Chapter II-2

Automatic Speech Recognition System

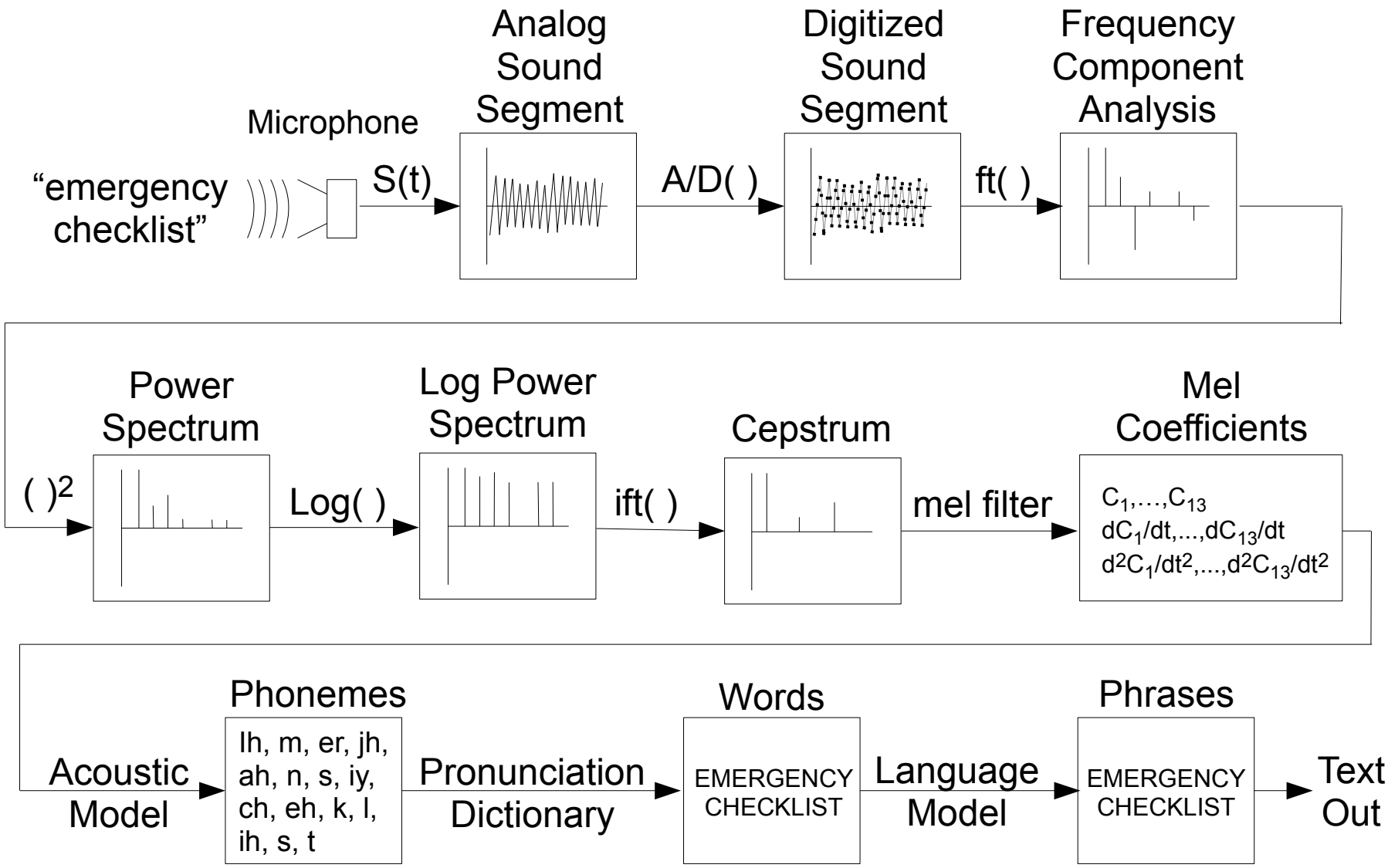
In simplest terms, an Automatic Speech Recognition (ASR) system maps spoken word sounds to text. One might imagine a system that has the sound of each word stored in a file. To recognize a word, the system subtracts the incoming word sound from each stored word sound and selects that word with the least difference. Unfortunately, like snowflakes, no two word sounds are identical, even for the same word. There are differences in length, volume, inflection, pitch, and phasing of the underlying tones. These differences require a different approach be considered.

The human cochlea provides a model as a starting point for an ASR. It consists of a coiled tapered tube lined with sensitive hairs or cilia. Sound introduced into the wide end of cochlea travels down the tapered tube a distance approximately in proportion to the wavelength of the sound. Thus, the high frequency components stimulate the cilia near the wide end of the cochlea and the low frequency components stimulate the cilia farther down. This has the effect of separating the sound into its frequency components in the same way as a Discrete (digital) Fourier Transform (DFT) separates a digital signal into its frequency components. Appendix A provides a short informal tutorial of the DFT.

So, nature has given us the first step in the signal processing chain. Without delving into the details here, it was discovered that analyzing short intervals of time (10 – 20 ms), and some additional steps along with an Inverse Fourier Transform (Cepstrum Analysis) could generate a sequence of *mel feature vectors* that encapsulated the essential information in the sound signal for further processing. See http://en.wikipedia.org/wiki/Mel-frequency_cepstrum. These feature vectors typically consist of thirteen (13) frequency components, and thirteen first and second "differences" of the vector sequence, for a total of 39 vector elements. (Ideally, the "differences" are obtained using signal processing techniques to avoid the amplification of noise that would result from actual finite differencing of the inherently noisy vector sequences. But, some ASR systems do use raw differences effectively.)

The next major development was the use of Hidden Markov Models to determine the best word selection for a given sequence of mel feature vectors. Briefly, an an HMM Acoustic Model determines the best sequence of phonemes from the mel feature vectors. A pronunciation dictionary is then used to find the best word matches. Finally, a Language Model may be used to group words into phrases and sentences. (Since The Language Model in our system here is only perfunctory, it only passes through the bag of words found.) Figure II.2.1 illustrates the ASR process. The acoustic model typically requires many hours of training by multiple speakers to obtain the necessary statistics to be useful as a speaker independent speech recognizer. See http://en.wikipedia.org/wiki/Speech_recognition.

Fortunately, there are many speech recognizers available, both commercial and freeware. So, we need not build our own ASR for our Cognitive Pilot Assistant. A fielded system might find some benefit to employing a commercial product. But, the freeware systems developed by Carnegie-Mellon University (CMU), funded by the Defense Advanced Research Projects Agency (DARPA) and the National Science Foundation (NSF), work very well for our illustration purposes here. As will be seen, the software for our CPA is very modularized, so substituting a different recognizer should be straightforward.



Typical Automatic Speech Recognition Process

Figure II-2.1

On the accompanying CD-ROM, there are directories for Sphinx 3 and Pocket Sphinx. Either may be used to build SpeechServer. The make files are MakeServerS3 and MakeServerPS respectively.

Figure II-2.2 shows the program flow for XXX.c. (Need Figure II-2.2)