EECS225d Audio Signal Processing in Humans and Machines **Spring 1995**

Lecture 6: January 30, 1995

Lecturer: Prof. Nelson Morgan *Scribe: Eric Fosler*

6.1 Lecture overview

This lecture concluded Professor Morgan's overview of the history of speech recognition, focusing on the 80's to the present, and also delved into some insights on the problems and issues of automatic speech recognition. Topics included:

- The 80's
  - Corpora
  - Front Ends
  - Hidden Markov Models for CSR
  - 2nd (D)ARPA project
  - Neural Networks
- Why study ASR?
- Why is ASR hard?
- ASR dimensions

6.2 A little bit of history: the 80's

The 80's did not see many advances in speech recognition science from the 60's and 70's; most efforts were concentrated in scaling techniques to new domains. That is not to say that there were *no* advances; for instance, new front-end processing techniques were developed in this time period. In addition, several techniques, including recognition with Hidden Markov Models, were extended from isolated words to contuous speech. For the most part, however, the structure of speech recognition systems did not change; they were just trained on a larger quantity of data.

6.2.1 Large Corpora Collection

Prior to 1986 or so, the speech recognition community did not have any widely-accepted common databases for training recognition systems. This made comparisons between labs difficult, since nobody trained or tested on the same acoustic data. This holds true for the pattern recognition and handwriting recognition communities to this day.

In 1986, collection began on the TIMIT corpus[NIST90], which was to become the first widely-used standard corpus. A 61-phone alphabet was chosen to represent phonetic distinctions. The sentences in TIMIT were chosen to be phonetically balanced, meaning that a sizable representation of each phone could be found across the entire training set. The data was recorded at Texas Instruments, and phonetically segmented at MIT using first an automatic segmenter, followed by manual inspection and repair of the alignments by graduate students. This resulted in a database where for every phone uttered by a speaker, the time boundaries of the phone in the speech signal are marked. Even though errors still exist in the TIMIT database, it still remains one of the largest and most widely used hand-labeled phonetic corpora for training.

With the advent of the 2nd major ARPA speech program, a new task, called "Resource Management" was defined, with a new database[PFBP88] of speech. The task was surprisingly like the task from the first ARPA task in the 70's. The major differences were that the grammar had a somewhat greater perplexity[1], and the recordings were made of read speech. Sentences were constructed from a 1000-word language model, so that no out of vocabulary words were encountered during testing. The corpus contains 21,000 utterances from 160 speakers. A component of the Resource Management task was speaker-independent recognition; that is, the systems were trained on many speakers, and were tested on speakers not in the training set.

Later on in the program, the focus shifted to the Wall Street Journal Task– recognizing read speech from the Wall Street Journal[2]. The first test was constrained to be a 5,000 word vocabulary test with no out-of-vocabulary words; later, a 20k task with out-of-vocabulary words was developed. This task allowed the development of better language models using the larger vocabularies.

Another corpus that was developed in parallel with the RM/WSJ tasks was the ATIS corpus, which contains spontaneous speech about the airline reservation domain. This task is a speech understanding task (as opposed to a speech recognition task). Systems not only had to produce word strings, but also attempt to derive some semantic meaning from these word strings and perform an appropriate function. For instance, if the user said "Show me the flights from Boston to San Francisco," the system should respond by showing a list of flights. Interaction continued with the system in order to reach some goal; in this case, ordering airline tickets. This domain was more practical than the WSJ task, but the vocabulary size was smaller. Systems today are now rather good at this task.

ARPA (the Advanced Research Projects Agency) funded the collections of these corpora, and the collection processes were managed by NIST, the National Institute of Standards and Technology (formerly the National Bureau of Standards). NIST subcontracted a lot of collection work out to sites such as SRI and Texas Instruments. These and other corpora are now distributed through the Linguistic Data Consortium (LDC), which is based at the University of Pennsylvania, in Philadelphia, PA.

6.2.2 Front Ends

The 80's also saw a few advances in front ends, i.e. programs which extract features from the speech signal. Of note are Mel Cepstrum, suggested by Bridle and developed by Mermelstein; PLP (Perceptive Linear Prediction) by Hermansky[3], Delta Cepstral Coefficients by Furui, and other work in auditory models by Seneff, Ghitza, and others.

6.2.3 Hidden Markov Models for Continuous Speech

Another advance in the 1980's was the use of dynamic programming models for continuous speech recognition. Around the same time, Vintsyuk of the Soviet Union, Bridle of England, Sakoe in Japan, and Ney working in Germany all developed recognition systems using one form of dynamic programming or another. The first application of the Baker-Jelinek HMM ideas to continuous speech soon followed, in a system built by

[1] Perplexity is a measure of the average number of words that can follow any word.

[2] The task is now called CSRNAB (Continuous Speech Recognition of North American Business News), which reflects new sources of data such as the San Jose Mercury News

[3] Hynek Hermansky will be giving one of the lectures later this semester.

BBN. With the advent of the new ARPA task, many groups such as CMU, SRI, MIT-Lincoln, and AT&T also hopped on the HMM bandwagon; it is now one of the dominant paradigms in the speech recognition community.

The 80's did not really see much improvement in the science of speech recognition. Rather, groups tended to try to improve performance through feats of engineering. Labs cried out for more and more data to train their models (another reason for the move from Resource Management to Wall Street Journal). Also, with the increase in computer speed, recognizers could use more complicated search techniques within the same amount of time.

6.2.4 1986-7: The 2nd (D)ARPA Task

In 1986, ARPA[4] began funding a second speech recognition task, i.e. the Resource Management task mentioned above. This task involved reading sentences derived from a 1000-word vocabulary. The sentences were questions and commands designed to manipulate a naval information database, although the systems did not actually have to interface with any database; ratings were based on word recognition. Sample sentences from the corpus[PFBP88] include:

- Is Dixon's length greater than that of Ranger?
- What is the date and hour of arrival in port for Gitaro?
- Find Independence's alerts.
- Never mind.

An evaluation of participating systems was held frequently; the average was about once every 8 months to a year. Sites would receive a CD-Rom with test data, and send NIST the sentences produced by their recognizer, where the results would be officially evaluated. Groups were on the honor system to send in their results fairly; they all remained pretty honest since researchers in this field tend to move around from one group to another.

The competition tended to make systems converge on good, similar systems, with everybody trying out tricks that the other labs were using. This had the advantage that everybody could easily incorporate improvements developed at other sites. However, all of the systems now look pretty much the same.

Reddy and Zue said in 1983 that 20 kiloword tasks would require 100K MIPS machines. The ARPA project provided many engineering advances, however, and today on current machines (100-200 MIPS), many sites can do 60K word vocabulary read speech in real time, with less than 10 percent word error. The competition also inspired other sites which were not funded by the project. For example, Cambridge University in England wrote HTK, a HMM toolkit, in order to compete, which has been widely distributed since. It is now possible to use HTK with a little tweaking to get recognition results close to those of the major sites.

As said before, the ARPA program really did not engender any new advances in speech recognition science. Recognition under realistic conditions (e.g. noisy speech, telephone bandwidth) still leaves a lot to be desired. The systems that were built for the program for the most part just scaled to exploit more training data.

[4] This U.S. government agency was originally known as ARPA, but later became DARPA (the D standing for Defense), but after a few years reverted back to ARPA.

6.2.5 The Return of Neural Nets

The field of neural networks suffered a large blow when Minsky and Papert wrote their 1969 book *Perceptrons*, proving that the one of the popular net architectures of the time (Perceptrons) could not even represent the simple function XOR[5]. With the advent of back-propagation, a training technique for Multi-Layer Perceptrons (MLPs), in the early 80's the neural network field experienced a resurgence.

One application of neural networks to speech classification in the early 80's was Morgan's 1981 work with a committee machine which judged whether a section of speech was voiced or unvoiced. Makino, in 1983, used a simple Time-Delayed Neural Network (a close cousin to an MLP in which the input layer includes a delayed version of itself in order to provide a simple context-delay mechanism) to perform vowel recognition; Lang, Hinton, and Waibel in 1988 expanded this technique to add these delayed versions at multiple layers in the net. Also, in 1986, Gold used Hopfield Nets to classify both vowels and consonants.

By the late 80's, 30-40 labs were experimenting with neural networks, both in isolated and continuous contexts. Approximately the same number of sites were using HMM technology for recognition, although many more resources were usually focused on HMMs. Some sites, including Professor Morgan's group at ICSI, are using hybrid HMM/ANN techniques, where the neural network is used as a phonetic probability estimator, and the HMM is used to search through the possible space of word strings comprised of the phones from the ANN.

6.2.6 Other Work

Many of the first speech recognizers were segment-based; that is, the recognizer tried to guess the boundaries of phone segments in the speech signal, and then tried to do recognition based on this segmented speech. Many of the HMM systems that were developed in the 70's and 80's turned to a more frame-based system, where the base units for recognition were small, constant duration sections, or frames, of speech. The 80's saw several systems going back to segment-based modeling as well as variable frame rate modeling.

Other work included developing robust techniques for microphone and channel adaptation, handling noise, and speech under stress and accent. There was also quite a bit of work on identification of speakers, and the related field of speaker verification. Several groups also tried to incorporate more advanced models of syntax and semantics into their recognizers.

6.3 Why study ASR?

Why do we study automatic speech recognition? For one thing, there is a lot of money at stake: speech recognition is potentially a multi-billion dollar industry in the near future, if the field doesn't collapse. Currently, the amount earned in simple telephone applications is nearing 1/2 billion dollars.

We do know some things about speech recognition already, so we have a solid base of work to start from. But it is also clear that we don't know enough. We don't have human-quality speech recognition; things fall apart when a small amount of change is made to the speech signal, like switching microphones.

Speech recognition is potentially very useful. Sample applications include:

- Telephone applications: currently in voice-mail like systems, one has to follow a series of touch-tone button presses to navigate through a hierarchical menu. Speech recognition has the potential to cut

[5] While other network architectures were (and still are) available, including the perceptron's cousin, the Multi-Layer Perceptron, the perceptron had nice properties which made it relatively easy to train.

through the menu hierarchy.

- Hands-free operation: There are many situations in which you want to leave your hands free, but yet issue commands to a device. Phoning out on a car phone, or working in a factory are two examples of such situations. In the latter case, some limited vocabulary systems already exist.
- Applications for handicapped people: Speech recognition is a natural alternative interface to computers for people with limited mobility in their arms and hands, or with sight limitations.
- For some applications, speech may be a more natural interface than keyboard or mouses.
- Dictation: General dictation is a more advanced application, since it requires a much larger vocabulary than say, a replacing a menu system. One speech company, Dragon, has a rather good dictation system on the market called "Dragon Dictate".
- Translation: Another advanced application is translation from one language to another. The Verbmobil project in Germany is both a collaborative and competitive effort to provide language to language translation. The target is to translate from German to Japanese by way of English, but currently they are only working on the German to English parts.

6.4 Why is ASR hard?

• There are several reasons why speech recognition is hard. First of all, natural speech is continuous; it doesn't have pauses between the words. This makes it difficult to determine where the word boundaries are, among other things. Also, natural speech contains disfluencies. People change their mind in mid-sentence on what they want to say, will often accidentally switch phones (as in the phrase "teep a kape"= "keep a tape"), and fill thinking time with "uh"s and "ah"s.

• Natural speech can also change with differences in global or local rates of speech, pronunciations of words within and across speakers, and phonemes in different contexts. The result of this is that we can't just say that X is the spectral representation which corresponds to "ah". The spectrum will change, often quite dramatically, if any of the above conditions are changed.

• Large vocabularies are often confusable. A 20,000 word vocabulary is more likely to have more words that sound like each other than a 10 word vocabulary. There is also the issue of out-of-vocabulary words; for some tasks, no matter what words you have in your vocabulary, there will always be words you haven't seen before. How to model these unknown words is an important unsolved issue.

• As we said before, recorded speech is variable over room acoustics, channel characteristics, microphone characteristics, and background noise. In telephone speech, the particular switches used by the telephone company will have certain effects on the transmitted speech signal, as well as the telephone handset. Different microphones have different frequency responses; tilting a microphone at different angles will also change the frequency response. Nasal sounds will be louder if the microphone is closer to the nose. All of these factors can change the characteristics of the speech signal- a difference that human can often accommodate for, but current recognition systems cannot.

• The algorithms for training recognition systems must be chosen carefully, for large training times are not practical for research purposes. According to rumor, IBM once developed a training algorithm that took a year to run on an IBM 370 with 2 floating point arrays. One can imagine that they didn't run the algorithm very often!

• If we want to replace other modes of input with speech recognition, a high degree of accuracy must be obtained. This means that not only do we have to do well, our systems have to know when they are working well and when they are not, involving some sort of confidence factor in designing a response to an input.

• Another thing that makes natural speech hard is coarticulation effects. The physical realization of a phone can change in different phonetic contexts. For instance, consider the phone /t/ in the following words:

- take
- stake
- tray
- straight
- butter
- Kate

In the case of *take*, the /t/ is aspirated, where as in *stake* it is not. The influence of r in *tray* and *straight* make /t/ come out more like a combination of [t] and [ch]. In *butter*, the /t/ is realized as a quick touch of the tongue against the alveolar ridge, also known as a flap. Finally, in *Kate* the /t/ is sometimes not released (especially in fast speech), so there is no large burst of noise at the end of the word (cf. *take*).

• Besides differences in pronunciations of words within and between speakers, phonological variation often happens at the phrase level. For instance, the phrase "What are you doing?" often comes out sounding like "Whatcha dune?" In the same way, "Juwana eat?" is often the realization of "Do you want to eat." In continuous speech, we also run into the problem that strings of words often sound like each other. Consider the following examples:

- It's not easy to wreck a nice beach.

  It's not easy to recognize speech.

  It's not easy to wreck an ice beach.

- Moes beaches am big you us.

  Most speech is ambiguous.

- sly drool

  slide rule

- say s

  say yes

In addition, even if we get the words right, it's not clear what the semantic content of the words are. Here are some example headlines from newspapers:

- Carter Plans Swell Deficit
- Farmer Bill Dies In House
- Nixon To Stand Pat On Watergate Tapes
- Stud Tires Out

We find most of these headlines funny because there is an alternate semantic interpretation one would not expect to see in the news (and sometimes, it's the first interpretation we get).

6.5 ASR Dimensions

Now that we've talked about how hard speech recognition is, we need to talk about some dimensions of ASR problems. If someone claims to get 99% accuracy on some task, that doesn't tell us anything unless we know some more information about the task she is working on.

6.5.1 Task Parameters

• One dimension of an ASR task is whether the task is speaker-dependent (SD) or speaker-independent (SI). An SD system is one that has been trained mostly on one particular speaker, and test on the same speaker. An SI system is trained on many speakers, and tested on a (usually) disjunct set of speakers. The Bell Labs digit recognizer discussed in [DBB52], for instance, was speaker dependent in a sense, since it had to be adjusted for every speaker. Most of the ARPA tasks like Resource Management and WSJ have both SD and SI components. The definition of speaker-independent is slight difficult to pin down, however. One researcher recently noted that most of the systems today are really not speaker-independent, because, for example, English recognizers will generally fail on speakers who speak English as a second language.

• Another dimension is whether the task is to recognize isolated speech, continuous speech, or spot keywords. The first type of task requires the speaker to put small silences in between each word, and is in general less difficult than recognizing continuous speech, where the word boundaries are not so apparent. A third type of task which falls between the two is keyword spotting, where the recognizer has a list of words that it tries to spot in the continuous speech input. The system must have a confidence factor about the match to prevent false matching of words not in the keyword list.

• The lexicon (vocabulary) size also introduces another parameter. In general, a 20,000 word task is going to be much harder than a 10 word task. But one must take into account the confusability of the particular words in question. For instance, recognizing the "E" set of the alphabet (i.e. the letters $b, c, d, e, g, p, t, v, z$) may be harder than some tasks with a much larger vocabulary.

• Constraints placed on the task may also change the difficulty. Systems that operate on on the blocks world domain[6] will have a more constrained grammar than systems which understand radio news broadcasts. The perplexity of a grammar is a measure of how constrained the grammar is; it is essentially the geometric mean of the branching factor[7] of the grammar. Another constraint on the task is whether the speech input is natural or read speech. Each type of task has different problems, e.g. amount of disfluencies, rate of speech, etc. Real natural speech is very difficult to recognize.

• The recording conditions also play a part in determining the difficulty of an ASR task. The recording may range from wide-band high quality microphones to cellular phones in a moving car. Telephones in particular have a bandwidth of less than 4 kHz, which means that it is more difficult to distinguish high energy consonants, like /f/ and /s/. For example, figure 6.1 shows spectrograms of recordings of the words "foo" and "sue" at 8 kHz bandwidth (high quality microphone) and 8 kHz (telephone) bandwidth. /f/ and /s/ appear as the noise before the relatively straight lines which represent the vowel. Note that it is more difficult to distinguish /f/ and /s/ in the 4 kHz spectral pictures than in the 8 kHz pictures. This is because most of the energy in /f/ and /s/ is above 4 kHz.

• Telephone speech also introduces other challenges. The range of speakers that have access to telephone speech have a greater variability. There is also a larger variability in background noise, and one must account for channel distortion from echo and crosstalk in phones.

[6] The blocks world domain has been a favorite domain in artificial intelligence where objects are geometric solids such as blocks and are to be manipulated by a robot arm.

[7] i.e. how many words can follow another word

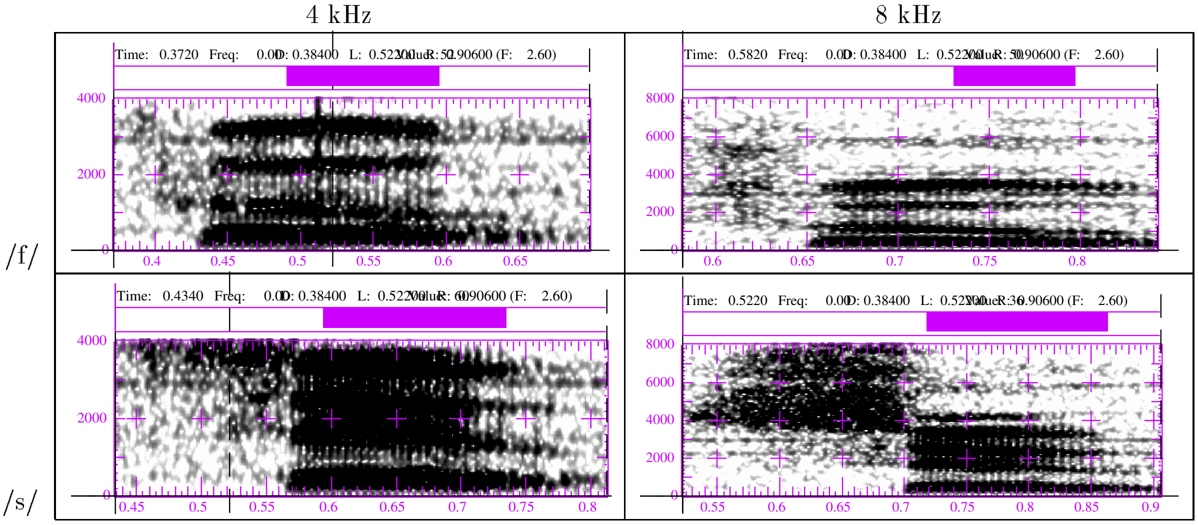

Figure 6.1: Spectrograms of foo and sue sampled a with 4 kHz and 8 kHz bandwidth restrictions.

6.5.2 A sample domain: letters of the alphabet

As an example of a speech recognition task, we present classification of letters of the alphabet. This task has a vocabulary of 26 words, but many of them are confusable. For example, there are four major sets of letters which sound alike:

- E set: B C D E P T G V Z
- A set: J K
- EH set: M N F S
- AH set: I Y R

Recent recognition results from the Oregon Graduate Institute report an accuracy of 89% recognition on letters of the alphabet and 87% accuracy when dealing with spelled names from a telephone speech system trained on 800 speakers (12,500 letters), and tested on 400 speakers (4,200 letters). Consequently, a perceptual experiment was run where 10 listeners identified 3,200 letters from 100 alphabets and 100 spelled names. Human accuracy on the telephone speech database was about 90 to 95%, with the average about 93%. When the experiment was run with high-quality microphone speech, human accuracy jumps to around 99%. The lesson to be learned is that one must take in all of the parameters of a task to appreciate its difficulty.

6.5.3 Dealing With The Public

It is often hard to get the public to do what you want. Speech recognition systems must have appropriate interfaces so that neophytes can use the system easily. Professor Morgan played a tape of 3 data collection sessions at SRI and sessions with a single-digit recognizer, where users became confused and angry due to a relatively poor interface. All of this drove home the point that developing appropriate interfaces, especially for telephone systems, is also an important part of systems building.

References

[DBB52] K. H. Davis, R. Biddulph, and S. Balashek, "Automatic Recognition of Spoken Digits", The Journal of the Acoustical Society of America, pp. 637-642, November, 1952.

[NIST90] National Institute of Standards and Technology Speech Disc 1-1.1 *TIMIT Acoustic-Phonetic Continuous Speech Corpus*, NTIS Order No. PB91-505065, October, 1990.

[PFBP88] P. Price, W. M. Fisher, J. Bernstein, and D. S. Pallett, "The DARPA 1000-Word Resource Management Database for Continuous Speech Recognition," Proceedings of ICASSP'88, Paper S.13.21, pp. 651-654, April, 1988.