compared with acoustic/phonetic data. Each word is assigned a score indicating the closeness of match to a particular portion of data. Paths may be traced (parsed) though the word lattice by joining

up words that span consecutive portions of data to form sentence hypotheses.

6.2.1 An Example Word Lattice

The essential components of an entry in a word lattice are

- a word reference, either the actual word string or a pointer to a dictionary-like list;
- the start point of this particular entry;
- the end point of this particular entry (if the data being matched is phonemes rather than acoustic data, this could be inferred from the phoneme length of the particular word);
- a score indicating how close the word matches a particular portion of acoustic/phonetic data.

Words may appear more than once in a word lattice, by, for example, starting at the same point in the lattice but spanning different amounts of acoustic/phonetic data.

Table 5.1 is a high-level diagrammatical view of a word lattice. It shows how words span portions of the phoneme data. The position of the words on different levels in this simplified lattice is not too significant, in reality each word in a box would have associated with it a score representing how well it matches the phonemes spanned by the box. Several paths can be traversed through the lattice from the beginning to the end in addition to the correct path, for example, "this courses loss off tell air main to known as", or "these call us on soft law room an ten nice". A more detailed example is given in Table 6.1. This shows the word lattice generated for the part-sentence "the word", which is represented in phonemes as `D @ w 3 d`. For readability, there is some redundancy in the amount of information that is presented. Each field is described below.

1	2	3	4	5	6	7
frame = 1	the	ADV	Pu	start = 1	end = 2	score = 0.0

1. This field contains the frame at which subsequent word entries begin. In our work, each frame represents an individual phoneme.

2. This field contains the actual word that has been matched against a portion of acoustic/phonetic data.

3. This field represents the broad grammatical category of the word, possible values are article (ART), conjunction (CONJ), pronoun (PRON), preposition (PREP), noun (NOUN), verb (VERB), adverb (ADV), adjective (ADJ) and interjection (INTERJ).

4. This field contains the OALD part of speech (POS) code, representing a finer grammatical categorisation than the previous field.

5. This field contains the start frame of the data that this entry has been matched against.

6. This field contains the end frame of the data that this entry has been matched against.

7. This field contains the score obtained by matching the phoneme representation of the word against a portion of the data (see section 6.2.3).

6.2.2 Why Make Use of a Word Lattice?

A word lattice is a convenient intermediate data structure between the construction of word-level hypotheses and the construction of sentence-level hypotheses. A word lattice summarises the information obtained from the acoustic-phonetic and word hypothesis stages. In addition, the quality of a word lattice can be determined (see section 7.2). During development, word lattice generation and word lattice parsing

```
frame = 1
              the ADV    Pu  start = 1    end = 2    score = 0.0
              the ART    R-  start = 1    end = 2    score = 0.0
              the ADV    Pu  start = 1    end = 1    score = 5.0
              the ART    R-  start = 1    end = 1    score = 5.0
             they PRON   QN  start = 1    end = 1    score = 5.0
           though ADV    Pu  start = 1    end = 1    score = 5.0
           though CONJ   V-  start = 1    end = 1    score = 5.0
           either ADJ    OA  start = 1    end = 2    score = 5.0
           ...
frame = 2
                a ART    S-  start = 2    end = 2    score = 0.0
             away ADJ    OA  start = 2    end = 3    score = 1.7
             away ADV    P+  start = 2    end = 3    score = 1.7
             away ADJ    OA  start = 2    end = 4    score = 1.7
             away ADV    P+  start = 2    end = 4    score = 1.7
               an ART    S-  start = 2    end = 2    score = 5.0
frame = 3
             word VERB   H0  start = 3    end = 5    score = 0.0
             word NOUN   K6  start = 3    end = 5    score = 0.0
            words VERB   Ha  start = 3    end = 5    score = 2.5
            words NOUN   Kj  start = 3    end = 5    score = 2.5
            world NOUN   K6  start = 3    end = 5    score = 2.5
             were VERB   Gc  start = 3    end = 4    score = 3.3
             were VERB   Ic  start = 3    end = 4    score = 3.3
             word VERB   H0  start = 3    end = 4    score = 3.3
              ...
frame = 4
            heard VERB   Jc  start = 4    end = 5    score = 3.3
            heard VERB   Jd  start = 4    end = 5    score = 3.3
            third NOUN   K6  start = 4    end = 5    score = 3.3
            third ADJ    OA  start = 4    end = 5    score = 3.3
frame = 5
              add VERB   J0  start = 5    end = 5    score = 5.0
                d NOUN   Ki  start = 5    end = 5    score = 5.0
              day NOUN   M6  start = 5    end = 5    score = 5.0
              die VERB   I5  start = 5    end = 5    score = 5.0
              die NOUN   K6  start = 5    end = 5    score = 5.0
               do VERB   G5  start = 5    end = 5    score = 5.0
              'do VERB   J5  start = 5    end = 5    score = 5.0
               do NOUN   K6  start = 5    end = 5    score = 5.0
              i'd VERB   Gf  start = 5    end = 5    score = 5.0
```

Table 6.1: An Example Word Lattice

Class	*Name*	*Phonemes*
0	Plosive	p b t d k g
1	Affricative	tS dZ
2	Strong Fricative	s z S Z
3	Weak Fricative	f v T D h
4	Liquid/Glide	l r w j
5	Nasal	n m N
6	Vowel	i I E { A Q O U u 3 V @ aI eI oI aU @U I@ e@ U@

Table 6.2: Phoneme Classes used by AURAID

can be investigated in isolation, saving much time, by using a word lattice as an intermediate representation, stored in a file.

Towards the end of recognition, many systems make use of an n-best list, in other words a list of the best scoring n sentences, this is described in more detail in section 3.6. The two representations are equivalent, an n-best list can be reduced to a word lattice, and an n-best can be created by tracing paths (sentences) through a word lattice.

6.2.3 Dynamic Programming

There are three main approaches to using dynamic programming for continuous speech recognition: the two level algorithm [Sakoe, 1979], the level building algorithm [Myers and Rabiner, 1981], and the one pass algorithm [Bridle *et al.*, 1982]. Although each differs in detail, the two basic stages involved in each algorithm are word level analysis and phrase level analysis. In word level analysis, each word in the dictionary is matched against all possible (consecutive) sequences of the input phonemes. Phrase level analysis determines the best scoring sequence of words that spans the entire phoneme input. These two stages comprise the two level algorithm, the others being optimisations that integrate the two stages.

In AURAID a word level analysis using dynamic programming is undertaken, as

described above, but a beam search is used for the phrase level analysis. The word level analysis algorithm models explicitly the kinds of errors which may occur, both within words and between words. That is inserted phonemes, deleted phonemes and substituted phonemes. The distance or similarity score between phonemes can depend on a variety of factors, and varies from algorithm to algorithm. Most algorithms group phonemes into classes according to their confusability. The phoneme classes used by AURAID are based on manner of articulation and are shown in Table 6.2. The distance between phonemes within the same class is then less than that between phonemes from different classes. This can be measured, for example, by absolute values or logarithms of the probability of confusing one phoneme for another based on experimental data. It was found that long words are unduly penalised because of their length. To overcome this inadequacy the distance scores are normalised according to the length of the word being considered. The equations used in the word level analysis algorithm are:

$$
\begin{aligned}
S(w,1,t) = \min\{\quad & \frac{ins\_pen}{N(w)} + \frac{sub\_pen(w,1,t)}{N(w)} + \min_{r\in R}\{S(r,N(r),t-2)\}; \\
& \frac{sub\_pen(w,1,t)}{N(w)} + \min_{r\in R}\{S(r,N(r),t-1)\}; \\
& \frac{del\_pen}{N(w)} + \frac{sub\_pen(w,1,t)}{N(w)} + \min_{r\in R}\{S(r,N(r)-1,t-1)\}; \\
& \frac{2.0\times del\_pen}{N(w)} + \frac{sub\_pen(w,1,t)}{N(w)} + \min_{r\in R}\{S(r,N(r)-2,t-2)\}\ \}
\end{aligned}
\tag{6.1}
$$

$$
\begin{aligned}
S(w,2,t) = \min\{\quad & \frac{ins\_pen}{N(w)} + \frac{sub\_pen(w,2,t)}{N(w)} + S(w,1,t-2); \\
& \frac{sub\_pen(w,2,t)}{N(w)} + S(w,1,t-1); \\
& \frac{del\_pen}{N(w)} + \frac{sub\_pen(w,2,t)}{N(w)} + \min_{r\in R}\{S(r,N(r),t-1)\}; \\
& \frac{2.0\times del\_pen}{N(w)} + \frac{sub\_pen(w,2,t)}{N(w)} + \min_{r\in R}\{S(r,N(r)-1,t-2)\}\ \}
\end{aligned}
\tag{6.2}
$$

$$S(w,3,t) = \min\{ \quad \frac{ins\_pen}{N(w)} + \frac{sub\_pen(w,3,t)}{N(w)} + S(w,2,t-2);$$
$$\frac{sub\_pen(w,3,t)}{N(w)} + S(w,2,t-1); \qquad (6.3)$$
$$\frac{del\_pen}{N(w)} + \frac{sub\_pen(w,3,t)}{N(w)} + S(w,1,t-1);$$
$$\frac{2.0 \times del\_pen}{N(w)} + \frac{sub\_pen(w,3,t)}{N(w)} + \min_{r \in R}\{S(r,N(r),t-2)\} \ \}$$

$$S(w,p,t) = \min\{ \quad \frac{ins\_pen}{N(w)} + \frac{sub\_pen(w,p,t)}{N(w)} + S(w,p-1,t-2);$$
$$\frac{sub\_pen(w,p,t)}{N(w)} + S(w,p-1,t-1); \qquad (6.4)$$
$$\frac{del\_pen}{N(w)} + \frac{sub\_pen(w,p,t)}{N(w)} + S(w,p-2,t-1);$$
$$\frac{2.0 \times del\_pen}{N(w)} + \frac{sub\_pen(w,p,t)}{N(w)} + S(w,p-3,t-2) \ \}$$

where $S(w,p,t)$ represents the score for phoneme p of word w when matched against input phoneme t, R is the set of words in the dictionary used by AURAID and $N(r)$ is the length in phonemes of the r'th word. The three penalties, *ins_pen*, *del_pen* and *sub_pen* each return absolute values. For *ins_pen* and *del_pen*, this is independent of the particular phoneme being considered. *sub_pen* is divided into two separate cases: the first of these cases penalises substitutions in which the phonemes are of the same class; while the second case allows a different penalty to be used for phonemes which were substituted with ones of a different class. There are, therefore, four penalty values to be chosen. In previous work [Collingham and Garigliano, 1993] these settings were selected by hand and this approach to phoneme distance calculation produced better results than using logarithms which used the probability of confusing one phoneme for another. A further problem with using logarithms is that it necessitates a detailed assessment of the performance of the underlying phoneme recogniser to determine phoneme confusion likelihoods and

the like. In the next section an automatic approach to determining near-optimal settings for these parameters is presented. The data structure resulting from the dynamic programming stage is called a word lattice.

Equation 6.4 is the general equation used for dynamic programming matching, equations 6.1, 6.2 and 6.3 being for words of phoneme length 1, 2 and 3 respectively. In the general equation, a minimum score choice is taken between: the previous input phoneme being an insertion error; the current input phoneme being correct or a substitution error; or a deletion of the previous phoneme of the current word. In addition, the last line of each equation represents the occurrence of two consecutive deletion errors. Consecutive insertion errors are not modelled because they are not produced by the simulated phoneme recogniser, although this would only require a simple extension to the equations. For short words, the first three equations perform the same calculation as the general equation but look back at previous words to determine what, if any, error has taken place. Finally, for each input phoneme the end score for each word is adjusted to represent the local score for that word if it were to end at that point in the input.

It is possible to analyse the performance of the word lattice generation algorithm in a variety of different phoneme error situations. There are three possible single phoneme error situations — deletion (D), insertion (I) or substitution (S). Extending this to two consecutive phoneme errors gives a further nine possible double phoneme error situations — DD, DI, DS, ID, II, IS, SD, SI, SS. This is reduced to seven possible situations because a deletion followed by an insertion (DI) and an insertion followed by a deletion (ID) are equivalent to a single substitution. We can examine the initial fragment of the word lattice to ensure that the word lattice generation algorithm handles the ten error situations sufficiently.

In the following paragraphs, lattice is generated for each of the ten error situations described above in addition to the "no error" situation. The input will be various corrupted forms of the word "best", which is made up of the phonemes `b E s t`. According to the Oxford Advanced Learner's Dictionary, "best" can be a transitive verb (code H0), a superlative adjective (Os), an adverb (Pu) or a pro-

noun (Qx). The parameters used for this analysis are set as follows: 10.0 for an insertion, deletion or within-class substitution error, and 50.0 for an out-of-class substitution error. The values of the parameters have been chosen for simplicity to demonstrate the word lattice generation algorithm. The word score is calculated by dividing any penalty by the length of the word (in phonemes).

Other factors, such as estimated word frequency, are also taken into account before the word lattice is parsed. Common words are brought nearer the top of the lattice, and rare words are pushed nearer the bottom of the lattice. This is not shown here for simplicity, but is described in section 6.3.4.

No Errors

Phoneme input: b E s t

```
frame = 1
              best VERB   H0  start = 1    end = 4    score = 0.0
              best ADJ    Os  start = 1    end = 4    score = 0.0
              best ADV    Pu  start = 1    end = 4    score = 0.0
              best PRON   Qx  start = 1    end = 4    score = 0.0
```

Deletion Error

Phoneme input: b s t

```
frame = 1
            based VERB   Hc  start = 1    end = 3    score = 2.5
            based VERB   Hd  start = 1    end = 3    score = 2.5
             best VERB   H0  start = 1    end = 3    score = 2.5
             best ADJ    Os  start = 1    end = 3    score = 2.5
             best ADV    Pu  start = 1    end = 3    score = 2.5
             best PRON   Qx  start = 1    end = 3    score = 2.5
```

Insertion Error

Phoneme input: b E z s t

```
frame = 1
            bells VERB   Ha  start = 1    end = 3    score = 2.5
            bells NOUN   Kj  start = 1    end = 3    score = 2.5
             best VERB   H0  start = 1    end = 5    score = 2.5
             best ADJ    Os  start = 1    end = 5    score = 2.5
             best ADV    Pu  start = 1    end = 5    score = 2.5
             best PRON   Qx  start = 1    end = 5    score = 2.5
```

Substitution Error

Phoneme input: `b i s t`

```
frame = 1
               b NOUN   Ki  start = 1    end = 2    score = 0.0
              be VERB   G5  start = 1    end = 2    score = 0.0
              be VERB   I5  start = 1    end = 2    score = 0.0
            based VERB   Hc  start = 1    end = 4    score = 2.5
            based VERB   Hd  start = 1    end = 4    score = 2.5
             best VERB   H0  start = 1    end = 4    score = 2.5
             best ADJ    Os  start = 1    end = 4    score = 2.5
             best ADV    Pu  start = 1    end = 4    score = 2.5
             best PRON   Qx  start = 1    end = 4    score = 2.5
```

Deletion-Deletion Error

Phoneme input: `b t`

```
frame = 1
              beat VERB   J5  start = 1    end = 2    score = 3.3
              beat VERB   Jc  start = 1    end = 2    score = 3.3
              beat NOUN   K6  start = 1    end = 2    score = 3.3
              beat ADJ    Oq  start = 1    end = 2    score = 3.3
(24 lines deleted)
            battle VERB   I2  start = 1    end = 2    score = 5.0
            battle NOUN   M6  start = 1    end = 2    score = 5.0
              best VERB   H0  start = 1    end = 2    score = 5.0
              best ADJ    Os  start = 1    end = 2    score = 5.0
              best ADV    Pu  start = 1    end = 2    score = 5.0
              best PRON   Qx  start = 1    end = 2    score = 5.0
```

Deletion-Substitution Error

Phoneme input: `b z t`

```
frame = 1
                b NOUN   Ki  start = 1    end = 1    score = 5.0
               be VERB   G5  start = 1    end = 1    score = 5.0
               be VERB   I5  start = 1    end = 1    score = 5.0
(11 lines deleted)
            based VERB   Hc  start = 1    end = 3    score = 5.0
            based VERB   Hd  start = 1    end = 3    score = 5.0
             best VERB   H0  start = 1    end = 3    score = 5.0
             best ADJ    Os  start = 1    end = 3    score = 5.0
             best ADV    Pu  start = 1    end = 3    score = 5.0
             best PRON   Qx  start = 1    end = 3    score = 5.0
```

Insertion-Insertion Error

Phoneme input: b E z s s t

```
frame = 1
            bells VERB   Ha  start = 1    end = 3    score = 2.5
            bells NOUN   Kj  start = 1    end = 3    score = 2.5
                b NOUN   Ki  start = 1    end = 1    score = 5.0
               be VERB   G5  start = 1    end = 1    score = 5.0
               be VERB   I5  start = 1    end = 1    score = 5.0
(13 lines deleted)
            beams VERB   Ja  start = 1    end = 3    score = 5.0
            beams NOUN   Kj  start = 1    end = 3    score = 5.0
             best VERB   H0  start = 1    end = 3    score = 5.0
             best ADJ    Os  start = 1    end = 3    score = 5.0
             best ADV    Pu  start = 1    end = 3    score = 5.0
             best PRON   Qx  start = 1    end = 3    score = 5.0
```

Insertion-Substitution Error

Phoneme input: b E i z t

```
frame = 1
                b NOUN   Ki  start = 1    end = 1    score = 5.0
               be VERB   G5  start = 1    end = 1    score = 5.0
               be VERB   I5  start = 1    end = 1    score = 5.0
(16 lines deleted)
            beams VERB   Ja  start = 1    end = 4    score = 5.0
            beams NOUN   Kj  start = 1    end = 4    score = 5.0
             best VERB   H0  start = 1    end = 5    score = 5.0
             best ADJ    Os  start = 1    end = 5    score = 5.0
```

```
          best ADV    Pu  start = 1    end = 5    score = 5.0
          best PRON   Qx  start = 1    end = 5    score = 5.0
```

Substitution-Deletion Error

Phoneme input: b i t

```
frame = 1
             b NOUN   Ki  start = 1    end = 2    score = 0.0
            be VERB   G5  start = 1    end = 2    score = 0.0
            be VERB   I5  start = 1    end = 2    score = 0.0
          beat VERB   J5  start = 1    end = 3    score = 0.0
          beat VERB   Jc  start = 1    end = 3    score = 0.0
          beat NOUN   K6  start = 1    end = 3    score = 0.0
          beat ADJ    Oq  start = 1    end = 3    score = 0.0
          beam VERB   J0  start = 1    end = 2    score = 3.3
          beam NOUN   K6  start = 1    end = 2    score = 3.3
(41 lines deleted)
        battle VERB   I2  start = 1    end = 3    score = 5.0
        battle NOUN   M6  start = 1    end = 3    score = 5.0
          best VERB   H0  start = 1    end = 3    score = 5.0
          best ADJ    Os  start = 1    end = 3    score = 5.0
          best ADV    Pu  start = 1    end = 3    score = 5.0
          best PRON   Qx  start = 1    end = 3    score = 5.0
```

Substitution-Insertion Error

Phoneme input: b i z s t

```
frame = 1
             b NOUN   Ki  start = 1    end = 2    score = 0.0
            be VERB   G5  start = 1    end = 2    score = 0.0
            be VERB   I5  start = 1    end = 2    score = 0.0
         beams VERB   Ja  start = 1    end = 3    score = 2.5
         beams NOUN   Kj  start = 1    end = 3    score = 2.5
          beam VERB   J0  start = 1    end = 2    score = 3.3
          beam NOUN   K6  start = 1    end = 2    score = 3.3
(32 lines deleted)
         based VERB   Hc  start = 1    end = 5    score = 5.0
         based VERB   Hd  start = 1    end = 5    score = 5.0
          best VERB   H0  start = 1    end = 5    score = 5.0
          best ADJ    Os  start = 1    end = 5    score = 5.0
          best ADV    Pu  start = 1    end = 5    score = 5.0
          best PRON   Qx  start = 1    end = 5    score = 5.0
```

6.2.4 Robust Parameter Estimation

Over the past 30 years, three main streams of evolutionary algorithm have been independently developed: genetic algorithms [Holland, 1975], evolutionary programming [Fogel *et al.*, 1966] [Fogel, 1992], and evolutionary strategies (recent review by [Bäck *et al.*, 1991]). Each of these has been inspired by the search processes of biological evolution, and have led to robust optimisation techniques that have been successfully applied to a wide range of problems.

A well known general purpose heuristic search algorithm such as hill climbing can encounter difficulties with parameter optimisation.

> *... hill climbing suffers from various problems. These problems are most conspicuous when hill climbing is used to optimize parameters.*
>
> [Winston, 1992]

One of the typical problems is with an optimal point that turns out to be a local maximum rather than a global maximum.

By maintaining a population of solutions, an evolutionary algorithm is able to exploit those that are promising while exploring other regions of the search space. In this way a parallel search is achieved. Over successive iterations, new solutions are produced as variations of those that have survived to that point in time, and the worst solutions are probabilistically pruned using a "survival of the fittest" strategy (analogous to natural selection). In this way, the population evolves toward optimal solutions.

Genetic algorithms and evolutionary programming, though both inspired by the search processes of natural evolution, each place a different emphasis on what is believed to be driving the evolutionary process. Genetic algorithms model specific genotypic transformations while evolutionary programming emphasises phenotypic adaptation. The genotype being the underlying representation used to encode a possible solution, while the phenotype is its realisation. For example, the information contained in human genes is the genotype, and the human form the

corresponding phenotype.

When using genetic algorithms (GAs), solutions are usually represented as binary strings. The underlying hypothesis of GAs is that by combining subsections of solutions, short highly fit segments of each binary string are propagated throughout the population, and combine to form larger fitter segments of each binary string. This is known as the building block hypothesis [Goldberg, 1989], and is a fundamental principle of GAs. The evolutionary programming (EP) perspective of the evolutionary process is very different from the bottom up approach of GAs. By determining how well solutions are performing in the current environment, improvements are made via a flow of information from the environment back to the underlying genotypic representation. The emphasis is, therefore, on phenotypic adaptation rather than genotypic transformation. In this way a top-down approach to solution improvement is adopted as opposed to the bottom-up approach of GAs.

Previous applications of evolutionary algorithms to natural language processing problems have shown early success [Nettleton and Garigliano, 1994]. The approach offers the adaptability which is often absent from purely symbolic approaches, while at the same time attempting to make the most of well constructed theories [Garigliano and Nettleton, 1994]. The work presented in the remainder of this section considers the application of EP to the problem of finding the required dynamic programming parameters for word lattice generation (see Figure 6.2), and is part of a paper written by the author and a colleague [Nettleton and Collingham, 1995].

Holland [Holland, 1975] identifies the following four components of an adaptive system: an environment of the system; a set of structures; a measure of the performance of each structure; an adaptive plan. How these concepts are mapped is discussed in the remainder of this section.

The environment is a continuous stream of phonemes from which a word lattice is generated according to the algorithms discussed above. Each solution is represented by four floating point numbers which are constrained to be in the range

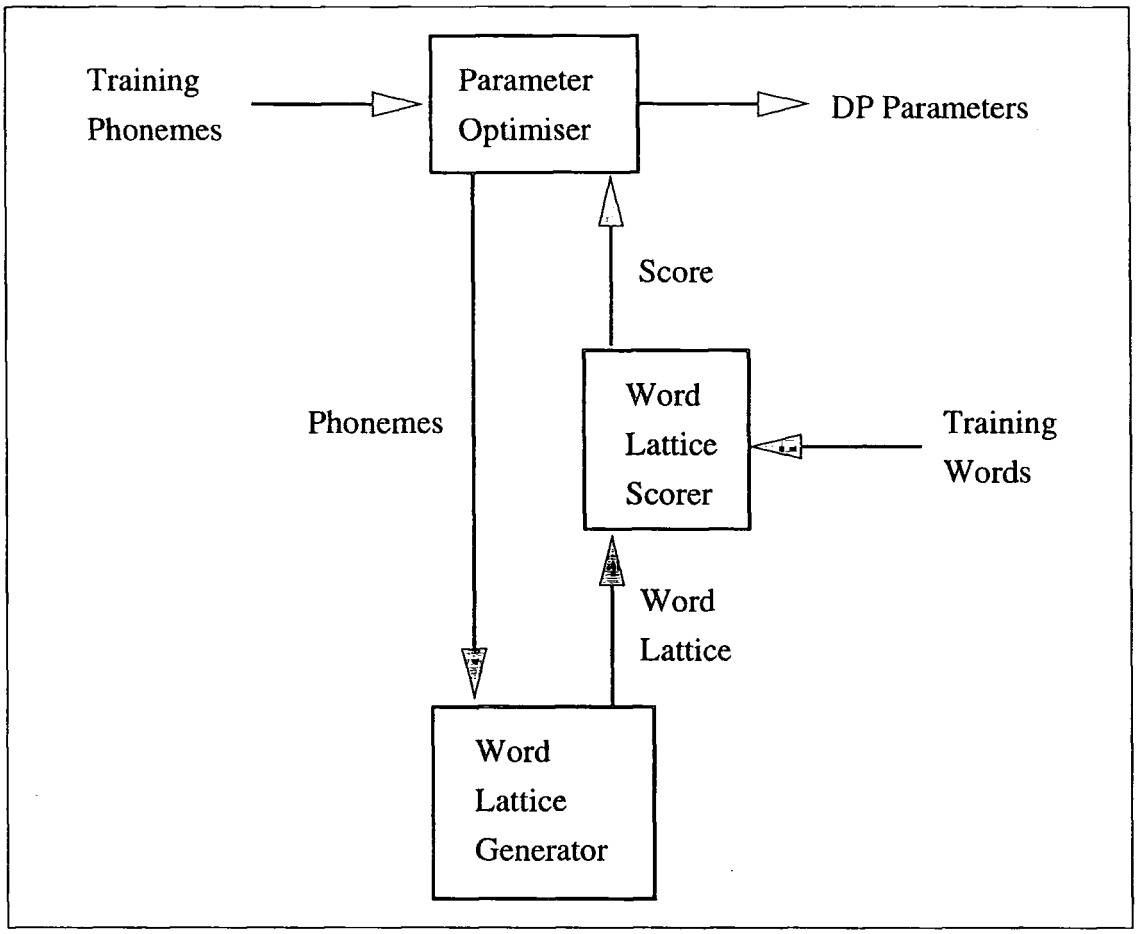


Figure 6.2: The Dynamic Programming Parameter Optimiser

[1,256] — each of these corresponds to the penalties discussed in section 6.2.3. In practice it isn't necessary to restrict the parameter range, but this was done in order to allow for future comparison with GAs.

A fitness measure is needed in order to determine the fitness of each solution within the current environment. This takes into account the average rank of correct words — this measure is explained in detail in section 7.2. Fitnesses were calculated according to the formula:

$$fitness = \frac{10.0 \times rank1 + 10.0 \times rank2}{2.0}$$

where $rank1$ and $rank2$ are the average rank of the correct words in the lattice for

the two different data sets used to estimate the dynamic programming parameters.

Each parent solution in the population is mutated by an amount governed by its fitness to produce a child solution. Fitter solutions must be less likely to be mutated to the same degree as less fit parents, and so each component, x_i, of a solution X, is mutated according to the formula:

$$x_i' = x_i + \sqrt{\text{fitness}(X)} \cdot N(0,1) \qquad i \in \{1,...,4\} \tag{6.5}$$

where fitness(X) is the fitness of solution X and $N(0,1)$ is a standard normal random variable. The above formula was selected since it allows for solutions with a poor fitness to be mutated by a large amount, while at the same time reducing the chance that the mutated parameters fall outside of the permitted range [1,256]. Should a mutation result in a parameter falling outside of this range then it is set to the nearest allowable value. A tournament means of selection is then used to probabilistically prune the worst solutions.

The following is an outline of the evolutionary program used.

1. Randomly initialise a parent population of solutions. Each solution is represented using 4 floating point numbers which are constrained to the range [1,256].

2. Evaluate each member of the parent population using the fitness function discussed.

3. Mutate each member of the parent population, by an amount related to its fitness, to give a member of the child population.

4. Evaluate each member of the child population.

5. For each member of the child and parent populations:

(a) Select at random a number, TOURN, of solutions from the parent and child populations.

(b) Count the number of these solutions whose fitness is less than or equal to that of the current selected solution. This number is the score for the selected solution.

6. Rank the scores of the solutions.

7. Select the solutions which rank in the top half of the list and replace the parent population with these solutions.

8. If the termination criteria is not met then go to step 3.

Comparison tests between genetic algorithms and evolutionary programming for parameter optimisation have been performed in detail [Nettleton and Collingham, 1995] and are described below. For both the GA and EP a population of 50 was used, and each was executed over 50 generations. The tournament size for EP was set at three. For each of the GA and EP, 11 trials were carried out using 20% phoneme corruption, and 31 trials for each of 30% corruption and 40% corruption. An analysis of the results showed that the evolutionary programming algorithm outperformed the genetic algorithm, but statistical tests showed that the differences were not significant.

Comparison of Genetic Algorithm and Evolutionary Programming

This section presents the results of applying a GA and EP to the problem of estimating the penalties of equations 6.1–6.4. Other than variations in solution representation and the details of the EP's mutation operator (discussed below), the GA and EP used are identical to those described in the standard texts mentioned above. The problem used a data set of 113 words corrupted to varying degrees and a dictionary of 1984 words.

In implementing the GA, the subsymbolic representation adopted is that of a binary string. Each of the penalties is encoded as a binary string of length

eight, and these are concatenated together to form one string. Since there are four penalties to be encoded the size of the subsymbolic representation's search space is $256^4 \approx 4 \times 10^9$.

In applying EP to the penalty optimisation problem a real-valued subsymbolic representation is adopted. Each of the penalties are stored as real numbers (six decimal places), and are constrained to the range [1,256]. A child is produced from a parent by mutating each parameter x_i according to equation 6.5 described above.

For both the GA and EP a population of 50 was used, and they were executed over 50 generations. The tournament size for EP was set at three. For each of the GA and EP, 11 trials were carried out using `corrupt20`, and 31 trials for each of `corrupt30` and `corrupt40`. The fitness of the best solution found in each of the runs is shown in Table 6.3 together with the generation at which the best solution was discovered (in parenthesis). The mean and standard deviation of each set of results is also given.

The Figures 6.3, 6.4 and 6.5 each show the online and offline performance of the median run of the GA and EP for the data `corrupt20`, `corrupt30` and `corrupt40` respectively. The offline performance is the average fitness of all of the solutions in a particular generation, while the online performance is the average fitness of all solutions that have been generated up to a certain generation.

The results of the trials conducted with `corrupt20` showed that in each trial both the GA and EP found optimal or near optimal solutions. No difference in performance was observed.

A comparison of the performance of the GA and EP for `corrupt30` indicate that EP outperformed the GA. The result was not statistically significant ($t = 1.04$ with $DF = 52$ gave $P > 0.1$) unless the EP outlier (2.3) and the GA outlier (2.0) were removed ($t = 2.36$ with $DF = 56$ gave $P < 0.05$).

With `corrupt40` the results obtained showed that EP outperformed the GA. The result was not statistically significant ($t = 1.31$ with $DF = 59$ gave $P > 0.1$)

unless the EP outlier (2.9) was removed ($t = 2.17$ with $DF = 54$ gave $P < 0.05$).

The statistical test that was applied was a Smith-Satterthwaite modified one tailed t-test, DF indicates the number of degrees of freedom [Weiss and Hassett, 1991].

Table 6.3: The best solutions found by each the GA and EP for various levels of phoneme corruption. Each algorithm was run 31 times (except for the data file `corrupt20` which was run 11 times) and the generation at which the best solution was found is shown in parenthesis.

Corruption	20%		30%		40%	
Algorithm	EP	GA	EP	GA	EP	GA
Fitness of best solution found	1.1 (0)	1.1 (0)	1.4 (21)	1.4 (42)	2.3 (22)	2.6 (25)
	1.0 (30)	1.0 (2)	1.4 (38)	1.5 (3)	2.5 (44)	2.4 (6)
	1.1 (0)	1.1 (0)	1.6 (23)	1.5 (18)	2.4 (17)	2.4 (26)
	1.1 (0)	1.1 (0)	1.6 (48)	1.4 (15)	2.6 (30)	2.2 (2)
	1.1 (0)	1.0 (27)	1.5 (37)	1.6 (8)	2.4 (14)	2.4 (8)
	1.1 (0)	1.0 (11)	1.4 (34)	1.5 (15)	2.9 (48)	2.3 (1)
	1.0 (5)	1.1 (0)	1.4 (11)	1.5 (28)	2.3 (35)	2.4 (18)
	1.0 (4)	1.1 (0)	1.5 (16)	1.6 (5)	2.3 (8)	2.4 (19)
	1.0 (42)	1.0 (6)	1.5 (9)	1.5 (6)	2.4 (49)	2.6 (4)
	1.0 (4)	1.0 (21)	1.4 (8)	1.5 (5)	2.4 (8)	2.4 (24)
	1.0 (12)	1.0 (3)	1.6 (25)	1.5 (6)	2.3 (9)	2.4 (26)
			2.3 (28)	1.5 (9)	2.3 (47)	2.6 (1)
			1.4 (30)	1.4 (19)	2.3 (25)	2.1 (16)
			1.5 (39)	1.7 (8)	2.3 (15)	2.2 (2)
			1.4 (49)	1.6 (8)	2.3 (30)	2.6 (0)
			1.4 (15)	1.4 (14)	2.2 (0)	2.2 (0)
			1.4 (49)	1.5 (16)	2.3 (13)	2.2 (14)
			1.5 (46)	1.5 (32)	2.3 (24)	2.4 (12)
			1.5 (0)	1.5 (0)	2.3 (10)	2.5 (17)
			1.4 (8)	1.5 (9)	2.3 (7)	2.4 (1)
			1.5 (17)	1.6 (0)	2.4 (36)	2.4 (11)
			1.7 (26)	2.0 (7)	2.1 (49)	2.2 (37)
			1.4 (6)	1.7 (0)	2.3 (38)	2.4 (9)
			1.4 (40)	1.5 (12)	2.3 (37)	2.4 (18)
			1.4 (7)	1.5 (1)	2.2 (34)	2.4 (9)
			1.6 (44)	1.6 (6)	2.2 (44)	2.4 (39)
			1.4 (20)	1.5 (33)	2.4 (31)	2.4 (21)
			1.4 (1)	1.5 (0)	2.3 (19)	2.5 (10)
			1.6 (13)	1.6 (10)	2.3 (16)	2.4 (19)
			1.5 (21)	1.5 (4)	2.4 (10)	2.6 (11)
			1.4 (14)	1.5 (8)	2.4 (15)	2.3 (20)
Mean (2 d.p.)	1.05	1.05	1.50	1.54	2.35	2.39
SD (3 d.p.)	0.052	0.052	0.172	0.114	0.139	0.133

Figure 6.3: Online and offline performance for the median trial of the GA and EP with the data file `corrupt20`.

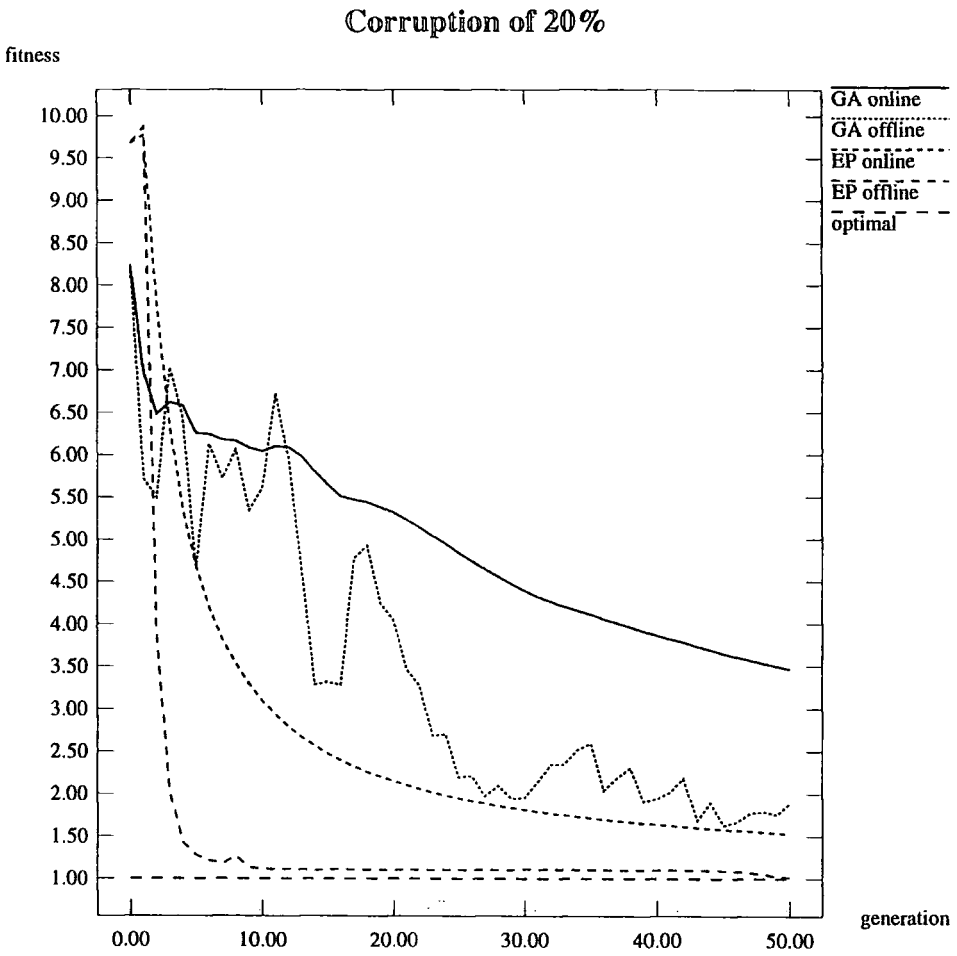


Figure 6.4: Online and offline performance for the median trial of the GA and EP with the data file `corrupt30`.

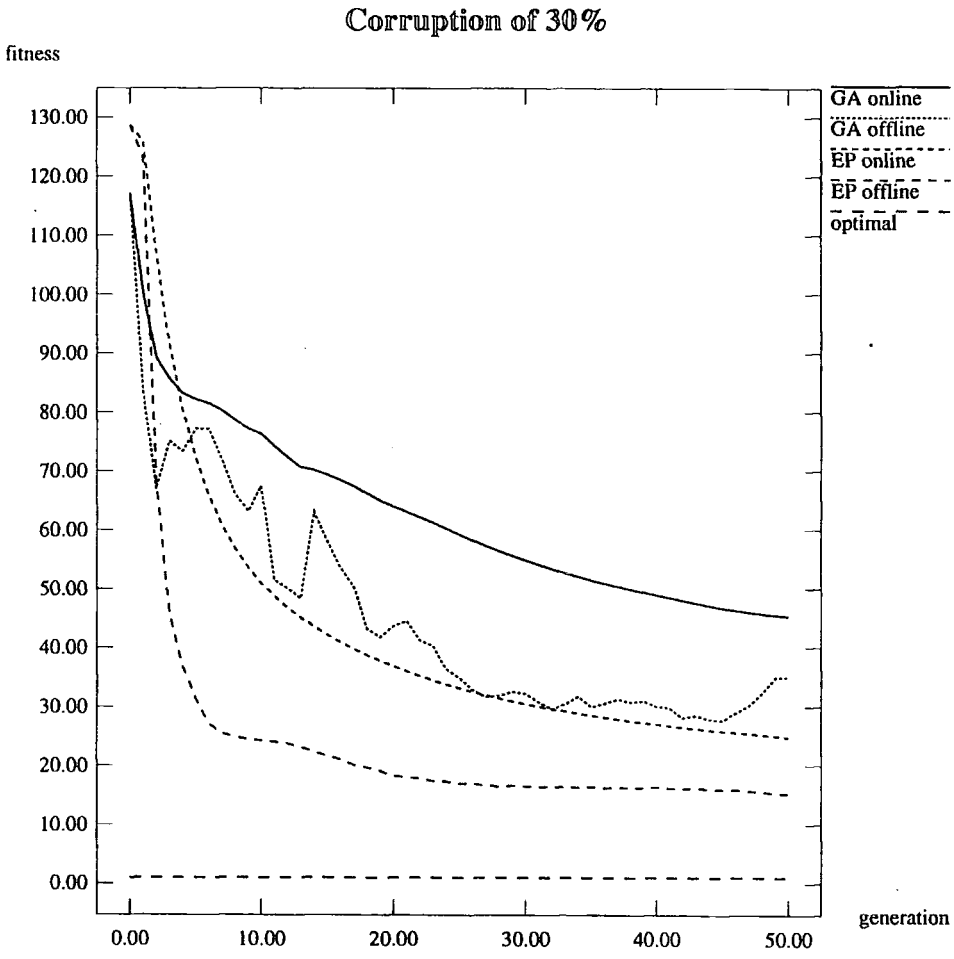
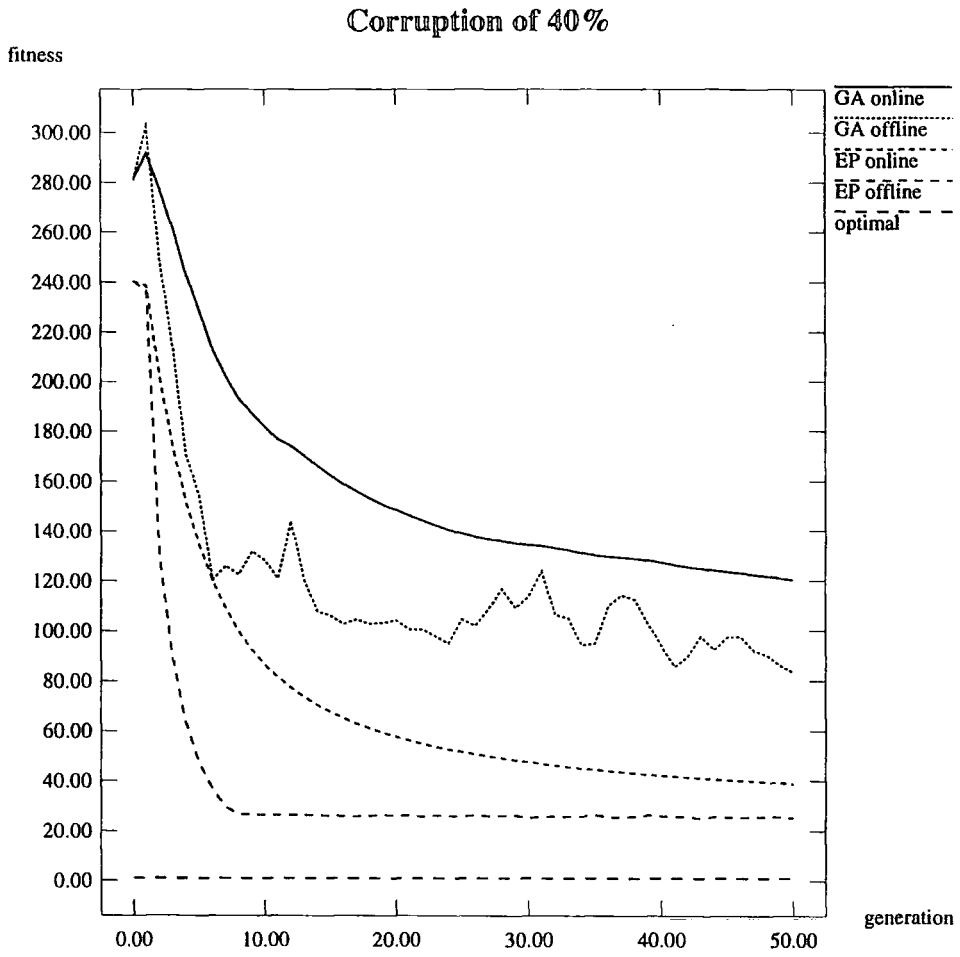


Figure 6.5: Online and offline performance for the median trial of the GA and EP with the data file `corrupt40`.



6.2.5 Dictionary

The dictionary chosen for this research was the machine-readable form of the Oxford Advanced Learner's Dictionary (OALD), described in section 3.4.11. The role of the dictionary is to provide, for each word in the system vocabulary, one or more pronunciations (in phoneme form) and one or more syntactic categories. For example:

computer	k@mpjut@r	K6%
control	k@ntr@Ul	H4%,M6%
course	kOs	J2$,M6*

The three fields are word, pronunciation (in phonemes) and syntactic categories. In this example, the syntactic categories K and M represent nouns, and H and J represent verbs; the %, $ and * characters indicate normal, rare and common frequency of occurrence.

On reflection, it would have been helpful if a more modern syntactic classification system had been used at the beginning of this research, such as that used by the SEC corpus and the CLAWS system. Ultimately, it was the convenience of having the relevant information provided by the same source that was the main selection criteria. A small number of syntactic codes, in addition to those provided by the OALD, were added to the system dictionary by hand. These covered possessive nouns and accusative/nominative pronouns.

6.3 Word Lattice Parsing

The aim of word lattice parsing (see Figure 6.6) is to produce the best sequence of words that spans (a portion of) a word lattice according some criteria. A simple method of parsing would be to start at a definite word boundary, for example either at the beginning of a sentence (or the end of a sentence) or when a pause in speech

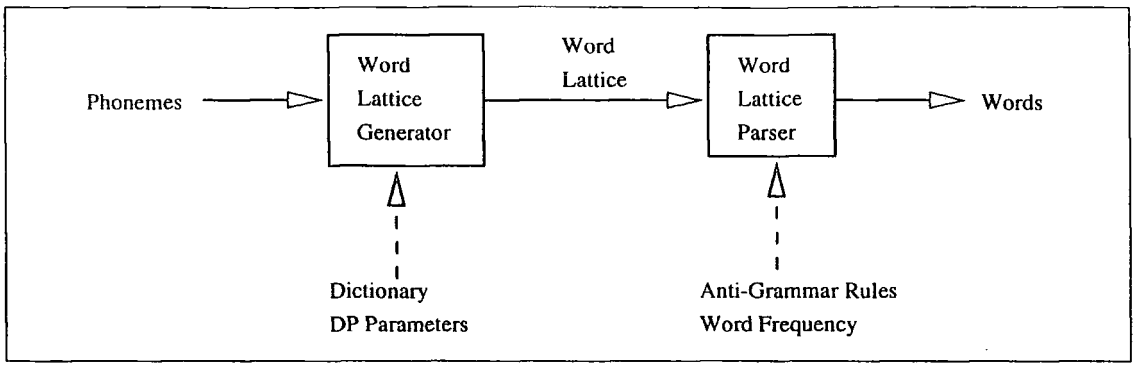


Figure 6.6: A Block Diagram of the AURAID System

occurs, and work forwards (or backwards) through the lattice, at each stage taking the best scoring word from the lattice that fits just after (or before) the current best word. Repeat this process until the end (or beginning) of the sentence is reached. This, essentially, is the method used, with the addition that contributing knowledge affects the choice of words from the lattice, so that it is not always the best scoring word that is selected.

6.3.1 Parse Initiation

The word lattice produced by the dynamic programming stage needs to be broken into chunks of manageable size for the parsing stage. Each chunk must finish at the end of a word. At certain points during the processing, ends of words can be identified, either by pauses in the speech, or by "common consent" of best words at different input phonemes, an example of this latter case is shown in Table 5.1. A word end must definitely exist at the "z" recognised phoneme because no word in the lattice that spans this phoneme does so in any position other than at the word end.

6.3.2 Sentence Hypotheses

During word lattice parsing, an ordered list of sentence hypotheses is maintained. A sentence hypothesis consists of

- a field indicating the frame at which this sentence hypothesis ends;
- a score indicating how good this sentence hypothesis is according to the criteria used;
- a list of words (and their associated information) that make up the sentence hypothesis.

The word information consists of

- the word string itself;
- the phoneme representation of the word;
- a field indicating the frame at which this word ends;
- a field indicating the frame at which this word starts;
- the local word score determined during word lattice generation;
- the broad syntactic class of the word;
- the OALD syntactic classification of the word;
- a pointer link to the next word in the list;

As the parse progresses, sentence hypotheses are added to and removed from the hypothesis list.

6.3.3 Beam Search

Initially, all possible starting words are added to the sentence hypothesis list. A number of these are then expanded as determined by their score and the score of the lowest scoring word. For the first three expansions of the sentence hypothesis list, all sentence hypotheses that are within 30% of the best scoring sentence hypothesis are expanded. In subsequent expansions, only those hypotheses that are within 20% of the best scoring sentence hypothesis are expanded. To keep the search space to a manageable level, any sentence hypothesis whose score is not within 50% of the best scoring sentence hypothesis is pruned.

This type of extended best-first search is know as a beam search. During the word lattice parse, the width of the beam is broader in the initial stages of the search, and narrower later on, taking on the shape of a pyramid.

A pure best-first search would expand only the best sentence hypothesis at each stage. This would be an acceptable approach if the phoneme error rate was very low as it is likely that the (partial) correct sentence would be expanded ahead of all other candidates. When the phoneme error rate is high, however, more sentence hypotheses need to be expanded at each stage during the search through the word lattice to give other sources of knowledge, such as syntax and semantics, a chance of recovering the poorly matching correct sentence. Under these circumstances, selecting only the best matching sentence would lead to a very poor level of word recognition.

The beam search also makes use of guesses about the score incurred by each sentence hypothesis over the remaining portion of the sentence being processed. These guesses are in the form of underestimates [Winston, 1992]. Each sentence hypothesis in the sentence hypothesis list is a recognition for part of the sentence being processed. The hypotheses all start at the same point but reach to different parts of the sentence being processed. An underestimate score is calculated for each sentence hypothesis based on the remaining amount of the sentence being processed multiplied by some constant determined empirically.

A simple extension of the search thus described would produce a search technique known as A*. The A* search is a best-first (or beam) search that makes use of underestimates of distance remaining as described above, but also discards redundant paths. In other words, if several paths reach the same node in the search, only the best scoring of these paths is kept alive, the others being removed from the search space. The knowledge (described in the next section) that we use to aid the word lattice parse can result in sentence hypotheses being given a bonus or a penalty. This is consistent with [Paul, 1992], who states that the A* is only suitable for word lattice parsing (stack decoding) when a no-grammar or unigram language model is used.

For example, given two sentence hypotheses P, consisting of words p_1, p_2, p_3, and Q, consisting of words q_1, q_2, both hypotheses stretching to node n in the word lattice and with scores 20 and 25 respectively. Using the A* algorithm, we would prune Q because it has a worse score than P — we only keep the best sentence hypothesis that reaches a particular node. If we were to extend P by one more node by adding word w to span the phonemes between node n and node $n+1$, P may now have a score of 30 at node $n+1$. If we had kept Q in list of sentence hypotheses it may have a lower score even though it would have been extended by the same word, w, because of the grammar penalties (or bonuses) incurred for the new hypotheses P, consisting of words p_1, p_2, p_3, w, and Q, consisting of words q_1, q_2, w.

A short-circuit condition is built into the beam search that we use, in order that a particularly unfruitful parse of a portion of a word lattice may be aborted. This is activated when the parse repeatedly fails to extend the best sentence hypotheses beyond a particular point in the word lattice. The search is aborted and the current best sentence hypothesis is displayed. This short-circuit condition is essential to avoid the possibility of long delays during recognition.

During our earlier studies [Collingham and Garigliano, 1992] [Collingham and Garigliano, 1993], the word lattice parsing algorithm could handle phonemes that had been inserted into and deleted from the input by allowing a particular sentence

hypothesis to ignore ("skip") a phoneme, or by allowing a phoneme to be "shared" by two different words (co-articulation). This would incur a small penalty. In the first example below, a phoneme has been incorrectly inserted between the two words, and in the second example a phoneme has been incorrectly deleted between the two words.

```
just          to
dZ V s t   k   t @

just   to
dZ V s t @
```

In the first example, the word lattice would contain the word `just` spanning the first set of phonemes, and the word `to` spanning the last set of phonemes, and probably a word like `stick` spanning the "`s t k`" phonemes. This was handled by allowing the parsing algorithm to skip over the inserted phoneme. In the second example, the word lattice would contain the word `just` spanning the "`dZ V s t`" phonemes, and the word `to` spanning the "`t @`" phonemes. This was handled in AURAID by allowing the parsing algorithm to parse the "`t`" phoneme twice, enabling both words to span it.

However, a recent analysis of the performance of the individual components of the word lattice parsing algorithm have shown that the performance gain is negligible compared to the great cost of incorporating the skip and share algorithm [Johnson *et al.*, 1994b]. The skip algorithm was successful in one aspect in particular and that was in allowing the word lattice parse to skip over part word disfluencies and filled pauses, for example

...the qu the answer ...

...the er first thing ...

However, this still did not make the skip algorithm worth retaining. Further work is being undertaken on extending the skip algorithm and in other areas of

speech repair [Garigliano *et al.*, 1993b] [Johnson *et al.*, 1994a] [Johnson *et al.*, 1994c].

6.3.4 Contributing Knowledge

Several sources of knowledge may be incorporated into the word lattice parsing stage. Two that have been successfully implemented are the *anti-grammar rules* (dealt with in the next section), and the *word frequency* information.

Word Frequency

A portion of the dictionary used in this research, extracted from the OALD, is shown on page 105. The final column contains word frequency information. In the OALD, word frequencies are divided into three classes: common (about 200 different words), normal (the vast majority of words) and rare (a few "hand-selected" words). The frequencies (or rarity codes) are attached to tags rather than to words because a word can be common in one use and rare in another. For example, "course" is common as a noun and rare as a verb.

The phonemic match score of a word, determined during the word lattice generation stage, is decreased if the associated frequency is common, and increased if the frequency is rare. This does improve the recognition performance of the system [Johnson *et al.*, 1994b]. The OALD rarity tags are very broad; it is believed that introducing more, accurate, classes would substantially benefit recognition performance. This has to be done with care, because the more rarity codes that are introduced, the more domain dependent the system becomes.

6.4 Anti-Grammar

6.4.1 Introduction

Many speech recognition systems restrict what may be spoken by use of a grammar (or statistical language model). Spontaneous speech, in other words naturally spoken English, is very rarely completely grammatically correct, however an analysis of the data we have collected shows that people do *not* speak in a completely ungrammatical way [Garigliano *et al.*, 1993b], and that speech is not necessarily broken into distinct sentences but more often than not multiple sentences without pauses, or partial sentences (or individual words and part words) that link pieces of speech together. A further problem within pieces of speech is that of repair, in other words the correction of previously spoken words. This leads us to the conclusion that it is not possible to define a complete grammar for spontaneous speech in the same way that a grammar is used for written (and "clean" spoken) English, or in the same way that a statistical language model is trained for recognition using read speech.

Instead, we have taken the opposite approach by developing a set of rules that can be used to check the syntactic *incorrectness* of sequences of words. We call these syntactic rules an "anti-grammar" as most of the rules are used to penalise certain syntactic constructs rather than identify syntactically correct sequences. This inverted method of modelling follows naturally from the fact that it makes sense to keep the size of the model to a minimum for efficiency reasons. For a constrained task it is efficient to model the few legal sentences, but once the balance changes, so that there are more legal sentences than illegal ones, it is more efficient to model the smaller set of illegal sentences.

The anti-grammar is really an extreme case. It could be possible to have varying degrees of penalties and bonuses so that some grammatical constructs are heavily penalises indicating that they never occur, some are given smaller penalties indicating a certain amount of rarity, some are given no penalty indicating a neutral

occurrence, some are given a small bonus indicating a certain amount of commonness, and some may be given a large bonus indicating that they are very common constructs. It should be possible to derive these penalties from a corpus, given the future availability of a large spontaneous speech corpus labelled with the appropriate grammatical categories.

It is appropriate to mention several other unconventional grammatical constraint models which are similar to the anti-grammar. The TAGGIT program was used to tag the Brown corpus [Marshall, 1987]. It made use of both positive and negative context frame rules for word tag disambiguation. The constraint grammar and formalism and tagger/parser developed at Helsinki University also uses both positive and negative rules or constraints, involving words and tags (and their combinations), to eliminate incompatible candidate analyses [Karlsson *et al.*, 1995]. Karlsson also details some of the advantages of hand-crafted constraints over purely probabilistic tagging systems.

6.4.2 Details

Currently, the anti-grammar is made up of four parts.

- 116 simple rules concerning sequences of particular syntactic categories, for example:

  ```
  ADJECTIVE ARTICLE ADJECTIVE
  ```

- more complicated rules concerning sequences of syntactic categories in addition to particular forms of words, for example:

  ```
  ARTICLE VERB(not 'ing' form)"
  ```

- rules concerning words that behave in a strange manner, for example, `not` and `very`

- common constructs of spoken English are given an advantage, for example,

```
to VERB
very ADJ
```

The full annotated list of rules can be found in appendix A.

Initially the word lattice parsing unit was developed without the aid of any contributing knowledge. It became clear that the word sequences that were selected by the parser were the closest match phonemically to the corrupted phoneme input. Many of the word sequences were, however, ungrammatical. Initially, the anti-grammar rules were developed in an *ad hoc* fashion in response to ungrammatical output from the word lattice parser. It soon became clear that this would not provide a complete enough solution and would certainly take some time to develop and test. Subsequent development of the anti-grammar occurred in three stages. Firstly, rules were constructed using the author's knowledge of the English language, taking into account the vagaries of spontaneous speech. This and the ad hoc approach yielded approximately 70 rules.

The second development phase involved semi-automatically tagging two of the lectures from the Durham lecture corpus (see section 3.4.6). This involved the use of the Xerox public domain part of speech tagging program [Cutting *et al.*, 1992]. The data was firstly tagged by the program, then any words that were clearly tagged incorrectly were amended by hand. A second tagging, using the program, then took these hand tagged words into account and produced much more accurately tagged data. N-gram frequency counts were then calculated for successive part of speech tag sequences (of length 1–4 tags); rarely occuring sequences were removed. This list was then "inverted" to determine which tag sequences did *not* occur in the data, and normalised to remove any duplication (for example, only a 2-tag sequence that occured within a 4-tag sequence would be retained). This resulted in a more complete list of anti-grammar rules, or constructs that do not occur much in everyday spoken English. This list was checked against the hand-built rules, resulting in an updated list of approximately 120 anti-grammar rules. During this process, note was taken of particular tag sequences that *did* occur very frequently in the data, these were added to the anti-grammar as rules that gave a bonus

score to a sentence hypothesis containing the sequence, rather than the more usual penalty.

The third development phase involved a revision of the anti-grammar rules containing verbs and adverbs. It was felt that the existing rules were too general, and needed restricting by taking into account more information on the particular verbs and adverbs being used (such as modal verbs). The rules were therefore adjusted using more detailed grammatical information on verb constructs [Hardie, 1992] [Sinclair, 1990].

6.4.3 Analysis

The questions that needs to be asked are: which knowledge source actually benefits the system and how do the different knowledge sources cooperate to achieve the overall goal of the system? An investigation of each knowledge source used by the system has been undertaken. Each knowledge source was investigated to identify the advantages and disadvantages of using it and the effect on the system's overall performance when the knowledge source is used. The original word lattice parsing system was modified so that several versions of the system could be easily created. Each version processed ten test sentences and measurements were taken on the search space generated during each run and performance of each system. This allows the best system to be identified and the bottom line performance of each individual knowledge source, both alone and in co-operation with other knowledge sources, to be identified. The aim of the analysis was to decrease the search space of the system, thus increasing the time performance, without diminishing the accuracy of the system.

The data used in the analysis presented in this section was taken from a single lecture on software engineering given as the first introductory lecture for second year computer science students. The lecture contained 4903 words and 382 sentences, or part sentences, with an average of 12.84 words per-sentence. From this lecture ten representative sentences were selected. They were representative in sentence length

Switch	*System*				
	1	2	3	4	5
Skip/Share	x		x	x	x
Word Frequency		x	x	x	x
Anti-Grammar				x	x
500 Words	x	x	x	x	
2000 Words					x

Table 6.4: System Configuration During Knowledge Source Analysis

and speech disfluencies: five of the sentences contained repairs. Two dictionaries were used in the analysis. The first contained 528 words of which 354 were from the LOB corpus and 146 were category words which are important to the general field of lectures. The second contained 1985 words and was used to test the system's performance on a more realistically-sized vocabulary.

The switches that were built into the original system to allow different versions to be easily created were:

- Skip and share algorithm
- Word frequency information
- Anti-grammar rules
- 500 word dictionary
- 2000 word dictionary

The combinations of the switches which made up the six systems can be seen in Table 6.4. The first three knowledge sources are described elsewhere in this chapter.

The data collected on each run included: system time; elapsed times; position of the right hypothesis (RH) in the hypothesis list (HL); hypothesis score of the RH; error rate; word accuracy; words correct; the percentage position of the RH from the top of the HL (for example, 10% from the top of the list) and the percentage difference between the score of the top hypothesis and the score of the RH.

It must be noted that in identifying the position of the RH (the actual input) within the HL the type (VERB, etc.) of the word was used as well as the word itself along with the exact phoneme location. This makes the details very accurate and the figures seem lower than those systems whose performance is measured on word identification alone.

The results were compared to see if the included knowledge source had any effect and whether the effect, in combination with other knowledge sources, was beneficial to the system as a whole.

Results

The introduction of skip and share processing made very little difference to the overall performance of the system. It did not increase the systems performance though it did show promise in overcoming one of the problems of repair by bridging a part word, but the resulting string had such a low score that it was never expanded.

Word frequency information gave a definite increase in system performance for both system time and position of the RH. Though not producing a completely satisfactory result it did go some way towards moving the RH to the top of the hypothesis list.

A combination of skip and share processing and word frequency information showed a slight increase in performance over the word frequency information alone but this was mainly a time increase rather than a performance increase. This system, with a combination of skip and share processing and word frequency information, was taken as the basis for the rest of the analysis.

The systems using anti-grammar rules worked much better than the other systems as the RH was generally higher in the hypothesis list, though it was not necessarily top of the list. This is not a major problem to this work as the accuracy of the measurements are such that a higher position in the list is more

Sentence	*System*				
	1	2	3	4	5
1	69.2	53.8	53.8	61.4	46.1
2	46.1	61.5	61.5	53.8	53.8
3	46.1	46.1	53.8	46.1	46.1
4	66.6	58.3	58.3	66.6	50.0
5	75.0	75.0	75.0	83.3	83.3
6	58.3	50.0	50.0	75.0	50.0
7	100	91.6	91.6	100	91.6
8	54.5	72.7	63.6	72.7	54.5
9	63.6	54.5	54.5	54.5	54.5
10	70.0	70.0	70.0	80.0	80.0

Table 6.5: Percentage Word Accuracy Obtained for each System During Knowledge Source Analysis

desirable as it shows that the extra knowledge source is, in fact, being beneficial.

The performance of the system using a more realistic vocabulary of 1985 words was acceptable. The systems performance did not decrease as would be expected but increased for all sentences. The changes in system time fluctuated across the test sentences (some increased and some decreased) but the position of the RH in the list of hypotheses generally increased.

System Performance

Generally the system showed an increased performance both in system time and position of the RH when knowledge sources were combined. As well as this information word accuracy for each system was also calculated. This measure of accuracy was not deemed as important as the HL measurements as this research was interested in the progress of the RH when knowledge was added to the system. Table 6.5 shows the word accuracy for each of the systems that were tested. Further information on this analysis will be contained in [Johnson, to appear 1995]; it consists of several hundred pages of data, and is available for inspection from the authors.

6.5 Software Engineering Aspects of the Test-Bench

6.5.1 The Word Lattice Generator

The word lattice generator uses the SAM-PA machine readable phoneme representation. Should this underlying representation need to be changed, it will have little impact on the generator, nor on the parameter optimiser. Each only requires the knowledge of how the phonemes are grouped together into classes.

6.5.2 The Word Lattice Parser

The word lattice produced by the word lattice generator can be viewed as a starting point for linguistic constraint researchers so that syntactic and semantic constraint models (and others) may be researched without the need for speech recognition hardware. The basic word lattice parsing algorithm has been designed to incorporate different types of knowledge in a modular fashion.

The algorithm used during word lattice parsing is as follows.

1. Construct initial list of sentence hypotheses from the words that start at the first phoneme.
2. WHILE we haven't reached the goal
   (a) Extend each sentence hypothesis by each of its possible successor words obtained from the word lattice to create a list of new sentence hypotheses.
   (b) Score each new sentence hypothesis for each knowledge source.
   (c) Sum the individual knowledge source scores for each new sentence hypothesis and add it to the sum of the (acoustic match) word scores; this

total score becomes the sentence hypothesis score.

(d) Add the new sentence hypotheses to the old sentence hypothesis list.

(e) Sort according to the sentence hypothesis score.

(f) Prune any high scoring sentence hypotheses from the list.

3. ENDWHILE

4. Return the best scoring sentence hypothesis.

Step 2(a) in this algorithm demonstrates the modular interface: each new sentence hypothesis is passed to each knowledge source available in the system; the knowledge source simply returns a score. Currently, the individual scores for each knowledge source are summed with equal weighting. One drawback of this approach is that it doesn't allow manipulation of the sentence hypothesis list. One situation where this might be required is in the detection and correction of repair: the repair knowledge source may identify a possible repair in a sentence hypothesis and may then want to add a corrected sentence hypothesis onto the sentence hypothesis list.

The type (in C) of each knowledge source is therefore of the form

```
float knowledge_source (sentence_hypothesis sh)
```

although in practice, each knowledge source may require additional information such as the value of the goal, or the words of the previously recognised sentence for context.

The anti-grammar knowledge source makes use of the syntactic category information present in the dictionary. Should this underlying representation need to be changed, it will have little impact on the knowledge source because the syntactic information is not embedded in the anti-grammar rules but has been abstracted into a series of predicates — functions that take a particular syntactic category as a parameter and return a boolean result. For example, the function (in C) to check whether or not a word is in the third person singular would be:

```
int is_3rd_pers_sing(wr)
word_rec *wr;
{
   if (wr == NULL)
      return (1);
   if (wr->c2 == 'a')
      return (0);
   else
      return (1);
}
```

Once these low-level predicates have been altered to take into account any new syntactic category representation, the anti-grammar knowledge source will require no more alterations.

Chapter 7

Evaluation Framework

This chapter outlines the framework in which the work described in this thesis is evaluated. Addressing in particular phoneme recognition assessment, word lattice quality, the suitability of the anti-grammar, word recognition assessment and readability issues. The problem of evaluating recognition of spontaneous speech is also discussed, and a case for developing a new measure for assessing speech recognisers that handle spontaneous speech is presented. A brief mention is made of the early work in this area.

7.1 Phoneme Recognition Assessment

In order to put the word recognition rates of an automatic speech recognition system into context, it is important to know the phoneme recognition rate of the underlying acoustic-phonetic recogniser. In other words, all things being equal (such as language model, vocabulary size and the like), it is less impressive to achieve 80% word recognition given, say, 100% phoneme recognition than, say, 50% phoneme recognition.

Methods of evaluating phoneme recognition accuracy are similar to those for

determining word recognition accuracy, with the addition that group figures are also calculated. For the purposes of this research, phonemes have been grouped by manner of articulation, see Table 6.2. It is also useful to construct confusion matrices to show substitution errors. This is easier to do at the phoneme level than at the word level because of the limited number of phonemes, which is independent of vocabulary size.

Caveat — determining a phonemic transcription for a portion of speech by hand is a non-trivial task, especially when it comes to labelling vowel sounds. For this reason, when phoneme accuracy is measured, a machine-generated phoneme transcription is compared against a standard pronunciation dictionary-generated transcription.

The phoneme recognition assessment described in this section is guided by the phonetic analysis performed in [Browning *et al.*, 1990]. We use the following definitions:

$$\begin{aligned}
\textit{number of phonemes in correct transcription} &= p \\
\textit{number of phoneme substitution errors} &= s \\
\textit{number of phoneme deletion errors} &= d \\
\textit{number of phoneme insertion errors} &= i \\
\%\ \textit{phoneme substitution errors} &= 100 \cdot \frac{s}{p} && (7.1) \\
\%\ \textit{phoneme deletion errors} &= 100 \cdot \frac{d}{p} && (7.2) \\
\%\ \textit{phoneme insertion errors} &= 100 \cdot \frac{i}{p} && (7.3) \\
\%\ \textit{phoneme error} &= 100 \cdot \frac{s+d+i}{p} && (7.4) \\
\%\ \textit{phonemes correct} &= 100 \cdot \frac{p-(s+d)}{p} && (7.5) \\
\%\ \textit{phoneme accuracy} &= 100 \cdot \frac{p-(s+d+i)}{p} && (7.6)
\end{aligned}$$

Equivalent figures may be obtained for phoneme groups by first calculating p,

s, d and i for each group.

7.2 Word Lattice Quality

The quality of a word lattice may be evaluated by determining the positions within the lattice of the actual spoken words. We define the *average word rank* for a word lattice to be

$$average\ word\ rank\ =\ \frac{1}{n}\sum_{i=1}^{n} r_i \tag{7.7}$$

where n is the number of spoken words and r_i is the rank in the word lattice of the i'th spoken word at its correct start position. Within a lattice, equal scoring word hypotheses are given the same rank. The aim of any word lattice generation algorithm is to get this measure as near to 1 as possible.

Referring to the word lattice shown in Table 6.1, for the sentence fragment "the word", with "the" spanning frames 1–2, and "word" spanning frames 3–5. The rank of "the" at its correct start frame (1) is 1, and the rank of "word" at its correct start frame (3) is 1. The average word rank for this lattice is therefore 1.

Average word rank was used as the measure of fitness during the parameter estimation described in section 6.2.4.

7.3 Suitability of the Anti-Grammar

7.3.1 Perplexity

Perplexity is a measure of the constraint imposed by a grammar or language model, and is often called the average word branching factor, in other words the average number of alternative words at each point in the recognition. Perplexity is described

in more detail in section 3.2.5. Perplexity is defined as

$$perplexity = P(w_1, w_2, \ldots, w_n)^{-\frac{1}{n}} \tag{7.8}$$

where $P(w_1, w_2, \ldots, w_n)$ is the probability of occurrence of a sentence containing the words $w_1, w_2, \ldots, w_n$, given the language model or grammar of the speech recognition system. For statistical language models, such as trigram, which are generated from a large corpus of data, this probability is straightforward to compute. For the anti-grammar used in this research, the probability of a sentence occuring has to be estimated empirically, as follows.

1. Randomly generate q (q is large) sentences of length n words, using a vocabulary of size v;
2. Test each sentence for inclusion in the language covered by the language model, giving l legal sentences ($l \leq q$);
3. The maximum number of possible sentences is given by v^n;
4. The approximate number of legal sentences allowed by the language model is

$$approximate\ number\ of\ legal\ sentences = \frac{l}{q} \cdot v^n$$

5. The probability of a given legal sentence is therefore given by

$$P(w_1, w_2, \ldots, w_n) = \frac{1}{\frac{l}{q} \cdot v^n}$$

$$= \frac{q}{l} \cdot v^{-n}$$

6. The perplexity of the language model is therefore

$$perplexity = \left(\frac{q}{l} \cdot v^{-n}\right)^{\frac{-1}{n}}$$

$$= v \cdot \left(\frac{q}{l}\right)^{\frac{-1}{n}}$$

$$= v \cdot \left(\frac{l}{q}\right)^{\frac{1}{n}} \tag{7.9}$$

It must be pointed out that this section is concerned with measuring the perplexity of the language model and not of the task. In cases where a language model is determined statistically from a large corpus of data for a given task, the perplexity of the language model and the perplexity of the task are identical. The work presented in this thesis is not tied to any specific task and so no large collection of data is available for any of the domains used for evaluation. It is expected that the perplexity of the domains used for evaluation would be much lower than the perplexity of the language model.

It is useful to calculate the perplexity of the language model to determine the amount of restriction imposed upon a speaker, compared to the word recognition rate that is achieved.

7.3.2 Coverage

A useful measurement for evaluating the generality of the anti-grammar is to see what proportion of correct (or actually spoken, but *not* correct) sentences are rejected by the anti-grammar. This can be measured by checking whether or not sentences (tagged with their parts of speech) violate any of the anti-grammar rules. Several corpora exist that contain tagged data, however, the data tends to be text based (written English), or transcripts of read speech. There is very little tagged data in existence for spontaneous speech.

7.4 Word Recognition Assessment

Word recognition measurements were introduced in 3.2.6. The definitions given in this section are suitable measures for read speech. Their use for measuring

recognition performance on spontaneous speech is not so clear. They are presented here for completeness, and will be used for assessment purposes on the assumption that it is "good" to obtain a high percentage of words correct and a high word accuracy. The last section in this chapter, section 7.6, discusses this in more detail and presents the case for developing a new measure for assessing speech recognisers that handle spontaneous speech.

We use the following definitions for word recognition assessment:

$$
\begin{aligned}
\textit{number of words in correct transcription} &= w \\
\textit{number of word substitution errors} &= s \\
\textit{number of word deletion errors} &= d \\
\textit{number of word insertion errors} &= i \\
\%\ \textit{word substitution errors} &= 100 \cdot \frac{s}{w} && (7.10) \\
\%\ \textit{word deletion errors} &= 100 \cdot \frac{d}{w} && (7.11) \\
\%\ \textit{word insertion errors} &= 100 \cdot \frac{i}{w} && (7.12) \\
\%\ \textit{word error} &= 100 \cdot \frac{s+d+i}{w} && (7.13) \\
\%\ \textit{words correct} &= 100 \cdot \frac{w-(s+d)}{w} && (7.14) \\
\%\ \textit{word accuracy} &= 100 \cdot \frac{w-(s+d+i)}{w} && (7.15)
\end{aligned}
$$

7.5 Readability

A measure that must be taken into account is the readability of the speech recognition output. This cannot be measured simply in terms of the number of correctly recognised words, because spontaneous speech contains many disfluencies.

Many methods have been developed for measuring the readability of text. In most cases, the objective has been to grade texts for teaching purposes in schools. Many of these methods are based upon a statistical analysis of samples of text, de-

termining for example, the number of words per sentence, the number of syllables or letters per word, or the number of words containing more than two syllables. These statistics are then combined in some form to give a reading age corresponding to the difficulty of the text. These measures have received much criticism because there is no neat correlation between sentence or word length and reading difficulty. Nor are they applicable for measuring the readability of the output of a speech recognition system, for two reasons, firstly because the lack of punctuation in spontaneous speech makes the measures incalculable, and secondly, errors of recognition such as word insertion and deletion, may confuse the formulae giving inaccurate results. Instead a direct test that assesses how easily someone can read and comprehend a text is required.

For many years, comprehension tests have been used in schools to measure reading ability. A comprehension test consists of reading a passage of text and then answering questions on it. The disadvantages of comprehension tests are that they are difficult to construct, lengthy to administer, and the questions are often answerable using a person's prior knowledge rather than knowledge gained from reading a particular passage. They also do not test comprehension of the complete passage, since the questions that are asked cover only a small subset of the text.

A test known as the Cloze procedure was developed in 1953 to measure reading comprehension [Taylor, 1953]. Instead of preparing a passage with associated comprehension questions, every n'th word is removed from the passage, where n is typically five. The number of correct words guessed by a reader is then used as a measure of his or her understanding. An example of a Cloze passage (with answers) is given in Figure 7.1.

For measuring readability, the most accepted way to form a Cloze passage is to select one or more paragraphs that total 250–300 words, then with the exception of the first few sentences which remain unaltered, remove every fifth word [Bormuth, 1966]. Each removed word is replaced by a gap or ruled line of uniform length. Much debate has surrounded the applicability of the Cloze procedure for measuring reading ability, however its use for measuring the readability of a passage is widely

A car bomb exploded outside the Cabinet Office in Whitehall last night, 100 yards (1) _____ 10 Downing Street. Nobody (2) _____ injured in the explosion (3) _____ happened just after 9pm (4) _____ the corner of Downing (5) _____ and Whitehall. Police evacuated (6) _____ area. First reports suggested (7) _____ the bomb went off (8) _____ a black taxi after (9) _____ driver had been forced (10) _____ drive to Whitehall. The (11) _____ was later reported to (12) _____ burning fiercely.

Answers:

(1) from	(2) was	(3) which
(4) on	(5) Street	(6) the
(7) that	(8) in	(9) the
(10) to	(11) taxi	(12) be

Figure 7.1: An Example of a Cloze Passage

accepted. Even so, research has continued to evaluate the Cloze procedure as a suitable measure of readability. For example, removing every fifth word in a passage is quite mechanical, other studies have examined different values of n, and the balancing of the types (nouns, adjectives and the like) of words that are removed (known as "lexical Cloze"). Examination of the scoring process of the Cloze procedure has also taken place. In its purest form, answers are either right or wrong, in a more sophisticated form, a semantic score is used, with answers that have a similar meaning to the removed word scoring, say, half a mark. For the assessment of readability, all of this analysis has not led to any enhancements to the Cloze procedure that offer significantly improved results [Robinson, 1981], hence the simplest form of the Cloze procedure is used in this analysis.

7.6 Measurement of Meaning

In our heads we all have a notion of how words and concepts relate to each other. Each of us makes use of this information when we listen to someone speak or read a page in a book. In a pub, for example, there is often a lot of background noise,

either chat or music, yet when someone speaks to us we can tell what they have said despite not hearing each and every word that they spoke. We can use our knowledge of grammar to determine the kinds of words that fill the gaps in our guess as to what was actually spoken, we can use our knowledge of semantics to know what would make sense based on the context of previously spoken sentences. Indeed we could get the gist of a conversation by hearing only a few key words.

As discussed in section 3.2.6, the existing metric that is used to assess speech recognisers is to simply count the number of words correctly recognised. No account is taken of the importance of the words that are incorrectly recognised, nor the understandability of the resulting output of the recogniser. Research is being undertaken into a new metric provided by semantic distance to assess the meaning content of a text. To provide this measurement the algorithm does not rely on statistical means, such as counting word co-occurences. Instead a deep representation of meaning is used; semantic distance is derived from the structure of the data in this representation.

There are at least five key areas in which research into a measure of semantic distance will make an impact:

- assessment of speech recognition systems;
- use of domain knowledge to aid speech recognisers;
- summarisation and content scanning of text;
- topic spotting;
- assessment of machine translation.

No suitable metric for measuring semantic distance exists in any of these fields. The first area is discussed below as it is concerned with speech recognition evaluation, the other areas are briefly mentioned in section 9.2.

Assessment of Speech Recognition Systems

The existing metric that is used to assess the performance of automatic speech recognition systems is to simply count the number of words correctly recognised. No account is taken of the importance of the words that are incorrectly recognised, nor the understandability of the resulting output of the recogniser. Research is being undertaken to develop a metric that takes this into account, and would, for example, be able to say that a spoken text is recognised with 75% words correct and 85% of the original meaning [Short *et al.*, 1994b]. Although this work has not progressed far enough to be included in this thesis, future research by the author will develop this metric further.

This may appear to be an irrelevant measure to develop because the ultimate goal of automatic speech recognition is to achieve 100% word recognition. This may be true of "clean" speech that contains no errors, such as read speech, but is not true for natural spontaneous speech which contains many filled pauses, part words and sentence repair. For example, given the spoken input:

I err want the err ti time of the err first tr no the last train to err Newcastle

we would prefer our speech recogniser to come up with something like:

I want the time of the last train to Newcastle

which could be said to have a word accuracy of 50% and a meaning measure of 100% compared to the original spoken input. A measure of semantic distance should assist in the determination of which "errors" in the recognition are unimportant.

Chapter 8

Results

This chapter gives details of the data that was used for evaluation purposes and presents results for the areas outlined in the previous chapter.

8.1 Data Preparation

Two small sets of data were used for word lattice generation parameter estimation. These data sets were portions of (accurately) transcribed lectures given to undergraduate students in computer science at the University of Durham. The data sets consisted of 113 and 112 words respectively and are shown below. The sequence of characters <.> indicates a pause.

Parameter Estimation Data: evoldata1

for this lecture we're going to be looking at <.> <.> maintenance models what we're going to do is <.> <.> is to be looking at <.> <.> this in a historical context looking back in the literature and find out what various people think software maintenance is about and how they model the process <.> <.> it's quite useful to find this out to give us some sort of view on why certain ideas in maintenance have grown up <.> <.> so what this

lecture is is a series of models devised by various people and then what we're going to do in the next lecture is take one of those models apart and look into it in a lot more detail

Parameter Estimation Data: evoldata2

now <.> <.> the first <.> <.> thing to tell you <.> <.> is the book the recommended book for this course <.> <.> is that <.> <.> software engineering the third edition don't get the first or the third however cheap it is it's awe they're awful <.> <.> it's this book here <.> <.> it's eighteen or nineteen pounds <.> <.> but you've all got plenty of money so you can all afford it <.> <.> what i <.> <.> try to do on the course is that i don't <.> <.> exactly follow what's in that book you should see this book as supplementary reading i assume that you're reading the relevant sections and occasionally i will point out the chapter you should read that i don't have time to cover

A previously unseen lecture taken from the Lund corpus was used for evaluation: text number 12.6, described as a "popular lecture" and given by a male builder in 1972. The lecture did not contain any indication of pauses, so these were added by hand. The lecture was converted to phoneme form using pronunciations from the OALD. The lecture consisted of 5057 words, an extract is given below.

Evaluation Data: lunddata

well rather than give a talk about the history of stoke poges <.> <.> i felt it might be a little more interesting to you all <.> <.> to hear about my own life <.> <.> lived and growing up in this wonderful village of stoke poges <.> <.> i attended stoke school and <.> <.> i must say <.> <.> i was taught very thoroughly the three rs <.> <.> funnily enough my father went to the same school <.> <.> and he was one of the first pupils <.> <.> before that <.> <.> he used to go to the school next door to here <.> <.> and pay a penny a week <.> <.> along with all the other village boys for his education <.> <.> considering his schooling must have stopped at about fourteen years <.> <.> his beautiful copperplate writing and his reading with understanding was really remarkable <.> <.> i lived in my early life in wexham street <.> <.> it was a semi detached house <.> <.> built by my father and uncle with their own hands <.> <.> and we lived we were a family of five <.> <.> there were three children <.> <.> two sisters and myself <.> <.> we had a very big garden <.> <.> and we used to have to produce the produce from the garden <.> <.> the potatoes <.>

<.> the root crops <.> <.> store them keep them for the use of us during the whole of the winter <.> <.> we also kept <.> <.> as everyone in the village at that time kept chicken <.> <.> we kept a goat <.> <.> rabbits and occasionally we used to keep a pig <.> <.> the chickens was looked after by my sisters

To the existing dictionary of 1984 words were added 653 new words to give a system dictionary of 2637 words for the `lunddata` evaluation.

A collection of 113 previously unseen sentences taken from the Wall Street Journal corpus were used for a second evaluation. These sentences were used for the 1993 ARPA CSR evaluations. The sentences did not contain any indication of pauses, and *none* were added. This data set contains no disfluencies and is included to demonstrate the recognition ability of the system on read data. The sentences were converted to phoneme form using pronunciations from the OALD. The sentences consisted of 1923 words, an extract is given below.

Evaluation Data: wsjdata

bell canada enterprises incorporated said it plans an offering in europe of one hundred and fifty million dollars canadian of notes.

the five year ten percent notes were priced at one oh one.

lead underwriter is union bank of switzerland securities limited proceeds will be used to refinance short term debt.

bell canada enterprises is a telecommunications energy printing and real estate concern.

not surprisingly the davis zweig report has become more bearish dropping to a twenty five percent bond position around mid april.

yesterday it called for a complete move out of bonds and into money market funds.

meanwhile the bond market rallied sharply for the day.

it also would bar foreign companies from becoming primary dealers in US

government securities unless their governments give US companies the same right in their countries.

it is aimed at japan.

the federal reserve board recently accepted two japanese firms as primary dealers.

To the existing dictionary of 1984 words were added 404 new words to give a system dictionary of 2388 words for the `wsjdata` evaluation.

8.2 Phoneme Recognition Assessment

In the absence of a suitable front-end recogniser, the simulation program described in section 6.1.3 was used to generate corrupted phoneme input for the word lattice generator and parser. Corruption rates of 15% and 25% were simulated on the data used for evaluation. The files were corrupted as follows:

File : evoldata1.p.c15
Words : 113
Number of phonemes: 387

		NUM	SUB	DEL	INS
plosives	=	82	4	4	4
affrics	=	4	0	0	0
strfrics	=	36	2	1	2
wkfrics	=	28	2	2	1
liquids	=	48	2	2	1
nasals	=	44	4	4	1
vowels	=	145	11	7	4
TOTALS	=		25	20	13 (58)
TOTALS (%)	=		6.5	5.2	3.4 (15.0)

File : evoldata2.p.c15
Words : 112
Number of phonemes: 369

		NUM	SUB	DEL	INS
plosives	=	81	4	4	4
affrics	=	8	1	0	0
strfrics	=	25	1	1	1
wkfrics	=	37	2	3	2
liquids	=	40	2	2	1
nasals	=	33	3	3	1
vowels	=	145	11	7	4
TOTALS	=		24	20	13 (57)
TOTALS (%)	=		6.5	5.4	3.5 (15.4)

File : evoldata1.p.c25
Words : 113
Number of phonemes: 387

		NUM	SUB	DEL	INS
plosives	=	82	7	7	7
affrics	=	4	1	0	0
strfrics	=	36	3	2	3
wkfrics	=	28	3	4	2
liquids	=	48	3	3	2
nasals	=	44	7	6	2
vowels	=	145	19	11	7
TOTALS	=		43	33	23 (99)
TOTALS (%)	=		11.1	8.5	5.9 (25.6)

File : evoldata2.p.c25
Words : 112
Number of phonemes: 369

		NUM	SUB	DEL	INS
plosives	=	81	7	7	7
affrics	=	8	1	0	0
strfrics	=	25	2	2	2
wkfrics	=	37	4	5	3
liquids	=	40	3	3	2
nasals	=	33	5	5	1
vowels	=	145	19	11	7
TOTALS	=		41	33	22 (96)
TOTALS (%)	=		11.1	8.9	6.0 (26.0)

File : lunddata.p.c15
Words : 5057
Number of phonemes: 17206

		NUM	SUB	DEL	INS
plosives	=	3257	177	167	160
affrics	=	342	36	12	0
strfrics	=	1266	65	42	55
wkfrics	=	1682	107	126	74
liquids	=	2313	90	99	58
nasals	=	1763	172	145	40
vowels	=	6583	507	299	175
TOTALS	=		1154	890	562 (2606)
TOTALS (%)	=		6.7	5.2	3.3 (15.1)

File : lunddata.p.c25
Words : 5057
Number of phonemes: 17206

		NUM	SUB	DEL	INS
plosives	=	3257	291	275	268
affrics	=	342	63	19	15
strfrics	=	1266	109	75	90
wkfrics	=	1682	174	207	125
liquids	=	2313	150	161	97
nasals	=	1763	284	245	67
vowels	=	6583	839	495	293
TOTALS	=		1910	1477	955 (4342)
TOTALS (%)	=		11.1	8.6	5.6 (25.2)

```
File                : wsjdata.c15
Words               : 1923
Number of phonemes: 8006

                          NUM  SUB  DEL  INS
plosives    =        1770   96   91   88
affrics     =         144   16    4    3
strfrics    =         690   36   25   29
wkfrics     =         574   37   43   26
liquids     =         869   34   38   22
nasals      =         961   95   81   22
vowels      =        2998  232  137   81
-------------------------------------------
TOTALS      =              546  419  271 (1236)
TOTALS (%)  =              6.8  5.2  3.4 (15.4)

File                : wsjdata.c25
Words               : 1923
Number of phonemes: 8006

                          NUM  SUB  DEL  INS
plosive     =        1770  159  152  147
affric      =         144   26    7    5
strfric     =         690   59   41   48
wkfric      =         574   61   72   43
liquids     =         869   56   63   36
nasals      =         961  158  135   37
vowels      =        2998  387  228  135
-------------------------------------------
TOTALS      =              906  698  451 (2055)
TOTALS (%)  =             11.3  8.7  5.6 (25.7)
```

8.3 Word Lattice Quality

The word lattice generation parameters were optimised using the two data sets described above using the pre-evaluation dictionary containing 1984 words. The evolutionary programming algorithm (as described in section 6.2.4) with a population of 100 was executed over 100 generations using a tournament size of five. The best solution found had the following settings (to 1 d.p.) for the acoustic parameters:

Corruption Rate	*ins_pen*	*del_pen*	*sub_pen*	
			(same class)	(different class)
15%	97.6	92.4	84.1	218.3
25%	96.7	95.1	94.7	214.8

These parameters were then used to generate further word lattices for the two training data sets and the Lund lecture using the evaluation 2637 word dictionary, and also the WSJ sentences using the evaluation 2388 word dictionary. Average word ranks were calculated for each of the twelve word lattices and the results

Filename	*Word Count*	*Phoneme Error*	*Dictionary Size (Words)*	*Average Rank*
evoldata1	113	15.0	1984	1.2
evoldata2	112	15.4	1984	1.3
evoldata1	113	25.6	1984	1.7
evoldata2	112	26.0	1984	2.0
evoldata1	113	15.0	2637	1.3
evoldata2	112	15.4	2637	1.4
lunddata	5057	15.1	2637	1.5
wsjdata	1923	15.4	2388	2.2
evoldata1	113	25.6	2637	1.7
evoldata2	112	26.0	2637	2.2
lunddata	5057	25.2	2637	2.2
wsjdata	1923	25.7	2388	6.6

Table 8.1: Average Word Ranks for the Training and Evaluation Data

presented in Table 8.1. These figures show that despite increasing the size of the dictionary by 35%, the parameters are robust and produce good average word rank figures for the training data sets.

A useful experiment to perform upon the word rank data is to calculate cumulative word scores. This would reveal the proportion of words occurring at a given rank or better, and allows the observation that 95% of the spoken words occur at, for example, rank 15 or better. If this information were used to prune the word lattice of any words occurring at a rank worse than this, then the search space examined during word lattice parsing would be much reduced. Figure 8.1 shows cumulative word ranks at 15% phoneme error and Figure 8.2 shows cumulative word ranks at 25% phoneme error, on the two training data sets and the Lund lecture, with a dictionary of 2637 words, and on the WSJ sentences with a dictionary of 2388 words.

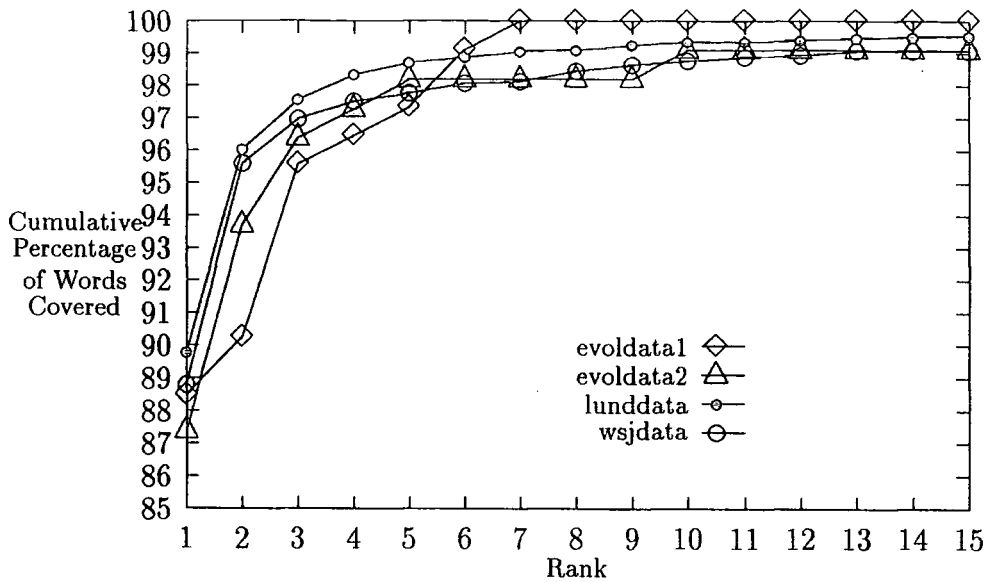


Figure 8.1: Cumulative Percentage of Words at each Rank at 15% Phoneme Error

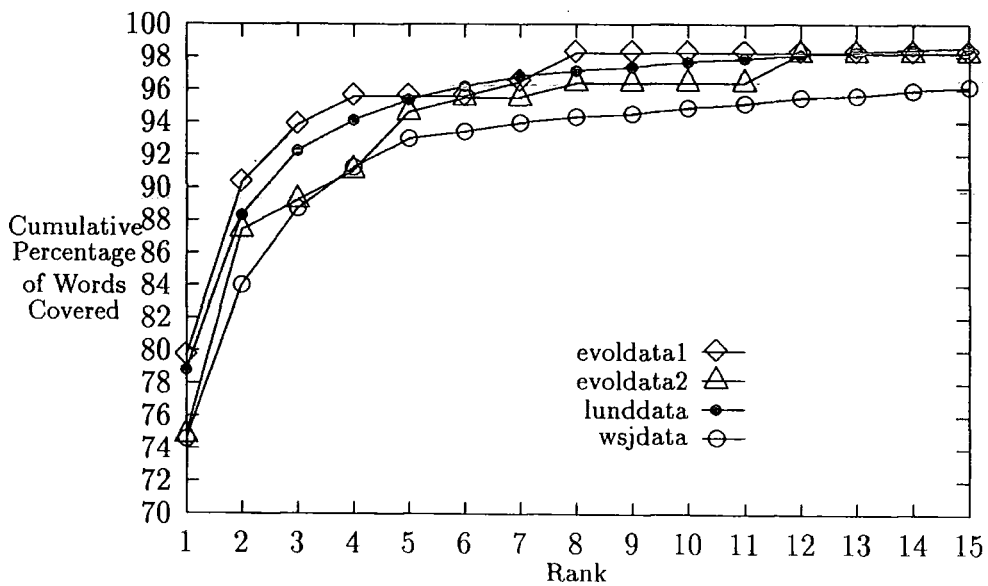


Figure 8.2: Cumulative Percentage of Words at each Rank at 25% Phoneme Error

8.4 Suitability of the Anti-Grammar

8.4.1 Perplexity

The perplexity of the anti-grammar was calculated according to the method given in section 7.3.1. 50,000 sentences (q) of length 12 words (n) using a vocabulary of 1984 (v) words were randomly generated. Using Equation 7.9, perplexity was calculated to be 1470. This experiment was repeated for a vocabulary of 2637 words, perplexity was calculated to be 1913. This information is summarised in Table 8.2.

Vocabulary Size (v)	*Number of Sentences (q)*	*Sentence Length (n)*	*Number of Legal Sentences (l)*	*Perplexity*
1984	50,000	12	1375	1470
2637	50,000	12	1064	1913

Table 8.2: Estimated Perplexity of the Anti-Grammar

8.4.2 Coverage

The anti-grammar was tested for coverage on the SEC corpus. This highlighted several problems of inadequate modelling: compound nouns (or other multiple noun sequences) and numbers are not handled very well. For example "Hong Kong teenagers" and "two hundred thousand tons". Other problems that were encountered were mainly with the different part of speech labelling schemes leading to incorrectly tagged words being passed to the anti-grammar; sentences were also long and although containing many implicit pauses these were not explicitly marked. On the lectures contained within the Durham lecture corpus, most problems concerning coverage were caused by isolated examples of severe speech repair.

8.5 Word Recognition Assessment

Word recognition figures were calculated for the two training data sets and for the Lund lecture, with a 2637 word dictionary, and for the WSJ sentence with a 2388 word dictionary. Results are presented with and without the anti-grammar to demonstrate the effectiveness of the language model. As well as an improvement in recognition rates, using the anti-grammar to reduce the search space has the effect of improving execution times by 30–40%. Recognition results are presented in Table 8.3, and in graph form in Figure 8.3 and Figure 8.4. Word recognition assessment was carried out using a dynamic programming scoring package supplied by CUED, based on the ARPA speech recognition evaluation software. Examples of system recognition are given in appendix B.

At a phoneme error rate of 15%, the anti-grammar improved the percentage words correct by 1.5%–5.9%, and at a phoneme error rate of 25%, the anti-grammar improved the percentage words correct by 6.2%–18.7%. The conclusion is therefore that the anti-grammar is more helpful at higher rates of phoneme error, but that it still brings an improvement in word recognition at lower rates of phoneme error.

Recognition times for the Lund lecture are given in Table 8.4. This table shows that with 15% phoneme error, word recognition occured at approximately 8 seconds per word, and with 25% phoneme error, word recognition occured at approximately 11 seconds per word. The execution times in the table were obtained using a multi-user SUN SparcCenter 2000. The results demonstrate that using the anti-grammar has little overhead on recognition times, yet still achieves an increase in word recognition.

Filename	*Word Count*	*Phoneme Error (%)*	*Words Correct (%)*	*Word Error (%)*			*Word Accuracy (%)*
				Ins	*Sub*	*Del*	
evoldata1	113	15.0	79.6	0.9	15.9	4.4	78.8
evoldata2	112	15.4	86.6	3.6	9.8	3.6	83.0
lunddata	5057	15.1	83.5	2.7	12.9	3.7	80.7
wsjdata	1923	15.4	82.3	2.9	13.6	4.1	79.5
evoldata1	113	25.6	76.1	5.3	21.2	2.7	70.8
evoldata2	112	26.0	81.2	3.6	16.1	2.7	77.7
lunddata	5057	25.2	73.1	5.0	21.0	5.8	68.2
wsjdata	1923	25.7	70.1	7.3	20.9	9.0	62.8
evoldata1 (no ag)	113	15.0	76.1	1.8	20.4	3.5	74.3
evoldata2 (no ag)	112	15.4	81.2	2.7	14.3	4.5	78.6
lunddata (no ag)	5057	15.1	77.6	3.0	17.7	4.7	74.6
wsjdata (no ag)	1923	15.4	80.8	3.3	13.9	5.3	77.5
evoldata1 (no ag)	113	25.6	69.9	6.2	18.6	11.5	63.7
evoldata2 (no ag)	112	26.0	62.5	7.1	33.0	4.5	55.4
lunddata (no ag)	5057	25.2	64.8	5.9	27.8	7.4	58.9
wsjdata (no ag)	1923	25.7	59.3	6.9	25.4	15.3	52.4

Table 8.3: Word Recognition Rates with a 2637 Word Dictionary

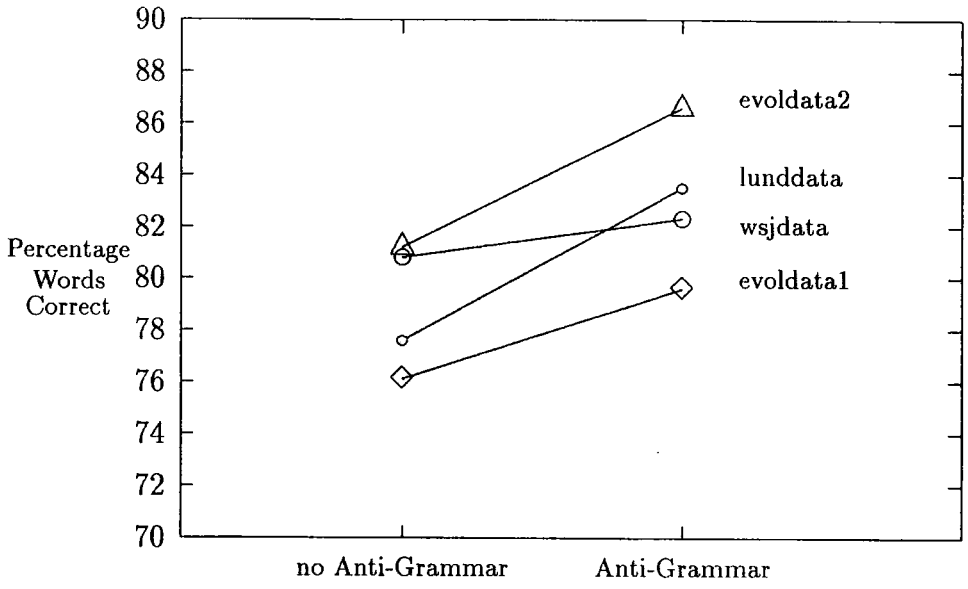


Figure 8.3: Word Recognition Rates for the Training Data at 15% Phoneme Error

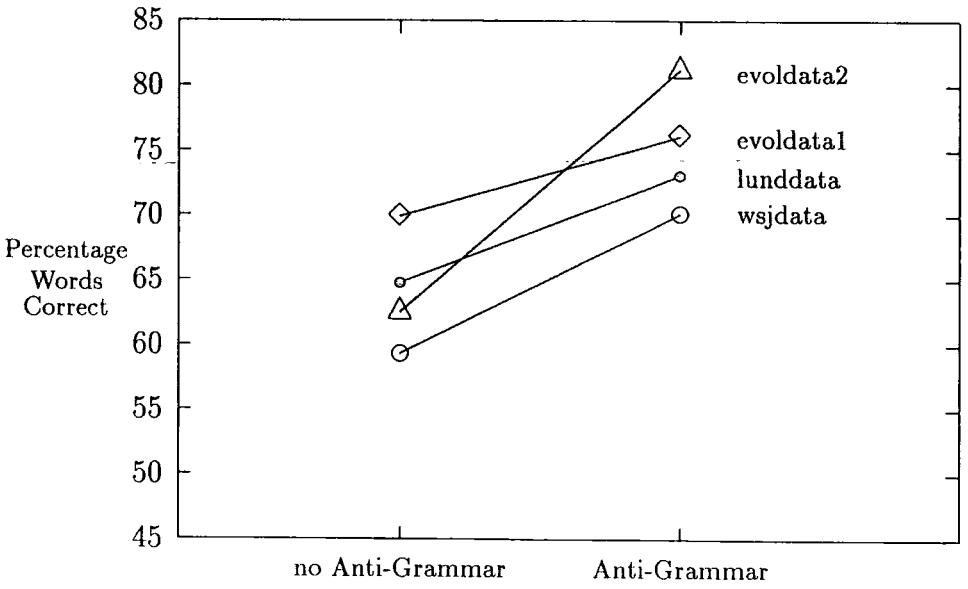


Figure 8.4: Word Recognition Rates for the Evaluation Data at 25% Phoneme Error

Lecture File	*Word Count*	*Time (Minutes)*	
		15% Phoneme Error	*25% Phoneme Error*
lunddata.part1	239	30	41
lunddata.part2	254	42	32
lunddata.part3	252	29	29
lunddata.part4	244	31	43
lunddata.part5	253	30	51
lunddata.part6	253	27	38
lunddata.part7	257	39	54
lunddata.part8	258	36	37
lunddata.part9	255	39	55
lunddata.part10	246	29	43
lunddata.part11	244	31	43
lunddata.part12	263	38	58
lunddata.part13	274	38	58
lunddata.part14	255	35	55
lunddata.part15	253	39	44
lunddata.part16	254	32	57
lunddata.part17	261	38	50
lunddata.part18	260	43	47
lunddata.part19	256	36	54
lunddata.part20	226	27	47
Totals	5057	689	936
Average Seconds Per Word		8.2	11.1
Lecture File	*Word Count*	*Time (Minutes)*	
		15% Phoneme Error	*25% Phoneme Error*
lunddata.part1.noag	239	32	30
lunddata.part2.noag	254	36	32
lunddata.part3.noag	252	28	24
lunddata.part4.noag	244	33	31
lunddata.part5.noag	253	28	48
lunddata.part6.noag	253	27	42
lunddata.part7.noag	257	40	47
lunddata.part8.noag	258	34	47
lunddata.part9.noag	255	39	65
lunddata.part10.noag	246	30	45
lunddata.part11.noag	244	29	44
lunddata.part12.noag	263	35	35
lunddata.part13.noag	274	36	56
lunddata.part14.noag	255	32	68
lunddata.part15.noag	253	39	43
lunddata.part16.noag	254	28	47
lunddata.part17.noag	261	45	53
lunddata.part18.noag	260	39	49
lunddata.part19.noag	256	35	55
lunddata.part20.noag	226	28	42
Totals	5057	673	903
Average Seconds Per Word		8.0	10.7

Table 8.4: Word Recognition Execution Times on the Lund Lecture Using a 2637 Word Dictionary (with and without the Anti-Grammar)

8.6 Readability

Cloze readability tests were given to fifteen recent graduates in computer science. Two texts were given to each participant: part of a transcribed lecture on software engineering, and part of the Lund corpus lecture. One of the texts was in its original form, and the other text was system output at 25% phoneme error (simulated).

The instructions to each participant are given in Figure 8.5. The four texts are given in Figure 8.6, Figure 8.7, Figure 8.8 and Figure 8.9, and the answers in Figure 8.10 and Figure 8.11. The results of the Cloze test are summarised in Table 8.5.

Although it is difficult to make any judgement on the meaning of *absolute* Cloze test scores because of the wide variability in textual material, Bormuth gave some general indications that may be used [Robinson, 1981]. He stated that a Cloze test score of less than 37% indicates that a reader would find a text frustratingly difficult; a score of over 57% indicates that the reader can reasonably be expected to understand the text.

The test results are lower than expected, indicating that even the original lectures, with mean Cloze test scores near Bormuth's borderline, are not very readable. One possible reason for this is that there is a certain amount of redundancy in *spoken* English: because it is so informal, the same thing can be said in many different ways; and also the spontaneous nature of spoken English is confusing when written down. In addition the written form does not contain prosodic information, and the reader does not have the context of the speech, for example the location or any gestures made by the speaker. The results for the system output were always going to be worse than the original, because of the high amount of word error. These results possibly invalidate the use of the Cloze procedure for measuring the output of speech recognisers at such high word error rates; on reflection the test should perhaps have been tried on system output of texts with 15% phoneme error.

Text	*Number of Blank Words*	*Mean Score*	*Best Score*	*Worst Score*
1. software engineering lecture (original)	50	31.4	38	15
2. Lund corpus lecture (original)	50	30.4	33	28
3. software engineering lecture (ASR output, 25% phoneme error)	50	15.9	22	9
4. Lund corpus lecture (ASR output, 25% phoneme error)	50	18.5	22	13

Table 8.5: Cloze Readability Assessment Results

The exercises contained in this test are known as "cloze exercises". A cloze exercise consists of the presentation of a passage from which a number of words have been deleted. The task is to attempt to guess the deleted words. The aim of the test is to measure the readability of the passages, *not* your language ability.

Having made your choice, don't be tempted to start filling the blanks too soon. First, read through the passage to the end, to get the general sense of it, how it is structured, where the topics seem to change, etc. Then go through it again, trying to fill each blank with *just one word* (i.e. not a phrase of two or more words).

Abbreviations (*UNESCO*), contractions (*I'd* or *we're*), hyphenated forms (*half-baked*) and dates (*13th* or *1978*) count as one word.

In choosing your response, you will need to look very carefully at the grammar of the construction, to see what kinds of words might fit; you will need to consider the meaning of the word; and you will have to decide what kind of style is used in a passage. The length of the line indicating a blank is constant and in no way related to the length of the missing word.

Each of the passages comes from an accurately transcribed undergraduate lecture in computer science, complete with all speech disfluencies. For this reason, any punctuation in the passages has been added by hand as accurately as possible. Some of the passages are the output from a speech recognition system.

Figure 8.5: Instructions for the Cloze Readability Assessment

That just to say what the course is in case you're confused. Course on software maintenance. We've got nine lectures, it's not very much time to say very much about this subject. Very briefly (1) _____ syllabus is as follows. (2) _____ may or may not (3) _____ exactly to this. This lecture is going to be (4) _____ introductory scenario. If I (5) _____ figure out where the (6) _____ switches are. That will (7) _____ . This, this lecture is (8) _____ to be an introduction. (9) _____ going to tell you (10) _____ bit more about maintenance (11) _____ I told you last (12) _____ . Then I'm going to (13) _____ a lecture or two (14) _____ about models of the (15) _____ process, and there are (16) _____ different types of models. Starting, it's almost, you can (17) _____ it's an historic, an (18) _____ review of models, bringing (19) _____ right up to date with current thinking. Erm, then (20) _____ going to, no we're (21) _____ , then we're going to (22) _____ at least one lecture (23) _____ how do we measure what happens in software maintenance. (24) _____ we measure old software (25) _____ say we should throw (26) _____ away. We should do (27) _____ to it, we should (28) _____ that to it. Quite (29) _____ interesting subject but not (30) _____ much work has been done on it.

Then we'll (31) _____ at the subject that's (32) _____ reverse engineering. Now what (33) _____ been doing, what we (34) _____ last year in software (35) _____ can be termed forward (36) _____ , ie we go through (37) _____ design etc and produce (38) _____ software at the end. (39) _____ reverse engineering is about, (40) _____ simply, is to take (41) _____ all these developers leave (42) _____ with and to try get back to what (43) _____ think may be the (44) _____ or specifications is about. (45) _____ , there is quite a (46) _____ of research going on (47) _____ reverse engineering, to try and capture the knowledge that's (48) _____ in current systems. This (49) _____ probably the most interesting (50) _____ of the course.

Figure 8.6: Cloze Passage Text 1: Software Engineering, Original

Well rather than give a talk about the history of stoke poges, I felt it might be a little more interesting to you all (1) _____ hear about my own (2) _____ . Lived and growing up (3) _____ this wonderful village of (4) _____ poges. I attended stoke (5) _____ and I must say (6) _____ was taught very thoroughly (7) _____ three Rs. Funnily enough my father (8) _____ to the same school, (9) _____ he was one of the first pupils. Before that (10) _____ used to go to (11) _____ school next door to (12) _____ and pay a penny (13) _____ week, along with all (14) _____ other village boys for (15) _____ education. Considering his schooling (16) _____ have stopped at about fourteen years, his beautiful copperplate (17) _____ and his reading with (18) _____ was really remarkable.

I (19) _____ in my early life (20) _____ wexham street. It was a semi detached house, built (21) _____ my father and uncle with their own hands. And (22) _____ lived we were a (23) _____ of five, there were three children, two sisters and (24) _____ . We had a very (25) _____ garden, and we used (26) _____ have to produce the (27) _____ from the garden. The (28) _____ . The root crops. Store them for the use of (29) _____ during the whole of (30) _____ winter. We also kept (31) _____ everyone in the village (32) _____ that time kept chicken. (33) _____ kept a goat, rabbits, (34) _____ occasionally we used to (35) _____ a pig. The chickens (36) _____ looked after by my (37) _____ . The goat I had (38) _____ milk myself. And that (39) _____ how we used to (40) _____ .

My mother was a (41) _____ industrious woman, used to (42) _____ all the jam and (43) _____ to last us throughout (44) _____ year. She found time (45) _____ make my fathers shirts, (46) _____ all our jerseys, for that's what we wore, all (47) _____ socks for the children. (48) _____ I remember she used (49) _____ make my suits up (50) _____ the age of about twelve.

Figure 8.7: Cloze Passage Text 2: Lund Lecture, Original

That just to say what the course is in course you're confused. Cause on software maintenance. We've got nine lectures, it's not very much time to say favourite much about this subject. Very briefly (1) _____ syllabus is as follows. (2) _____ may or may not (3) _____ exactly to this. Directories going to be (4) _____ introductory scenario. If I (5) _____ of figure out way air the (6) _____ switches are. At will (7) _____ . This, this lecture is up (8) _____ to be an introduction. All (9) _____ going to tell you (10) _____ bit more about maintenance (11) _____ I told you last (12) _____ . Then I'm go in to (13) _____ a lecture or two (14) _____ about models of the (15) _____ process, and air are (16) _____ different I'd place of models. Starting, built so least, you'd can (17) _____ an historic, month (18) _____ day few of. Bringing (19) _____ at up today up refer a. Erm, then (20) _____ go it, a no while (21) _____ , of then we're going to (22) _____ at least one later (23) _____ how do do we my she iterate open means in software main got a man a. (24) _____ we I'm sure old software (25) _____ say we should throw (26) _____ away. We should do (27) _____ to it, we should (28) _____ fact to it. Quite (29) _____ interesting subject but not (30) _____ much one has been company.
Of then why (31) _____ the subject that's (32) _____ reverse engineering. Now what (33) _____ been studying, what we (34) _____ last were in software (35) _____ can be termed forward (36) _____ , ie we gave through you (37) _____ doesn't etc and produce (38) _____ software at the end. Air (39) _____ engineering is about, (40) _____ simply, is to take (41) _____ all these developers leave (42) _____ with and today tried get bad to what (43) _____ think may be the (44) _____ or specifications is about. (45) _____ , though required to (46) _____ of researching on (47) _____ reverse engineering, to plan capture the knowledge it's (48) _____ in grants systems. This (49) _____ probably them interesting (50) _____ of the course.

Figure 8.8: Cloze Passage Text 3: Software Engineering, System Output

Well rather than give a talk about the history of stoke age is, I felt it might be a little interest to you all (1) _____ her about my own (2) _____ . Already and growing up (3) _____ this wonderful following of (4) _____ poges. I attended stoke (5) _____ and i'm asked say (6) _____ what not very thoroughly (7) _____ three Rs. Funnily enough my father (8) _____ to the aim school, (9) _____ he was one. Before at (10) _____ used ago to (11) _____ school next or to (12) _____ and pay a penny (13) _____ week, along now with all (14) _____ other village boys or they (15) _____ education. Considering his school in (16) _____ have stopped at about forty, if his beautiful copperplate (17) _____ and his read with (18) _____ was real air remarkable.

I (19) _____ in my early live (20) _____ wexham street. It was assuming detached house, bill (21) _____ my father and tackle where own hands. And (22) _____ early poured we worry (23) _____ , there worthwhile children, two sisters and (24) _____ . We had a very (25) _____ garden, and we used (26) _____ though have to reproduce the (27) _____ from the good. The (28) _____ . The right crops. Is story more the u of (29) _____ during the whole of (30) _____ winter. We also kept (31) _____ everyone in the fill each (32) _____ the got time kept chicken. (33) _____ kept a got, rabbits, (34) _____ occasionally we used to (35) _____ a. The chickens (36) _____ looked after by my (37) _____ . The go but I had (38) _____ milk my. Add at (39) _____ how we used to (40) _____ .

My mother was a (41) _____ industrious woman, used to (42) _____ all the jam and (43) _____ to last us throughout (44) _____ year. She found too i'm (45) _____ make my are a, (46) _____ it all hours is, for the its what we where, all (47) _____ socks for the children. (48) _____ I remember she used (49) _____ may my suits up (50) _____ the age of about twelve.

Figure 8.9: Cloze Passage Text 4: Lund Lecture, System Output

(1) the	(2) I	(3) stick
(4) an	(5) can	(6) light
(7) do	(8) going	(9) it's
(10) a	(11) than	(12) year
(13) spend	(14) talking	(15) maintenance
(16) various	(17) say	(18) historic
(19) you	(20) we're	(21) not
(22) have	(23) on	(24) can
(25) to	(26) it	(27) this
(28) that	(29) an	(30) very
(31) look	(32) called	(33) we've
(34) did	(35) engineering	(36) engineering
(37) requirements	(38) some	(39) what
(40) very	(41) what	(42) us
(43) we	(44) design	(45) and
(46) lot	(47) into	(48) embodied
(49) is	(50) part	

Figure 8.10: Answers to Cloze Passage Texts 1 and 3

(1) to	(2) life	(3) in
(4) Stoke	(5) school	(6) I
(7) the	(8) went	(9) and
(10) he	(11) the	(12) here
(13) a	(14) the	(15) his
(16) must	(17) writing	(18) understanding
(19) lived	(20) in	(21) by
(22) we	(23) family	(24) myself
(25) big	(26) to	(27) produce
(28) potatoes	(29) us	(30) the
(31) as	(32) at	(33) we
(34) and	(35) keep	(36) was
(37) sisters	(38) to	(39) is
(40) manage	(41) very	(42) make
(43) preserves	(44) the	(45) to
(46) knit	(47) our	(48) and
(49) to	(50) to	

Figure 8.11: Answers to Cloze Passage Texts 2 and 4

Chapter 9

Conclusions and Future Work

This chapter concludes the thesis by checking if this work has met its criteria for success; discussing future research directions; and describing what this work can offer researchers in the field of automatic speech recognition and also what it can offer the deaf community.

9.1 Conclusions

The criteria for the success of the work described in this thesis were given in section 1.2. The system will be evaluated according to each of these criteria. The project was unable to deliver a full working system as appropriate "off-the-shelf" speech recognition toolkits were not available (at least not until the end of the research period).

Scale : the system has a large vocabulary, currently containing over 2600 words; vocabulary size was discussed in section 3.2.7, large vocabulary was defined to be 1,000–5,000 words;

Robustness : the system was developed and evaluated on substantially different sets of data, demonstrating domain independence; the system proved to be robust on different input data and also when the vocabulary was increased by 35%, without requiring retraining (section 8.3);

Integration : the system has been designed to allow other sources of knowledge, such as semantics, repair or prosody, to be easily integrated into the word lattice parsing process; section 5.4 described the use of anti-grammar and word frequency knowledge, and discussed the integration of further sources of knowledge, giving the use of weak semantics as an example; the software engineering aspects of integration were described in section 6.5;

Feasibility : the system runs quickly on a multi-user SUN SparcCenter 2000, taking approximately 8 seconds to recognise each word (section 8.5);

Maintenability : the system is flexible enough to allow changes in word frequency information (section 6.3.4) and grammatical categorisation (section 6.2.5), which would bring benefits; the software engineering aspects of maintenance were described in section 6.5;

Usability : the system currently only uses a simulated continuous speech phoneme recognition system, and awaits connection to a suitable hardware front-end; a useful level of word recognition (73.1%) is achieved at the level of 25% phoneme error, a substantially higher recognition rate (83.5%) is achieved at the 15% phoneme error level (section 8.5); experience in the development of Palantype showed that a 75% correct transcription was very useful to well motivated deaf people.

Techniques : the system makes use of a variety of techniques: symbolic (anti-grammar rules), adaptive (word lattice generation parameters), statistical (word frequency information), heuristic (word lattice parsing) and corpus-based (anti-grammar rules and evaluation).

The achievements in the areas of scale and robustness have been achieved with the most success; usability has partially been fulfilled due to the word recognition

rates that are achieved, although full usability has not been achieved because the system makes use of a simulated phoneme recognition system.

9.2 Future Research Directions

There are several lines along which the research presented in this thesis should progress.

Phoneme Recognition

Work will continue on developing a phoneme recognition front-end in collaboration with the DRA and CUED. Accuracy of the front end needs to be at least 75% correct phoneme recognition. Integration between the phoneme recogniser and the word lattice generation software can then take place.

Word Frequency

The accuracy of the system would increase substantially if the word frequency information were improved by extending the number of frequency categories from three (very common, normal, very rare). This has to be done with care as the finer the level of granularity that is used, the more domain specific the information becomes.

Word Categories

The method of grammatically categorising each word, currently using the OALD, should be changed to use a more informative notation such as that in the SEC corpus and the CLAWS part of speech tagger.

Active Vocabulary

A "window" analysis of the text of a lecture, in other words dividing a lecture into, say, 20 sections and calculating word frequencies, indicates some interesting possibilities for future work. The one most likely to increase accuracy would be to implement some kind of active (or cache) vocabulary, in other words, a list of most recently recognised words is kept and these are given a preference over other words. This would have to be done at various levels because the most common words in English require special treatment as they occur every two or three words in a sentence.

Other Sources of Knowledge

Work has already begun on incorporating further sources of knowledge into the word lattice parsing stage. This includes work on repair — identifying repair in sentence hypotheses, correcting the repair and re-scoring the corrected hypothesis; semantics — using "weak" semantics to give a semantic likelihood score to the co-occurrence of verbs, nouns and adjectives in a sentence hypothesis; and prosody — analysing the prosodic properties of a portion of speech to build a "prosodic template" that can be matched against sentence hypotheses to give a prosodic score.

A further source of knowledge might be to use "weak" n-gram statistics for word (or part of speech) sequences. These are weak in the sense that they are few in number and only cover the most common constructs. The anti-grammar contains only a small number of "bonus" rules, this could be expanded and would help improve the accuracy of the system. Before this takes place, an evaluation study comparing the performance of an n-gram word or part of speech language model to that of the anti-grammar.

Generalised Test-Bench

Further work may be pursued to generalise the test-bench even more to provide a generic toolkit for linguistic constraint researchers so that syntactic and semantic constraint models could be researched at other sites without speech recognition hardware.

Integration with LOLITA

When a sufficiently high level of word recognition is achieved, say 85–90%, then the intention is to integrate the speech recognition system with the LOLITA natural language understanding system. This could then open up many branches of research as the LOLITA system provides many possible applications such as query, dialogue, summarisation and translation.

Measurement of Meaning

The utility of a meaning measurement was introduced in section 7.6 and five areas which could benefit were briefly mentioned. Speech recognition assessment was discussed in detail. The four remaining areas are described below: use of domain knowledge to aid speech recognisers; summarisation and content scanning of text; topic spotting and assessment of machine translation.

Automatic speech recognition systems are growing in vocabulary size day by day, with this comes the increased likelihood that words are going to be confused for each other during the recognition process, for example: "It is hard to recognise speech" could easily be recognised as: "It is hard to wreck a nice beach".

Increasingly, speech recognition systems are making use of domain specific knowledge in order to simplify the recognition task. In the example just given, we could imagine a scenario of a lecturer talking about artificial intelligence to a group of students and clearly the first sentence makes sense. The second in-

terpretation is completely out of context, and this could be detected if we had a measurement of semantic distance. The solution is not quite as simple as that, however, as a counter example to this use of semantic distance would be when a lecturer introduces an analogy using several out of context words. This problem could be overcome by a semantic clustering technique: the presence of several semantically related yet out of context words would not be penalised during the recognition process [Short *et al.*, 1994a]. Again, this relies on a measure of semantic distance.

In a world containing vast amounts of electronic information, summarisation and content scanning tools are becoming more and more desirable. In America, a large amount of ARPA funding is dedicated to the MUC (Message Understanding Conference) project in which several groups compete annually to produce a computer system that can extract relevant information from newswire articles on specific subjects, such as terrorism.

It is quite possible to build a very shallow system that could parse the newswire articles at the surface level, looking for special keywords for example, that would achieve quite a good level of performance. This kind of system would, however, fail completely if it were given a completely new subject domain. Typical problems faced by such shallow systems are those of negation, time and distance. A measure of semantic distance can be used to identify the crucial parts of a text, such that if certain information is missing, the meaning is completely altered. For example, "not guilty" is changed to "guilty", "100 miles from London" is changed to "in London", or "the week after next" is changed to "next week". To be more successful, a deeper syntactic and semantic analysis must take place, a task which is well suited to the LOLITA system being developed here at Durham [Garigliano *et al.*, 1993a]. To make such a system even more general purpose will require a measure of semantic distance.

Topic spotting is a mechanism that is often needed before the summarisation or content scanning process can take place. It involves spotting pieces of text that are relevant to a particular topic or subject. Again, it is quite possible to produce

a superficial domain specific system that uses a pattern matching approach, but for a more general system that can work in a variety of domains with the minimum of initialisation, a measure of semantic distance will be required.

Currently, the only way of assessing the performance of machine translation systems is to use a human knowledgeable in both source and target languages. A piece of text in one language could be converted into it's semantic representation and compared using a semantic distance measure to the semantic representation of the translated text in the second language.

9.3 Impact on the Field of Automatic Speech Recognition

The work described in this thesis has three main contributions to make to the field of automatic speech recognition:

- the most important contribution is the use of anti-grammar rules to check the syntactic incorrectness of sequences of words, providing a domain independent method of reducing the large search space, represented as a word lattice, whilst at the same time allowing normal spontaneous English to be spoken;
- a system designed to allow ease of integration with new sources of knowledge, such as semantics, prosody or repair, in effect providing a test-bench for determining the impact of different knowledge upon word lattice parsing without the need for the underlying speech recognition hardware.
- the use of evolutionary programming to determine near-optimal robust parameters for word lattice creation, making the system dependent upon only the performance of the underlying continuous speech phoneme recognition system; the parameters being robust enough to withstand changes in vocabulary and domain;

9.4 Impact on the Deaf Community

This research has not fully met the deaf user's *ideal* requirements of an automatic speech recognition system, outlined in section 2.4. However, the research satisfies some of these requirements and provides an initial stepping-stone for future work to satisfy those that remain. When the system is fully connected to a continuous speech phoneme recognition system, then full user evaluation may take place. The ultimate aim of a "talkwriter" is still many years away, but this research offers some interesting results that can contribute towards producing a useful system for deaf university students.

Appendix A

Anti-Grammar Rules

This appendix lists the anti-grammar rules used by the system. These can be categorised into rules that give a bonus to a sentence hypothesis containing a particular structure; simple rules that give a penalty to a sentence hypothesis containing a particular structure; and complicated rules that give a penalty to a sentence hypothesis containing a particular structure.

Rules That Give a Bonus

```
very ADJ

very ADV

PREP(to) VERB(to_verb_word)

VERB ADV("not") VERB VERB
                specifically:  modal + "not" + be + present participle
                               modal + "not" + be + past participle
                               modal + "not" + have + past participle

VERB VERB ADV(not "not") VERB
                specifically:  modal + "be" + ADV + present participle
                               modal + "be" + ADV + past participle
                               modal + "have" + ADV + past participle

VERB VERB VERB VERB
```

```
                specifically:  modal + "have" + "been" + present participle
                               modal + "have" + "been" + past participle

VERB VERB VERB
                specifically:  modal + "be" + present participle
                               modal + "be" + past participle
                               modal + "have" + past participle
                               "have" + "been" + present participle
                               "have" + "been" + past participle
```

Simple Rules That Give a Penalty

```
ADJ   ADJ   ART
ADJ   ADV   ART
ADJ   ADV   NOUN
ADJ   ART   ADV
ADJ   CONJ  NOUN
ADJ   PREP  CONJ
ADJ   PREP  PREP
ADJ   PRON  ADJ
ADJ   PRON  ART
ADJ   PRON  CONJ
ADJ   PRON  NOUN
ADJ   PRON  PREP
ADJ   PRON  PRON
ADV   ART   ADV
ADV   CONJ  NOUN
ADV   NOUN  ADJ
ADV   NOUN  ADV
ADV   NOUN  ART
ADV   NOUN  CONJ
ADV   NOUN  PRON
ADV   PRON  ART
ADV   PRON  CONJ
ADV   PRON  PREP
ART   ADJ   ADV
ART   ADJ   ART
ART   ADJ   PREP
ART   ADJ   PRON
ART   ADJ   VERB
ART   ADV   ADV
ART   ADV   ART
ART   ADV   CONJ
ART   ADV   NOUN
ART   ADV   PREP
ART   ADV   PRON
ART   ADV   VERB
ART   ART
ART   CONJ
ART   PREP
ART   PRON
```

```
CONJ  ADJ   ADV
CONJ  ADV   CONJ
CONJ  ART   ADV
CONJ  CONJ  ADJ
CONJ  CONJ  ADV
CONJ  CONJ  ART
CONJ  CONJ  CONJ
CONJ  CONJ  NOUN
CONJ  CONJ  VERB
CONJ  NOUN  ADV
CONJ  NOUN  ART
CONJ  PREP  ADV
CONJ  PREP  CONJ
CONJ  PREP  PREP
CONJ  PREP  PRON
CONJ  PRON  ART
CONJ  PRON  CONJ
NOUN  ADJ   ART
NOUN  ADJ   PRON
NOUN  ART   ADV
NOUN  PREP  CONJ
NOUN  PRON  ART
NOUN  PRON  CONJ
NOUN  PRON  PREP
PREP  ADJ   ADJ   VERB
PREP  ADV   CONJ
PREP  ADV   NOUN
PREP  CONJ  ADJ
PREP  CONJ  ART
PREP  CONJ  CONJ
PREP  CONJ  NOUN
PREP  CONJ  PREP
PREP  CONJ  VERB
PREP  PREP  ADV
PREP  PREP  CONJ
PREP  PREP  PREP
PREP  PREP  PRON
PREP  PRON  ART
PRON  ADJ   ADV
PRON  ADJ   ART
PRON  ADJ   CONJ
PRON  ADJ   PRON
PRON  ADV   NOUN
PRON  ART   ADV
PRON  CONJ  ADJ
PRON  CONJ  ART
PRON  CONJ  CONJ
PRON  CONJ  NOUN
PRON  CONJ  PREP
PRON  NOUN  ADJ
PRON  NOUN  ADV
PRON  NOUN  ART
PRON  NOUN  CONJ
PRON  PREP  ADV
PRON  PREP  CONJ
PRON  PREP  PREP
PRON  PREP  PRON
```

```
PRON  PRON  ADJ
PRON  PRON  ART
PRON  PRON  CONJ
PRON  PRON  PREP
VERB  ADJ   ADJ   VERB
VERB  CONJ  NOUN
ADJ   ADJ
NOUN  NOUN  NOUN
PRON  NOUN  NOUN
```

Complicated Rules That Give a Penalty

```
ADJ(not pre_determiner_word) ART ADJ

ADJ(not pre_determiner_word) ART NOUN

ART ADV(not "not" and not adv_modifies_adj)

ART VERB(not present participle and not past participle)

CONJ(normal) CONJ(normal)

PREP PRON(nominative)

PREP PRON(relative and "that")

PREP(not to) VERB(not present participle)

PRON(interrogative) PRON(interrogative)

PRON(relative) PRON(relative)

VERB ADV( not "to") VERB
                except for: modal verb + ADV + baseform
                            do verb + ADV + baseform
                            be verb + ADV + present participle
                            be verb + ADV + past participle
                            have verb + ADV + past participle

VERB ADV("not") ADV(not "not" and not "to") VERB
                except for: modal verb + ADV + ADV + baseform
                            do verb + ADV + ADV + baseform
                            be verb + ADV + ADV + present participle
                            be verb + ADV + ADV + past participle
                            have verb + ADV + ADV + past participle

VERB ADV("not" or "to") VERB VERB
                except for: modal verb + ADV + "be" + present participle
                            modal verb + ADV + "be" + past participle
                            modal verb + ADV + "have" + past participle
```

```
VERB PRON(nominative)

VERB VERB ADV(not "not" and not "to") VERB
                except for: modal verb + "be" + ADV + present participle
                            modal verb + "be" + ADV + past participle
                            modal verb + "have" + ADV + past participle

VERB VERB VERB VERB
                except for: modal verb + "have" + "been" + present participle
                            modal verb + "have" + "been" + past participle

VERB VERB VERB
                except for: modal verb + "be" + present participle
                            modal verb + "be" + past participle
                            modal verb + "have" + past participle
                            "have" + "been" + present participle
                            "have" + "been" + past participle

VERB VERB
                except for: do verb + baseform
                            modal verb + baseform
                            be verb + present participle
                            be verb + past participle
                            have verb + past participle

"a" word_with_initial_vowel

"a" plural_word

"an" word_without_initial_[vowel,h]

"an" plural_word

very(ADJ) not (ADJ or ADV)

ADJ VERB(not link_verb)

NOUN ADJ ADV(not "not") VERB

NOUN(not singular) NOUN

NOUN(singular) VERB(non_anomalous and not 3rd_person
                    and not past participle and not present participle)

PREP("to") VERB(not verb_that_can_follow_to)

genitive not (NOUN or ADJ)
```

Appendix B

Example System Recognition

This appendix shows system recognition for the first 31 sentences of the LUND lecture and the first 40 sentences of the WSJ sentences that were used for evaluation. In addition to the original sentence, system output is given for 15% and 25% phoneme corruption rates.

LUND Lecture

```
original       : well rather than give a talk about the history of stoke poges
15% corruption : well rather than give a talk about the history of stoke poges
25% corruption : l rather than give a talk about the things true of stoke
                 each is

original       : i felt it might be a little more interesting to you all
15% corruption : i've felt it night be a lit long more interesting to you all
25% corruption : i to

original       : to hear about my own life
15% corruption : to hear about my own life
25% corruption : era out my own life

original       : lived and growing up in this wonderful village of stoke
                 poges
15% corruption : live damn growing up in this wonderful village of stoke
                 poges
25% corruption : live do and grown real up in this wonderful village of
```

stoke poges

original : i attended stoke school and
15% corruption : i o attended stoke school and
25% corruption : i attended talk school and

original : i must say
15% corruption : i'm just say
25% corruption : i'm state

original : i was taught very thoroughly the three rs
15% corruption : i was taught very thoroughly the three rs
25% corruption : i was talk very thoroughly the three rs

original : funnily enough my father went to the same school
15% corruption : funnily enough my father went to the say school
25% corruption : funnily enough my my other went to the say and

original : and he was one of the first pupils
15% corruption : and he was one of the best pupils
25% corruption : he was one of the first pupils

original : before that
15% corruption : before that
25% corruption : before that

original : he used to go to the school next door to here
15% corruption : he used to goat others collected or to here
25% corruption : he used to go together complexity door to here

original : and pay a penny a week
15% corruption : and pay a penny a week
25% corruption : add pay a penny away week

original : along with all the other village boys for his education
15% corruption : along with although other village boys for how is education
25% corruption : along with all the other villagers for his education

original : considering his schooling must have stopped at about fourteen years
15% corruption : are considering his schooling must have stopped at about fourteen years
25% corruption : considering his schooling must have stop at about thought emu's

original : his beautiful copperplate writing and his reading with understanding was really remarkable
15% corruption : his beautiful copperplate writing and his reading with understanding was really remarkable
25% corruption : is beautiful top pay right inadequate thing with understanding was really remarkable

original : i lived in my early life in wexham street
15% corruption : i live in mileage life in wexham street
25% corruption : i lived in my early wife in wexham street

original : it was a semi detached house

```
15% corruption : it was a semi detached how
25% corruption : it was a semi detached face

original       : built by my father and uncle with their own hands
15% corruption : built by my father and some cliff thrown hands
25% corruption : built by my rather second uncle with their own hands

original       : and we lived we were a family of five
15% corruption : and we lived we were a family of five
25% corruption : and we'll of day we where a family of five

original       : there were three children
15% corruption : there were three children
25% corruption : there were reach children

original       : two sisters and myself
15% corruption : two show sisters and myself
25% corruption : two sisters and myself

original       : we had a very big garden
15% corruption : we had a very big hard
25% corruption : we had a very big garden

original       : and we used to have to produce the produce from the garden
15% corruption : and we used to have to produce the produce from the garden
25% corruption : and we used to have to produce the produce from the garden

original       : the potatoes
15% corruption : the potatoes
25% corruption : the potatoes

original       : the root crops
15% corruption : the right crops
25% corruption : the room crops

original       : store them keep them for the use of us during the whole of
                 the winter
15% corruption : store them might keep them for the use of us during the
                 whole of the winter
25% corruption : show storey them keep then for the yes of us during the
                 whole of the winter

original       : we also kept
15% corruption : we'll so kept
25% corruption : we also kept

original       : as everyone in the village at that time kept chicken
15% corruption : as everyone enough village at that time kept chicken
25% corruption : as everyone in the off village at at time kept chicken

original       : we kept a goat
15% corruption : we kept ago
25% corruption : we kept about

original       : rabbits and occasionally we used to keep a pig
15% corruption : read but sound occasionally we used to keep a pig
25% corruption : all rabbits add occasionally we use to keep a pig
```

```
original       : the chickens was looked after by my sisters
15% corruption : the chickens was looked after by my sisters
25% corruption : the chicken was looked after by my sisters
```

WSJ Sentences

```
original       : bell canada enterprises incorporated said it plans an
                 offering in europe of one hundred and fifty million dollars
                 canadian of notes
15% corruption : bell canada enterprises incorporated said it plans an
                 offering in europe of one he under a damn day fitting
25% corruption : bell canada enterprises incorporated shed its plans another in

original       : the five year ten percent notes were priced at one oh one
15% corruption : the five year done percent notes were priced at one oh one
25% corruption : the five year ten a cent notes were priced at one one

original       : lead underwriter is union bank of switzerland securities
                 limited proceeds will be used to refinance short term debt
15% corruption : lead underwriter is union bank of switzerland securities
                 limited proceeds will be way used to refinance short term
                 debt
25% corruption : old under i countries you on back of switzerland securities
                 limited proceeds quick be used to refinance short term that

original       : bell canada enterprises is a telecommunications energy
                 printing and real estate concern
15% corruption : able canada enterprises is a telecommunications them any
                 printing and really state concern
25% corruption : bell am a day a m enterprises is a telling communications
                 energy printing

original       : not surprisingly the davis zweig report has become more
                 bearish dropping to a twenty five percent bond position
                 around mid april
15% corruption : not surprisingly the davis zweig report has become more
                 bearish dropping to identify percent bond position around
                 industrial
25% corruption : not side price only the day his zweig report has become more
                 bearish dropping to a present if of percent by opposition
                 around made april

original       : yesterday it called for a complete move out of bonds and
                 into money market funds
15% corruption : yesterday it called for a complete out of bonds and in a
                 money market funds
25% corruption : yesterday it called for a complete out of bonds should into
                 money market funds

original       : meanwhile the bond market rallied sharply for the day
```

15% corruption : on meanwhile the bond more it rallied sharply for the day
25% corruption : meanwhile the bond market rallied chart like or the day

original : it also would bar foreign companies from becoming primary dealers in US government securities unless their governments give US companies the same right in their countries
15% corruption : it also would bar foreign companies from becoming primary dealers in you best government securities unless their governments give US kemp gives others summarise in their countries
25% corruption : it also would bar far a ??? US government you're into his a more less their governments give US companies the seemed white in their countries

original : it is aimed at japan
15% corruption : it is aimed at japan
25% corruption : it is and at japan

original : the federal reserve board recently accepted two japanese firms as primary dealers
15% corruption : the federal reserve go board recently accepted two japanese firms as primary dealers
25% corruption : the federal reserve kind recently accepted two japanese firms as primary dealers

original : dayton hudson fell one to fifty in active trading
15% corruption : dayton hudson fell one to fifty in active true being
25% corruption : dayton hudson fell what a fitting enough give trading

original : after the market closed the minnesota legislature passed an anti takeover bill aimed at thwarting dart group's interest in acquiring the minneapolis based retailer
15% corruption : after the market closed the minnesota legislature past a ninety takeover between at thwarting art group's interesting acquiring the minneapolis based retailer
25% corruption : after the market closed the minnesota legislature passed a man cut ever became pat thwarting doubt group's interest in acquiring the minneapolis best retailer

original : but it wasn't immediately clear if the bill would end takeover speculation about dayton hudson
15% corruption : but it wasn't immediately clear if the able would end takeover speculation about dayton hudson
25% corruption : at but it wasn't immediately clarify the bill would end takeover speculation about pay turned hudson

original : analysts said some traders in raw material markets continue to sell out their commodity positions to raise money to meet margin calls on their stock holdings
15% corruption : analysts set some traders in raw material markets continue to sell out their commodity positions to raise money to meet margin calls on air stock though old things
25% corruption : analysts sets somebody's in raw material markets continue to sell out air commodity positions to raise money to manage in calls on their stock holdings

original : as a result of the stock market's recent severe volatility brokerage houses have been demanding more cash or other collateral from investors who have bought stock with borrowed money
15% corruption : as a result of the stood market's recent severe volatility brokerage houses everybody making more cash or other collateral from investors who have bought stock with the borrowed money
25% corruption : as a result of the stock making have ??? bought stock with bottom something

original : the stock market however did undergo a rebound yesterday
15% corruption : the top market however did undergo a rebound yesterday
25% corruption : the stock market however did undergo american yesterday

original : mr ziegler said the company earned about thirty seven cents a share in the fourth quarter
15% corruption : mr ziegler said the company earned about off thirties even sets american the fourth quarter
25% corruption : miss are ziegler said the components about thirties even cents a share in the fourth quarter

original : in the year ago period the company earned two point three million dollars or thirty two cents a share
15% corruption : in the year ago period cannot company earned two point three million dollars or thirty two cents a share
25% corruption : in the year ago or could the come a thursday two set i gets a share

original : fourth quarter sales rose to about ninety million dollars from seventy two point two million dollars in the fourth quarter of nineteen eighty six
15% corruption : fourth quarter sales rose to about ninety million dollars from seventy two point two million to dollars in the fourth quarter of nineteen basics
25% corruption : fourth quite are sales rose to about ninety mid claim do

original : bearings incorporated authorized the purchase of as many as six hundred thousand shares or about twelve percent of its common stock
15% corruption : bear incorporated over eyes the purchase of as any as sixty hundred thousand shares or boy about twelve percentages common stock
25% corruption : bearings incorporated around the purchase of as many as six funds around of thousand shares or about twelve percent of you its common stock

original : the company said it may buy the shares in the open market or in negotiated transactions from time to time depending on market conditions
15% corruption : the company said it may buy the shares in the open market or in negotiated transactions frightening to time depending on market conditions
25% corruption : either company did may buy the shares in the open market or

in negotiated transactions from time to get i'm depending on mid conditions

original : shares acquired will be held for corporate purposes including benefit plans and stock option plans
15% corruption : shares acquired will be held or incorporate purposes including benefit plans and stock option plans
25% corruption : shares acquired while be hell feet corporate purposes us including benefit line and stock option plans

original : the banking concern hopes to complete the sale within two weeks the sources said
15% corruption : the banking concern hopes to complete the sale within two weeks the sources shed
25% corruption : the banking concern hopes to computers sale within two weeks these sources said

original : the transaction is expected to produce an estimated gain of one hundred and forty million to a hundred and fifty million dollars for the first quarter
15% corruption : the transactions expected to produce an estimated gain of one hundred and forty million to a hundred and fifty million dollars for they've first water
25% corruption : the transaction expected produce an estimated gain of one understand forty men to a hundred and fifty million dollars far the first quarter

original : the schwab unit has a book value of about seventy million dollars and bankamerica has made a capital loan of about fifty million dollars to the operation
15% corruption : others while unit as a book value of about seventeen alone dollars and bankamerica has made a capital low of about fifty million dollars together operation
25% corruption : the schwab unit has ago call you of a doubt seventy will until is and bankamerica high as makers capital line of about he forty alone dollars to the operation

original : neither bankamerica nor mr schwab would comment
15% corruption : neither bankamerica or mr schwab would comment
25% corruption : never up bankamerica near mr schwab would comment

original : it seems that few people have anything good to say about the recent budget compromise
15% corruption : it seems that few people of having nothing good to say about the recent budget compromise
25% corruption : it seems at few people have anything by good to say about the recent budget compromise

original : neither do i but it should be pointed out that the compromise is rather good by historical standards
15% corruption : either do i but should be pointed out that the compromise is rather good by historical standards
25% corruption : my either do i but it each should be pointed out at the compromise is rather good by historical standards

original : first keep in mind that the level of government spending is all that matters as far as our economy is concerned

15% corruption : first key in mind that the level of government found thing is although adding matters as far as our economy is concerned
25% corruption : first keep in mind that the level of government spending is all that matters as far as our economy is concerned

original : whether it is financed by taxes or by a deficit which is just postponed taxes is irrelevant
15% corruption : whether it is financed by taxes or by a deficit which is just postponed taxes is irrelevant
25% corruption : whether it is financed big taxes or by a deficit which is just postponed taxes irrelevant

original : thus our only concern should be to reduce government spending and if that can be achieved only by raising taxes simultaneously so be it
15% corruption : this either only concern child beat a reduce government spending adding fact can be achieved only by raising taxes simultaneously so be it
25% corruption : also our only up concerned beat a ripple you government heading and fifth that can be a t only by raise e taxes simultaneously so be it

original : but the new agreement would narrow the wage rise in the first year to thirty five cents an hour from the original fifty cents an hour
15% corruption : but the new agreement would narrow the wage reason the first year to thirty five cents an our if rather original fifty cents an our
25% corruption : but the new agreement would now rather way raise in the first year to thirty five cents a near from the worry join left extension our

original : second and third year wage increases would be tied to the consumer price index with a cap of thirty five cents an hour
15% corruption : as second and heard year wage increases would beat i'd together consumer price inadequate cap of thirty five cents an our
25% corruption : second add off third year way june classes would beat i'd to the consumer be price in does with a cap of thirty five sets a near

original : as a result over three years the wage increases would total about seven percent down from eight percent under last week's agreement
15% corruption : as a result over three years the increases would total a bought seven percent down from eight percent under last we agreement
25% corruption : as a result over e years the which increases looked little about seven percent down from eight percent under last you week's agreement

original : domestic revenue gained twenty percent to two zero seven point six one billion yen helped by japan's expanding economy
15% corruption : domestic revenue good twenty percent to two zero seven points talks one bill helped by japan's expanding economy

25% corruption : domestic revenue great went percent to two zero seven points six one billion new how helped by japan's expanding economy

original : international revenue rose five point nine percent to six hundred and forty one point thirty eight billion yen
15% corruption : international revenue rose five point into cent tasks hundred and forty one point thirty eight billion when
25% corruption : international reason you're as five point name a cent to six this hundred

original : the strong yen encouraged more japanese to travel abroad
15% corruption : the strong yen encouraged or japanese to travel abroad
25% corruption : the strong yen encouraged

original : consumer credit which grew at a robust ten point one percent annual rate in august is likely to show a slower growth pace for september
15% corruption : consumer credit which grew at a robust ten point one percent annual rate investors like later show a slow right growth patience for september
25% corruption : consumer credit which great across testing point one percent annual way turn august is likely to show a slower great for september

original : soft retail spending as evidenced by the recent chain store sales report plus somewhat lower auto sales may contribute to the credit decline
15% corruption : soft retail spending as evidenced by the recent chain store sales report plus somewhat lower auto sales may contribute to the credit decline
25% corruption : soft retail spending as evidenced by the recent chain stories report less somewhat low a white sales may contribute to the read it decline

original : the consensus calls for a four billion dollar increase in september compared with a gain of five point four billion dollars the previous month
15% corruption : the consensus calls for before billion dollar increase in september compared with a gain of five point or billion dollars the previous month
25% corruption : the consensus calls for a four billion during across in september compared with a gain of five point of fault along close the previous month

Bibliography

[Ainsworth, 1988] W. A. Ainsworth, *Speech Recognition by Machine*, volume 12 of *IEE Computing Series*, Peter Peregrinus Ltd., 1988.

[Crystal, 1987] D. Crystal, *The Cambridge Encyclopedia of Language*, Cambridge University Press, 1987.

[Holmes, 1988] J. N. Holmes, *Speech Synthesis and Recognition*, Aspects of Information Technology, Von Nostrand Reinhold (UK), 1988.

[Lea, 1980] W. A. Lea, editor, *Trends in Speech Recognition*, Prentice Hall Signal Processing Series, Prentice Hall, Inc., 1980.

[Lee, 1988] K.-F. Lee, *Large-Vocabulary Speaker-Independent Continuous Speech Recognition: The SPHINX System*, PhD thesis, Computer Science Department, Carnegie-Mellon University, April 1988.

[Luger and Stubblefield, 1993] G. Luger and W. Stubblefield, *Artificial Intelligence: Structures and Strategies for Complex Problem Solving*, Benjamin/Cummings, 1993.

[Rich and Knight, 1991] E. Rich and K. Knight, *Artificial Intelligence*, McGraw-Hill, 2nd edition, 1991.

[Waibel and Lee, 1990] A. Waibel and K.-F. Lee, *Readings in Speech Recognition*, Morgan Kaufmann, 1990.

References

[Ainsworth, 1988] W. A. Ainsworth, *Speech Recognition by Machine*, volume 12 of *IEE Computing Series*, Peter Peregrinus Ltd., 1988.

[Alleva *et al.*, 1992] F. Alleva, H. Hon, X. Huang, M. Hwang, R. Rosenfeld, and R. Weide, "Applying SPHINX-II to the DARPA Wall Street Journal CSR Task", in *Proceedings of the DARPA Speech and Natural Language Workshop*, pages 393–398, Morgan Kaufmann, 1992.

[Appelt and Jackson, 1992] D. E. Appelt and E. Jackson, "SRI International Results February 1992 ATIS Benchmark Test", in *Proceedings of the DARPA Speech and Natural Language Workshop*, pages 95–100, Morgan Kaufmann, February 1992.

[ARPA, 1994] *Proceedings of the ARPA Human Language Technology Workshop*, Morgan Kaufmann, 1994.

[Aubert *et al.*, 1994] X. Aubert, C. Dugast, R. Kneser, V. Steinbiss, S. Besling, and H. Ney, "The Philips Large Vocabulary CSR System", in *Proceedings of the ARPA Spoken Language Systems Technology Workshop*, Morgan Kaufmann, March 1994, Oral Presentation.

[Bäck *et al.*, 1991] T. Bäck, F. Hoffmeister, and H. Schwefel, "A Survey of Evolution Strategies", in *Proceedings of the Fourth In-*

ternational Conference on Genetic Algorithms, Morgan Kaufmann, 1991.

[Baggia and Rullent, 1993] P. Baggia and C. Rullent, "Partial Parsing as a Robust Parsing Strategy", in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, volume 2, pages 123–126, 1993, Minnesota.

[Baggia *et al.*, 1992] P. Baggia, E. Gerbino, E. Giachin, and C. Rullent, "Real-Time Linguistic Analysis for Continuous Speech Understanding", in *Proceedings of the 3rd Conference on Applied Natural Language Processing*, April 1992.

[Bahl *et al.*, 1989] L. R. Bahl, R. Bakis, J. Bellegarda, P. F. Brown, D. Burshtein, S. K. Das, P. V. de Souza, P. S. Gopalakrishnan, D. Kanevsky, R. L. Mercer, A. J. Nadas, D. Nahamoo, and M. A. Picheny, "Large Vocabulary Natural Language Continuous Speech Recognition", in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 465–467, 1989, Glasgow.

[Baker *et al.*, 1992] J. Baker, J. Baker, P. Bamberg, K. Bishop, L. Gillick, V. Helman, Z. Huang, Y. Ito, S. Lowe, B. Peskin, R. Roth, and F. Scattone, "Large Vocabulary Recognition of Wall Street Journal Sentences at Dragon Systems", in *Proceedings of the DARPA Speech and Natural Language Workshop*, pages 387–392, Morgan Kaufmann, 1992.

[Ballard and Jones, 1990] B. Ballard and M. Jones, "Computational Linguistics", in S. C. Shapiro, editor, *Encyclopedia of Artificial Intelligence*, volume 1, pages 133–151, Wiley, 1990.

[Barry *et al.*, 1989] W. J. Barry, J. Wells, and A. J. Fourcin, "SAMPA: Further Development", Technical Report SAM-UCL-003, ESPRIT Project 2589 (SAM) Multi-Lingual Speech Input/Output Assessment, Methodology and Standardisation, November 1989.

[Bates *et al.*, 1992] M. Bates, R. Bobrow, P. Fung, R. Ingria, F. Kubala, J. Makhoul, L. Nguyen, R. Schwartz, and D. Stallard, "Design and Performance of Harc, the BBN Spoken Language Understanding System", in *Proceedings of the International Conference on Spoken Language Processing (ICSLP)*, pages 241–244, October 1992, Alberta, Canada.

[Bates *et al.*, 1993] M. Bates, R. Bobrow, P. Fung, R. Ingria, F. Kubala, J. Makhoul, L. Nguyen, R. Schwartz, and D. Stallard, "The BBN/HARC Spoken Language Understanding System", in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, volume 2, pages 111–114, 1993, Minnesota.

[Bear *et al.*, 1993] J. Bear, J. Dowding, E. Shriberg, and P. Price, "A System for Labeling Self-Repairs in Speech", Technical Report 522, SRI International, February 1993.

[Beardon, 1989] C. Beardon, *Artificial Intelligence Terminology: A Reference Guide*, Ellis Horwood, 1989.

[Bernstein and Danielson, 1992] J. Bernstein and D. Danielson, "Spontaneous Speech Collection for the CSR Corpus", in *Proceedings of the DARPA Speech and Natural Language Workshop*, pages 373–378, Morgan Kaufmann, February 1992.

[Blomberg, 1989] M. Blomberg, "Synthetic Phoneme Prototypes In A Connected-Word Speech Recognition System", in *Pro-*

ceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 687–690, 1989, Glasgow.

[Bobrow *et al.*, 1992] R. Bobrow, R. Ingria, and D. Stallard, "Syntactic/Semantic Coupling in the BBN Delphi System", in *Proceedings of the DARPA Speech and Natural Language Workshop*, pages 311–315, Morgan Kaufmann, February 1992.

[Bocchieri, 1994] E. Bocchieri, "The ATT ATIS System", in *Proceedings of the ARPA Spoken Language Systems Technology Workshop*, Morgan Kaufmann, March 1994, Oral Presentation.

[Bormuth, 1966] J. R. Bormuth, "Readability: A New Approach", *Reading Research Quarterly*, 1(3):79–132, 1966.

[Bridle *et al.*, 1982] J. S. Bridle, M. D. Brown, and R. M. Chamberlain, "An Algorithm for Connected Word Recognition", in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 899–902, 1982, Paris.

[Browning *et al.*, 1990] S. R. Browning, R. K. Moore, K. M. Ponting, and M. J. Russell, "A Phonetically Motivated Analysis of the Performance of the ARM Continuous Speech Recognition System", in *Proceedings of the Institute of Acoustics Speech and Hearing Conference*, November 1990, Windermere.

[Browning *et al.*, 1991] S. R. Browning, J. McQuillan, M. J. Russell, and M. J. Tomlinson, "Texts of Materials Recorded in the SI89 Speech Corpus", Technical Report 142, DRA Speech Research Unit, 1991.

[Calmet and Campbell, 1993] J. Calmet and J. A. Campbell, "Artificial Intelligence and Symbolic Mathematical Computations", in *Proceedings of the Conference on Artificial Intelligence and Symbolic Mathematical Computations*, volume 737 of *Lecture Notes in Computer Science*, pages 1–19, Springer-Verlag, 1993.

[Clery, 1989] D. Clery, "Scottish Software may run Voice-Controlled Computer", *New Scientist*, page 34, 17th March 1989.

[Collingham and Garigliano, 1992] R. J. Collingham and R. Garigliano, "A Word Lattice Parsing Algorithm for Naturally Spoken English", in *Proceedings of the 4th Australian International Conference on Speech Science and Technology*, December 1992, Brisbane.

[Collingham and Garigliano, 1993] R. J. Collingham and R. Garigliano, "Using Anti-Grammar and Semantic Categories for the Recognition of Spontaneous Speech", in *Proceedings of Eurospeech, the 3rd European Conference on Speech Communication and Technology*, ESCA, September 1993, Berlin.

[Compton, 1947] J. Compton, editor, *Spoken English: It's Practice in Schools and Training Colleges*, Methuen & Co. Ltd., London, second edition, 1947.

[Cutting *et al.*, 1992] D. Cutting, J. Kupiec, J. Pedersen, and P. Sibun, "A Practical Part-of-Speech Tagger", in *Proceedings of the 3rd Conference on Applied Natural Language Processing*, April 1992.

[Dahl *et al.*, 1994] D. Dahl, M. Linebarger, N. Nguyen, and L. Norton, "Unisys Activities in Spoken Language Understanding", in *Proceedings of the ARPA Spoken Language Systems*

Technology Workshop, Morgan Kaufmann, March 1994, Oral Presentation.

[Digalakis *et al.*, 1994] V. Digalakis, H. Murveit, P. Monaco, H. Bratt, J. Butzberger, and M. Weintraub, "SRI November 1993 CSR Hub Evaluation", in *Proceedings of the ARPA Spoken Language Systems Technology Workshop*, Morgan Kaufmann, March 1994, Oral Presentation.

[Downey and Russell, 1992] S. N. Downey and M. J. Russell, "A Decision Tree Approach to Task Independent Speech Recognition", in *Proceedings of the Institute of Acoustics Speech and Hearing Conference*, November 1992, Windermere.

[Downey, 1993] S. N. Downey, "Experiments in Sub-Phonemic Hidden Markov Modelling", Technical Report Memorandum 4732, Speech Research Unit, DRA Malvern, St. Andrew's Road, Malvern, Worcs. WR14 3PS, February 1993.

[EC, 1991] "Linguistic Research and Engineering in the Framework Programme 1990–1994: Technical Background Document", European Community, July 1991.

[Engelmore and Morgan, 1988] R. Engelmore and A. Morgan, editors, *Blackboard Systems*, The Insight Series in Artificial Intelligence, Addison Wesley, 1988.

[Fallside, 1989] F. Fallside, "Progress in Large Vocabulary Speech Recognition", *Speech Technology*, 4(4):14–15, April/May 1989.

[Fogel *et al.*, 1966] L. J. Fogel, A. J. Owens, and M. J. Walsh, *Artificial Intelligence through Simulated Evolution*, J. Wiley, New York, 1966.

[Fogel, 1992] D. B. Fogel, *Evolving Artificial Intelligence*, PhD thesis, University of California, San Diego, USA, 1992.

[Francis and Kucera, 1979] W. N. Francis and H. Kucera, *Manual of Information to Accompany a Standard Corpus of Present-Day Edited American English, for Use with Digital Computers*, 1979.

[Garigliano and Nettleton, 1994] R. Garigliano and D. J. Nettleton, "The Interplay of Symbolic and Adaptive Techniques: Two Case Studies", in *Proceedings of the IEE Colloquium on Symbolic and Neural Cognitive Engineering*, 1994, London.

[Garigliano *et al.*, 1993a] R. Garigliano, R. G. Morgan, and M. H. Smith, "The Lolita System as a Contents Scanning Tool", in *Proceedings of the 13th International Conference on Artificial Intelligence, Expert Systems and Natural Language Processing*, Avignon, May 1993.

[Garigliano *et al.*, 1993b] R. Garigliano, K. Johnson, and R. J. Collingham, "A Data-Supported Case for a Spontaneous Speech Grammar", in *Proceedings of Eurospeech, the 3rd European Conference on Speech Communication and Technology*, ESCA, September 1993, Berlin.

[Gauvain *et al.*, 1994a] J. Gauvain, L. Lamel, G. Adda, and M. Adda-Decker, "The LIMSI Continuous Speech Dictation System: Evaluation on the ARPA Wall Street Journal Task", in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 1994, Adelaide.

[Gauvain *et al.*, 1994b] J. Gauvain, L. Lamel, G. Adda, and M. Adda-Decker, "The LIMSI Nov93 WSJ System", in *Proceedings of the ARPA Spoken Language Systems Technology Workshop*, Morgan Kaufmann, March 1994, Oral Presentation.

[Giachin *et al.*, 1990] E. P. Giachin, A. E. Rosenberg, and C.-H. Lee, "Word Juncture Modeling using Phonological Rules for HMM-Based Continuous Speech Recognition", in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 737–740, 1990, Albuquerque, New Mexico.

[Goldberg, 1989] D. E. Goldberg, *Genetic Algorithms in Search, Optimization and Machine Learning*, Addison-Wesley, 1989.

[Guzy and Edmonds, 1986] J. J. Guzy and E. A. Edmonds, "Definitely Not Pattern Matching: A Method In Automatic Speech Recognition", in *Proceedings of the Institute of Acoustics*, volume 8, pages 425–442, 1986.

[Hardie, 1992] R. G. Hardie, *Collins Pocket English Grammar*, HarperCollins Publishers, 1992.

[Harkins, 1988] J. E. Harkins, "Speech Recognition for Communication between Deaf and Hearing People", in *Proceedings of Speech Tech'88*, pages 268–270, Media Dimensions Inc., New York, 1988.

[Hatazaki *et al.*, 1989] K. Hatazaki, Y. Komori, T. Kawabata, and K. Shikano, "Phoneme Segmentation using Spectrogram Reading Knowledge", in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 393–396, 1989, Glasgow.

[Hirschman *et al.*, 1991] L. Hirschman, S. Seneff, D. Goodine, and M. Phillips, "Integrating Syntax and Semantics into Spoken Language Understanding", in *Proceedings of the DARPA Speech and Natural Language Workshop*, pages 366–371, Morgan Kaufmann, February 1991.

[Hirschman *et al.*, 1993] L. Hirschman, M. Bates, D. Dahl, W. Fisher, J. Garofolo, D. Pallett, K. Hunicke-Smith, P. Price, A. Rudnicky, and E. Tzoukermann, "Multi-Site Data Collection and Evaluation in Spoken Language Understanding", in *Proceedings of the DARPA Human Language Technology Workshop*, Morgan Kaufmann, March 1993.

[Hirst, 1987] G. Hirst, *Semantic Interpretation and the Resolution of Ambiguity*, Cambridge University Press, 1987.

[Hochberg *et al.*, 1994] M. Hochberg, A. Robinson, and S. Renals, "ABBOTT: The CUED Hybrid Connectionist-HMM Large Vocabulary Recognition System", in *Proceedings of the ARPA Spoken Language Systems Technology Workshop*, Morgan Kaufmann, March 1994, Oral Presentation.

[Holland, 1975] J. H. Holland, *Adaption in Natural and Artificial Systems*, University of Michingan Press, 1975.

[Hon and Lee, 1990] H.-W. Hon and K.-F. Lee, "On Vocabulary Independent Speech Modelling", in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 725–728, 1990, Albuquerque, New Mexico.

[Hon and Lee, 1991] H.-W. Hon and K.-F. Lee, "Recent Progress in Robust Vocabulary Independent Speech Recognition", in *Proceedings of the DARPA Speech and Natural Language Workshop*, pages 258–263, Morgan Kaufmann, February 1991.

[Huang *et al.*, 1993] X. Huang, F. Alleva, H.-W. Hon, M.-Y. Hwang, K.-F. Lee, and R. Rosenfeld, "The SPHINX-II Speech Recognition System: An Overview", *Computer Speech and Language*, 2:137–148, 1993.

[Hunt, 1988] M. J. Hunt, "Evaluating the Performance of Connected-Word Speech Recognition Systems", in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 457–460, 1988, New York.

[Isaar and Ward, 1994] S. Isaar and W. Ward, "Flexible Parsing: CMU's Approach to Spoken Language Understanding", in *Proceedings of the ARPA Spoken Language Systems Technology Workshop*, Morgan Kaufmann, March 1994, Oral Presentation.

[Johansson *et al.*, 1978] S. Johansson, G. Leech, and H. Goodluck, *Manual of Information to Accompany the Lancaster-Oslo/Bergen Corpus of British English, for Use with Digital Computers*, 1978.

[Johnson *et al.*, 1994a] K. Johnson, R. J. Collingham, and R. Garigliano, "Data-Supported Case for the Extended Coverage of Repairs in the Recognition of Natural Speech", in *Proceedings of the Institute of Acoustics Autumn Conference : Speech and Hearing*, November 1994, Windermere.

[Johnson *et al.*, 1994b] K. Johnson, R. Garigliano, and R. J. Collingham, "Data-Based Control of the Search Space Generated by Multiple Knowledge Bases for Speech Recognition", in *Proceedings of the International Conference on Spoken Language Processing (ICSLP)*, September 1994, Yokohama, Japan.

[Johnson *et al.*, 1994c] K. Johnson, R. Garigliano, and R. J. Collingham, "The Effect of Repair on Speech Recognition Performance", in *Submitted to the 4th Conference on Applied Natural Language Processing*, October 1994, Stuttgart, Germany.

[Johnson, to appear 1995] K. Johnson, *High Level Knowledge within Speech Recognition*, PhD thesis, University of Durham, (to appear) 1995.

[Karlsson *et al.*, 1995] F. Karlsson, A. Voutilainen, J. Heikkila, and A. Antilla, editors, *Constraint Grammer: A Language Independent System for Parsing Unrestricted Text*, Mouton de Gruyter, 1995.

[Klatt, 1977] D. H. Klatt, "Review of the ARPA Speech Understanding Project", *Journal of the Acoustical Society of America*, 62(6):1345–1366, December 1977.

[Kubala *et al.*, 1992] F. Kubala, C. Barry, M. Bates, R. Bobrow, P. Fung, R. Ingria, J. Makhoul, L. Nguyen, R. Schwartz, and D. Stallard, "BBN Byblos and Harc February 1992 ATIS Benchmark Results", in *Proceedings of the DARPA Speech and Natural Language Workshop*, pages 72–77, Morgan Kaufmann, February 1992.

[Lamel *et al.*, 1986] L. F. Lamel, R. H. Kassel, and S. Seneff, "Speech Database Development: Design and Analysis of the Acoustic-Phonetic Corpus", in *Proceedings of the DARPA Speech Recognition Workshop*, pages 100–109, Morgan Kaufmann, February 1986.

[Lamel, 1993] L. F. Lamel, "A Knowledge-Based System for Stop Consonant Identification Based on Speech Spectrogram Reading", *Computer Speech and Language*, 2:169–191, 1993.

[Lea, 1980] W. A. Lea, editor, *Trends in Speech Recognition*, Prentice Hall Signal Processing Series, Prentice Hall, Inc., 1980.

[Lee *et al.*, 1989a] C.-H. Lee, B.-H. Juang, F. K. Soong, and L. R. Rabiner, "Word Recognition using Whole Word And Subword Models", in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 683–686, 1989, Glasgow.

[Lee *et al.*, 1989b] K.-F. Lee, H.-W. Hon, M.-Y. Hwang, S. Mahajan, and R. Reddy, "The Sphinx Speech Recognition System", in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 445–448, 1989, Glasgow.

[Lee *et al.*, 1990] K.-F. Lee, H.-W. Hon, and R. Reddy, "An Overview of the Sphinx Speech Recognition System", *IEEE Transactions on Acoustics, Speech and Signal Processing*, 38(1):35–45, January 1990.

[Lee, 1988] K.-F. Lee, *Large-Vocabulary Speaker-Independent Continuous Speech Recognition: The SPHINX System*, PhD thesis, Computer Science Department, Carnegie-Mellon University, April 1988.

[Linggard, 1988] R. Linggard, "Language Processing Beyond Speech Recognition", *British Telecom Technology Journal*, 6(2):289–305, April 1988.

[Ljolje and Riley, 1992] A. Ljolje and M. Riley, "Optimal Speech Recognition using Phone Recognition and Lexical Access", in *Proceedings of the International Conference on Spoken Language Processing (ICSLP)*, pages 313–316, October 1992, Alberta, Canada.

[Mackey, 1989] A. Mackey, "Centre Gives Deaf Students Some High-Tech Help", *Los Angeles Times*, 1989, Sunday, September 24th.

[Makhoul, 1989] J. Makhoul, "Towards Spoken Language Language Systems", *Speech Technology*, 4(4):34–46, April/May 1989.

[Marshall, 1987] I. Marshall, "Tag Selection Using Probabilistic Methods", in R. Garside, G. Leech, and G. Sampson, editors, *The Computational Analysis of English: A Corpus-Based Approach*, chapter 4, pages 42–56, Longman, 1987.

[Miller, 1990] L. D. Miller, "Classroom Captioning or RTGD", TCN Newsletter, published by RapidText Inc., 18013 Sky Park Circle, Irvine, CA 92714, U.S.A., January 1990.

[Mitton, 1992] R. Mitton, *A Description of a Computer-Usable Dictionary File Based on the Oxford Advanced Learner's Dictionary of Current English*, June 1992.

[Moore *et al.*, 1994] R. Moore, M. Cohen, V. Abrash, D. Appelt, H. Bratt, J. Butzberger, J. Dowding, H. Franco, and D. Moran, "SRI's Recent Progress on the ATIS Task", in *Proceedings of the ARPA Spoken Language Systems Technology Workshop*, Morgan Kaufmann, March 1994, Oral Presentation.

[Morgan *et al.*, 1994] N. Morgan, G. Tajchman, N. Mirghafori, Y. Konig, E. Fosler, and C. Wooters, "Scaling a Hyprid HMM/MLP System for Large Vocabulary CSR", in *Proceedings of the ARPA Spoken Language Systems Technology Workshop*, Morgan Kaufmann, March 1994, Oral Presentation.

[Murveit *et al.*, 1993a] H. Murveit, J. Butzberger, V. Digalakis, and M. Weintraub, "Large Vocabulary Dictation Using SRI's DECIPHER(TM) Speech Recognition System: Progressive-Search Techniques", in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Pro-*

cessing (ICASSP), volume 2, pages 319–322, 1993, Minnesota.

[Murveit *et al.*, 1993b] H. Murveit, J. Butzberger, V. Digalakis, and M. Weintraub, "Large Vocabulary Dictation using SRI's Decipher Speech Recognition System: Progressive Search Techniques", in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, volume 2, pages 319–322, 1993, Minnesota.

[Myers and Rabiner, 1981] C. S. Myers and L. R. Rabiner, "A Level Building Dynamic Time Warping Algorithm for Connected Word Recognition", *IEEE Transactions on Acoustics, Speech and Signal Processing*, 29(2):284–297, April 1981.

[Nettleton and Collingham, 1995] D. J. Nettleton and R. J. Collingham, "Word Lattice Optimisation using Evolutionary Algorithms", *to be submitted to the Journal of Natural Language Engineering*, 1995.

[Nettleton and Garigliano, 1994] D. J. Nettleton and R. Garigliano, "Evolutionary Algorithms for Dialogue Optimisation in the LOLITA Natural Language Processor", in V. Rayward-Smith, editor, *Adaptive Computing: Genetic Algorithms*, 1994, to appear.

[Newell and Brooks, 1985] A. F. Newell and C. P. Brooks, "Verbatim Transcript of Speech by Possum Palantype", 1985, University of Dundee.

[Newell and Simon, 1976] A. Newell and H. A. Simon, "Computer Science as Empirical Inquiry: Symbols and Search", *Communications of the ACM*, 19(3):113–126, 1976.

[Newell *et al.*, 1988] A. F. Newell, J. L. Arnott, R. Dye, and A. Y. Cairns, "A Full-Speed Listening Typewriter Simulation", *International Journal of Man-Machine Studies*, 1988.

[Normandin *et al.*, 1994] Y. Normandin, D. Bowness, R. Cardin, Y. Chen, R. D. Mori, C. Drouin, D. Goupil, R. Kuhn, A. Lazarides, and E. Millien, "CRIM's December 1993 ATIS System", in *Proceedings of the ARPA Spoken Language Systems Technology Workshop*, Morgan Kaufmann, March 1994.

[O'Shaughnessy, 1992] D. O'Shaughnessy, "Analysis of False Starts in Spontaneous Speech", in *Proceedings of the International Conference on Spoken Language Processing*, pages 931–934, 1992, Alberta, Canada.

[Ostendorf *et al.*, 1994] M. Ostendorf, F. Richardson, S. Tibrewal, R. Iyer, O. Kimball, and J. Rohlicek, "Stochastic Segment Modeling for Continuous Speech Recognition", in *Proceedings of the ARPA Spoken Language Systems Technology Workshop*, Morgan Kaufmann, March 1994, Oral Presentation.

[Ovum, 1991] "Natural Language Markets: Commercial Strategies", Ovum Ltd., Rathbone Street, London, 1991.

[Pallett *et al.*, 1992] D. S. Pallett, N. L. Dahlgren, J. G. Fiscus, W. M. Fisher, J. S. Garofolo, and B. C. Tjaden, "DARPA February 1992 ATIS Benchmark Test Results", in *Proceedings of the DARPA Speech and Natural Language Workshop*, pages 15–27, Morgan Kaufmann, February 1992.

[Pallett, 1985] D. S. Pallett, "Performance Assessment of Automatic Speech Recognisers", *Journal of Research of the National Bureau of Standards*, 90(5):371–387, September-October 1985.

[Pallett, 1991] D. S. Pallett, "Session 2: DARPA Resource Management and ATIS Benchmark Test Poster Session", in *Proceedings of the DARPA Speech and Natural Language Workshop*, pages 49–57, Morgan Kaufmann, February 1991.

[Parry, 1990] S. Parry, "Statistics Ease Voice Translation", *New Electronics*, pages 65–68, April 1990.

[Paul and Baker, 1992] D. B. Paul and J. M. Baker, "The Design for the Wall Street Journal-based CSR Corpus", in *Proceedings of the International Conference on Spoken Language Processing (ICSLP)*, pages 899–902, October 1992, Alberta, Canada.

[Paul and Necioglu, 1993] D. B. Paul and B. F. Necioglu, "The Lincoln Large Vocabulary Stack Decoder HMM CSR", in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, volume 2, pages 660–663, 1993, Minnesota.

[Paul, 1992] D. B. Paul, "An Efficient A* Stack Decoder Algorithm for Continuous Speech Recognition with a Stochastic Language Model", in *Proceedings of the DARPA Speech and Natural Language Workshop*, pages 405–409, Morgan Kaufmann, February 1992.

[Paul, 1994] D. B. Paul, "The Lincoln Large Vocabulary Stack Decoder-Based HMM CSR", in *Proceedings of the ARPA Spoken Language Systems Technology Workshop*, Morgan Kaufmann, March 1994, Oral Presentation.

[Phillips *et al.*, 1992] M. Phillips, J. Glass, J. Polifroni, and V. Zue, "Collection and Analyses of WSJ-CSR Corpus at MIT", in *Proceedings of the International Conference on Spoken*

Language Processing (ICSLP), pages 907–910, October 1992, Alberta, Canada.

[Ponting and Russell, 1989] K. M. Ponting and M. J. Russell, "The ARM Project: Automatic Recognition of Spoken Airborne Reconnaissance Reports", Pattern Processing and Machine Intelligence Division, SP4 Research Note 86, The Royal Signals and Radar Establishment, Speech Research Unit, Malvern, Worcestershire, September 1989.

[Price *et al.*, 1988] P. Price, W. M. Fisher, J. Bernstein, and D. S. Pallett, "The DARPA 1000-Word Resource Management Database for Continuous Speech Recognition", in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 651–654, 1988, New York.

[Price, 1990] P. Price, "Evaluation of Spoken Language Systems: The ATIS Domain", in *Proceedings of the DARPA Speech and Natural Language Workshop*, pages 91–95, Morgan Kaufmann, June 1990.

[Rabiner and Juang, 1993] L. Rabiner and B.-H. Juang, *Fundamentals of Speech Recognition*, Prentice Hall, 1993.

[Rabiner *et al.*, 1989] L. R. Rabiner, C. H. Lee, B. H. Juang, and J. G. Wilpon, "HMM Clustering for Connected Word Recognition", in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 405–408, 1989, Glasgow.

[Rabiner, 1989] L. R. Rabiner, "A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition", *Proceedings of the IEEE*, 77(2):257–286, February 1989.

[RNID, 1990] The Royal National Institute for the Deaf, *Machine Transcription of Speech in Real Time*, Third RNID Symposium, June 1990, King's Fund Centre, London.

[Robinson *et al.*, 1994] A. Robinson, M. Hochberg, and S. Renals, "IPA: Improved Phone Modelling with Recurrent Neural Networks", in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 1994, Adelaide.

[Robinson, 1981] C. G. Robinson, "Cloze Procedure: a Review", *Educational Research*, 23(2):128–133, February 1981.

[Robinson, 1992] A. J. Robinson, "Recurrent Nets for Phone Probability Estimation", in *Proceedings of the DARPA Speech and Natural Language Workshop*, Morgan Kaufmann, 1992.

[Roth *et al.*, 1993] R. Roth, J. Baker, J. Baker, L. Gillick, M. Hunt, Y. Ito, S. Lowe, J. Orloff, B. Peskin, and F. Scattone, "Large Vocabulary Continuous Speech Recognition of Wall Street Journal Data", in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, volume 2, pages 640–643, 1993, Minnesota.

[Russell *et al.*, 1990a] M. J. Russell, K. M. Ponting, and S. M. Peeling, "The Armada Speech Recognition System", in *Proceedings of Voice Systems Worldwide*, 1990.

[Russell *et al.*, 1990b] M. J. Russell, K. M. Ponting, S. M. Peeling, S. R. Browning, J. S. Bridle, R. K. Moore, I. Galiano, and P. Howell, "The ARM Continuous Speech Recognition System", in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 69–72, 1990, Albuquerque, New Mexico.

[Russell, 1992a] M. J. Russell, "The Development of the Speaker Independent ARM Continuous Speech Recognition System", Technical report, DRA Speech Research Unit, 1992, Memorandum 4473.

[Russell, 1992b] M. J. Russell, "The Development of the Speaker Independent ARM Speech Recognition System", in *Proceedings of the Institute of Acoustics Speech and Hearing Conference*, November 1992, Windermere.

[Sagayama, 1989] S. Sagayama, "Phoneme Environment Clustering for Speech Recognition", in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 397–400, 1989, Glasgow.

[Sakoe, 1979] H. Sakoe, "Two-Level DP-Matching — A Dynamic Programming-Based Pattern Matching Algorithm for Connected Word Recognition", *IEEE Transactions on Acoustics, Speech and Signal Processing*, 27(6):588–595, December 1979.

[Sampson, 1987] G. Sampson, "Probabilistic Models of Analysis", in R. Garside, G. Leech, and G. Sampson, editors, *The Computational Analysis of English: A Corpus-Based Approach*, chapter 2, pages 16–29, Longman, 1987.

[Scattone *et al.*, 1994] F. Scattone, J. Baker, L. Gillick, J. Orloff, and R. Roth, "Dragon's Large Vocabulary Speech Recognition System", in *Proceedings of the ARPA Spoken Language Systems Technology Workshop*, Morgan Kaufmann, March 1994, Oral Presentation.

[Schwartz *et al.*, 1985] R. Schwartz, Y. Chow, O. Kimball, S. Roucos, M. Krasner, and J. Makhoul, "Context-Dependent Modelling for Acoustic-Phonetic Recognition of Continuous Speech",

in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1205–1208, 1985, Tampa.

[Seneff, 1989] S. Seneff, "Tina: A Probabalistic Syntactic Parser for Speech Understanding Systems", in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 711–714, 1989, Glasgow.

[Seneff, 1992] S. Seneff, "A Relaxation Method for Understanding Spontaneous Speech Utterances", in *Proceedings of the DARPA Speech and Natural Language Workshop*, pages 299–304, Morgan Kaufmann, 1992.

[Short *et al.*, 1994a] S. Short, R. J. Collingham, and R. Garigliano, "Making Use of Semantics in an Automatic Speech Recognition System", in *Proceedings of the Institute of Acoustics Autumn Conference on Speech and Hearing*, November 1994, Windermere, England.

[Short *et al.*, 1994b] S. Short, R. J. Collingham, and R. Garigliano, "'What did I say...?' — Using Meaning to Assess Speech Recognisers", in *Proceedings of the Institute of Acoustics Autumn Conference on Speech and Hearing*, November 1994, Windermere, England.

[Silverman and Morgan, 1990] H. F. Silverman and D. P. Morgan, "The Application of Dynamic Programming to Connected Speech Recognition", *IEEE ASSP Magazine*, pages 6–25, July 1990.

[Sinclair, 1990] J. Sinclair, editor, *Collins Cobuild English Grammar*, HarperCollins Publishers, 1990.

[Sondhi and Levinson, 1978] M. M. Sondhi and S. E. Levinson, "Computing Relative Redundancy To Measure Grammatical Constraint In Speech Recognition Tasks", in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 409–412, 1978.

[Souter and Atwell, 1994] C. Souter and E. Atwell, "Using Parsed Corpora: A Review of Current Practice", in N. Oostdjik and P. de Haan, editors, *Corpus-Based Research into Language*, pages 143–158, Rodopi, 1994.

[Spitz, 1991] J. Spitz, "Collection and Analysis of Data from Real Users: Implications for Speech Recognition/Understanding Systems", in *Proceedings of the DARPA Speech and Natural Language Workshop*, pages 164–169, Morgan Kaufmann, February 1991.

[Stallard and Bobrow, 1992] D. Stallard and R. Bobrow, "Fragment Processing in the DELPHI System", in *Proceedings of the DARPA Speech and Natural Language Workshop*, pages 305–310, Morgan Kaufmann, 1992.

[Stallard, 1994] D. Stallard, "Recent Work in Spoken Language Understanding in the BBN SLS Project", in *Proceedings of the ARPA Spoken Language Systems Technology Workshop*, Morgan Kaufmann, March 1994, Oral Presentation.

[Sutherland *et al.*, 1989] A. Sutherland, Y. Ariki, and M. A. Jack, "Osprey: A Continuous Speech Recognition System Based on Transputer Parallel Processing", Technical report, The Centre for Speech Technology Research, University of Edinburgh, 1989.

[Sutherland *et al.*, 1990] A. M. Sutherland, M. Campbell, Y. Ariki, and M. A. Jack, "Osprey: A Transputer Based Continuous Speech

Recognition System", in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 949–952, 1990, Albuquerque, New Mexico.

[Svartvik, 1992] J. Svartvik, *The London-Lund Corpus of Spoken English: Users' Manual*, 1992, Distributed by the Norwegian Computing Centre for the Humanities.

[Svendsen *et al.*, 1989] T. Svendsen, K. K. Paliwal, E. Harborg, and P. O. Husøy, "An Improved Sub-Word Based Speech Recogniser", in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 108–111, 1989, Glasgow.

[Taylor and Knowles, 1988] L. J. Taylor and G. Knowles, *Manual of Information to Accompany the SEC Corpus: The Machine-Readable Corpus of Spoken English*, January 1988.

[Taylor *et al.*, 1991] L. Taylor, G. Leech, and S. Fligelstone, "A Survey of English Machine-Readable Corpora", in S. Johansson and A.-B. Stenstrom, editors, *English Computer Corpora*, pages 319–354, Mouton de Gruyter, 1991.

[Taylor, 1953] W. L. Taylor, "Cloze Procedure: A New Tool For Measuring Readability", *Journalism Quarterly*, 30:415–433, 1953.

[Waibel, 1988] A. Waibel, *Prosody and Speech Recognition*, Research Notes in Artificial Intelligence, Pitman, London, 1988.

[Ward *et al.*, 1992] W. Ward, S. Isaar, X. Huang, H.-W. Hon, M.-Y. Hwang, S. Young, M. Matessa, F.-H. Liu, and R. Stern, "Speech Understanding in Open Tasks", in *Proceedings of the DARPA Speech and Natural Language Workshop*, pages 78–83, Morgan Kaufmann, 1992.

[Ward, 1991a] W. Ward, "Evaluation of the CMU ATIS System", in *Proceedings of the DARPA Speech and Natural Language Workshop*, Morgan Kaufmann, February 1991.

[Ward, 1991b] W. Ward, "Understanding Spontaneous Speech: The Phoenix System", in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 365–367, 1991, Toronto.

[Weiss and Hassett, 1991] N. A. Weiss and M. J. Hassett, *Introductory Statistics*, Addison Wesley, third edition, 1991.

[Winston, 1992] P. H. Winston, *Artificial Intelligence*, Addison Wesley, third edition, 1992, ISBN 0-201-53377-4.

[Woodland *et al.*, 1994a] P. Woodland, J. Odell, V. Valtchev, and S. J. Young, "Large Vocabulary Continuous Speech Recognition Using HTK", in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 1994, Adelaide.

[Woodland *et al.*, 1994b] P. Woodland, J. Odell, V. Valtchev, and S. J. Young, "The HTK Large Vocabulary Recognition System: An Overview", in *Proceedings of the ARPA Spoken Language Systems Technology Workshop*, Morgan Kaufmann, March 1994, Oral Presentation.

[Wright, 1989] R. D. Wright, "Automatic Speech Recognition: What does it Offer Deaf People?", Research Report 7, The Royal National Institute for the Deaf, Research Group, 105 Gower Street, London WC1E 6AH, September 1989.

[Young *et al.*, 1989] S. R. Young, A. G. Hauptmann, W. H. Ward, E. T. Smith, and P. Werner, "High-Level Knowledge Sources In Usable Speech Recognition Systems", *Communications of the ACM*, 32(2):289–305, February 1989.

[Zavaliagkos *et al.*, 1994] G. Zavaliagkos, T. Anastasakos, G. Chou, F. Kubala, C. Lapre, J. Makhoul, L. Nguyen, R. Schwarz, and Y. ZhaoYintdScwa, "BBN Hub System and Results", in *Proceedings of the ARPA Spoken Language Systems Technology Workshop*, Morgan Kaufmann, March 1994, Oral Presentation.

[Zue *et al.*, 1990] V. Zue, J. Glass, D. Goodine, M. Phillips, and S. Seneff, "The Summit Speech Recognition System: Phonological Modelling and Lexical Access", in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 49–52, 1990, Albuquerque, New Mexico.

[Zue *et al.*, 1992] V. Zue, J. Glass, D. Goddeau, D. Goodine, L. Hirschman, M. Phillips, J. Polifroni, and S. Seneff, "The MIT ATIS System: February 1992 Progress Report", in *Proceedings of the DARPA Speech and Natural Language Workshop*, pages 84–88, Morgan Kaufmann, 1992.

[Zue, 1985] V. W. Zue, "The Use of Speech Knowledge in Automatic Speech Recognition", *Proceedings of the IEEE*, 73(11):1602–1615, November 1985.