Overview of speech recognition

2.1 Introduction to pattern recognition

Prevalent speech recognition systems recognize a test speech signal, using statistical models of similar speech signals. The recognition is performed through the examination of test speech signal and statistical models of speech signals. There are two types of pattern learning, supervised and unsupervised pattern learning. Supervised learning involves training models with class labeled patterns. Unsupervised learning method learns pattern in the data by finding the implicit groupings in the data. Pattern recognition approach has three basic steps:\[60\]:

1. Parameter measurement
2. Pattern comparison
3. Decision making

The overall structure of pattern recognition approach is shown in figure 2.1. Parameter measurement phase/block transforms the input speech signal into a compact efficient set of parameters. It is important to have a good parametrization techniques for an efficient representation of a input speech signal. Section 2.3 discusses important parametrization techniques of speech signals for speech recognition. In the test phase, test input speech signal patterns are compared with the
reference patterns. The best match is selected using decision rules.

2.2 Speech science

Evidences of human speech production and perception provides vital information in developing mathematical models for speech sounds. Hence, it is important to focus on human speech production and perception. The study of acoustic-phonetic characteristics of speech sounds help in improved performance of speech recognition and classification of speech sounds[66]. These factors are of importance in the development of Human Computer Interaction (HCI) through the medium of human language[1].

2.2.1 Human speech production

The understanding of the mechanics of producing speech in humans is very useful in modeling/designing speech production/processing systems akin to humans. Production of speech in humans involves three major steps[60].

1. Formulation of a message in mind

2. Selection of a language code

3. Execution of neuromuscular commands

Speech production process begins when the talker formulates a message that he wants to speak. The formulated message is converted into a language code.
in the second step. Language code is necessary in order to understand the message clearly. The talker generates neuromuscular commands after finalizing the language code. Neuromuscular commands controls the physiological mechanism of sound generation. In addition, simultaneous control of articulatory motion including lips, jaw, tongue, and velum is important to generate proper sequence of sounds.

2.2.2 Acoustic-phonetics

Acoustic-phonetics is the study of the properties of time varying speech signal. Articulatory and auditory properties of humans could be used in developing efficient signal processing techniques for speech processing/recognition. Speech sounds are generated when the vocal tract, the air passage from the glottis to the lips is excited. The mode of excitation can be of three types.

1. Periodic
2. Aperiodic
3. Mixed

Voiced sounds, such as vowels are periodic. Sounds such as fricatives (produced by constricting air flow through a narrow channel at the place of articulation, for example, /f/, /sh/) are aperiodic in nature and stop consonants (produced by the sudden release of air from closed oral cavity) such as /dh/, /bh/ are examples for mixed mode of excitation. Figure 2.2 shows examples of speech sounds generated by different modes of excitation. The word “Akash” has vowel /aa/ in the beginning which is periodic followed by stop consonant /k/. The silence and sudden burst which is the property of stops can also be clearly seen in the figure. The word “Akash” ends with a unvoiced fricative /sh/ which is an example for aperiodic excitation. The spectrogram of /sh/ has no prominent formant peaks since its excitation is aperiodic in nature. However, spectrum of vowel /aa/ has prominent formant peaks seen as dark bands in the figure.
The vocal tract can be approximated as an uniform tube open at one end in the case of neutral vowel using the relationship

\[ f = \frac{c}{\lambda} = \frac{c}{4 \pi L} \quad (2.1) \]

where \( c \) is the velocity of sound in air and \( L \) is the length of vocal tract from glottis to lips. The frequency \( f \) is the fundamental frequency of resonance; it corresponds to the peaks in the frequency spectrum. These peaks are called as \textit{formants} of the signal. Change in size and shape of the resonant cavity results in different values of frequency, amplitude and bandwidth of formants along with distinct speech sounds.

### 2.2.3 Speech perception

Peripheral human auditory system consists of three components: the outer, middle and inner ear. The input acoustic signal captured by the pinna is transmitted through the auditory canal. Pinna and auditory canal constitute the outer ear.
Auditory canal ends with a membrane which is called as tympanic membrane or eardrum. Eardrum converts the acoustic energy to mechanical energy. Middle ear is made up of three bones: malleus, incus and stapes. The mechanical energy is passed through the middle ear and motion in the stapes impinges on the oval window in the inner ear, which is a flexible membrane. The motion in oval window sets the fluid within the cochlea in motion, which is transmitted to the basilar membrane within the cochlea. Different areas basilar membrane are sensitive to different frequencies. Thus, basilar membrane performs frequency analysis of the input sound. The frequency information transmitted by the auditory neurons connected to the basilar membrane is interpreted by the brain inorder to recognize and understand speech.

2.3 Signal processing for speech recognition

A few non-linear processing of sound frequencies and amplitudes by human ear are discussed in 2.3.2 for use in ASR systems. Digital Signal Processing techniques are widely used in analyzing and understanding speech signals. In most signal processing applications, the number of variables is very large. Hence, training these data may consume more memory and time. Also, more number of variable may lead to over fitting of model to training data and hence poor generalization of the data. Feature extraction helps in characterizing the data which get around above problems but still describe the data sufficiently and accurately. Selection of the features which best describes the given data set is important. In most of the cases it is application dependent and expert knowledge is required to select the features. Selection of best feature is a open research problem.

2.3.1 Short term speech processing

Speech signals are quasi-stationary signals. Speech signals can be modeled as a stationary signal over a short period in order to carry out frequency analysis of digital speech signal. In addition, Hidden Markov Models (HMM) are used in
speech recognition based on the principle that a speech signal could be viewed as a short-time stationary signal.

For a signal \( s(t) \), the short-time power spectrum is defined as a function of frequency \( f \) and time \( t \) as

\[
S(f, t) = | \int_{-\infty}^{+\infty} s(\tau) w(t - \tau) e^{j 2\pi f \tau} d\tau |^2
\]  

(2.2)

where \( w(t) \) is a window function. For most practical cases, the effective duration of \( w(t) \) is in the vicinity of 20 to 30 ms. The frequency range of interest in speech usually extends from 0 to 8 Hz. The short-time spectrum of speech contains nearly all of the important information in speech and has formed the basis for most of the present methods of characterizing speech in a parametric form.

### 2.3.2 Mel Frequency Cepstral Coefficients

Psycho-physical studies have shown that human perception of the frequency content of sounds does not follow a linear scale. This research has led to the idea of defining subjective pitch of pure tones. Thus for each tone with an actual frequency, \( f \), measured in Hz, a subjective pitch is measured on a “Mel” perceptual scale. As a reference point, the pitch of a 1kHz tone, 40 dB above the perceptual hearing threshold, is defined as 1000 mels. The mel scale is approximately linear below 1 kHz and logarithmic above, as shown in figure 2.3.

Mel Frequency Cepstral Coefficients (MFCC) procedure uses Mel filter banks to extract subjective information from the spectrum of speech signal. Mel filter has a triangular bandpass frequency response. The spacing as well as the bandwidth of mel filters is determined by a constant mel frequency interval. The spacing is approximately 150 mels, and width of the filter is 300 mels. The modified spectrum thus consists of output power of these filters \( \tilde{S}_k \) when the power spectrum is the input. Mel frequency cepstrum \( \tilde{c}_n \) is given by

\[
\tilde{c}_n = \sum_{k=1}^{K} (\log \tilde{S}_k) \cos \left[ n \left( k - \frac{1}{2} \right) \frac{\pi}{K} \right], \quad n = 1, 2, \cdots, L
\]  

(2.3)
Figure 2.3: Mel scale as a function of frequency.

where $L$ is the desired length of the cepstrum and $K$ is the filter order.

2.3.3 Temporal features

The temporal properties of speech play an important role in linguistic contrasts\cite{62}. Standard front ends such as MFCC, only represent the spectrum within a short analysis frame and thereby tend to neglect very important dynamic patterns in the speech signal\cite{35}. This deficiency has been partly overcome by adding temporal derivatives in the form of delta and delta-delta features to the set. Delta features effectively provide a comb filtering effect in the temporal modulation frequency domain.
2.4 Speech recognition using Hidden Markov Models

Let us consider machine recognition of a word (a sequence of speech sounds). Let each spoken word be represented by a sequence $O$ of speech feature vectors also called observations such as MFCCs.

$$O = o_1, o_2, \cdots, o_T$$  \hspace{1cm} (2.4)

where $o_t$ is the speech vector observed at time $t$. Hence, the word recognition problem can be regarded as that of computing

$$\text{argmax}_i \{P(w_i|O)\}$$  \hspace{1cm} (2.5)

where $w_i$ is the $i^{th}$ vocabulary word. Using Bayes’s rule, we have

$$P(w_i|O) = \frac{P(O|w_i) P(w_i)}{P(O)}$$  \hspace{1cm} (2.6)

So, $\text{argmax}_i \{P(w_i|O)\} \approx \text{argmax}_i \{P(O|w_i) P(w_i)\}$ So, the most probable word spoken depend on $P(O|w_i)$ for a given set of prior probabilities $P(w_i)$. Since the direct computation of joint conditional probability $P(o_1, o_2, \cdots | w_i)$ from spoken word is not practical, a Markov model is employed.

2.4.1 Markov model

A Markov model is a finite state machine (FSM) system changes state at regular time intervals. State transitions are controlled by a discrete probability distribution. Let $a_{ij}$ denote the probability of transition from state $i$ to state $j$. We assume that this probability $a_{ij}$ depends only on the current (at time $t$) state $i$, and does not depend on the identity of states before time $t$. This assumption is called the first order Markovian assumption. The set of probabilities is represented in the form of a $M \times M$ matrix, $A = \{a_{ij}\}$, where $M$ is the number states
in the FSM.

2.4.2 Hidden Markov Models

In a regular Markov model, the state is directly visible to the observer, and therefore the state transition probabilities are the only parameters. In a Hidden Markov Model (HMM), the state is not directly visible, but variables influenced by the state are visible. Each state has a probability distribution over the possible output observations. Therefore the sequence of observations generated by an HMM gives some information about the sequence of states.

If a spoken word or sentence is modeled by a HMM, a state of HMM may model a speech sound (or a part of a speech sound). The observations correspond to feature vectors such as MFCCs. Let $b_j(\theta)$ denote the probability distribution of observations associated with $j^{th}$ state, $j = 1, 2, \cdots M$. Given a trained HMM (denoted by $M$), the probability of observing a given observation sequence $(O = \{o_t\}, \ t = 1, 2, 3 \cdots T)$ when the sequence $(S = \{s(t)\}, \ t = 1, 2, \cdots T)$ can be calculated as

$$P(O, S|M) = \pi_{s(1)} b_{s(1)} (o_1) a_{s(1)s(2)} b_{s(2)} (o_2) \cdots a_{s(T-1)s(T)} b_{s(T)} (o_T) = a_{s(0)s(1)} \prod_{t=1}^{T} b_{s(t)} (o_t) a_{s(t)s(t+1)}$$

(2.7)

here $\pi_i$ denotes the probability of the system entering the model through $i^{th}$ state initially (at $t = 1$). The set of three parameters, $\pi = \{\pi_i\}$, $A = a_{ij}$ and $B = \{b_j\}$ characterize a HMM.

2.4.3 Training a Hidden Markov Model

Training a hidden Markov Model is the propose of finding the best parameter set $\lambda = \{\pi, A, B\}$ such that the probability, $P(O|M)$, feature vector sequence $(O)$ extracted from training speech matching the HMM is maximum.

This probability, $P(O|M)$ can be computed using equation (2.6) if the identities of state at every time instance (i.e, $S = \{s(t)\}$) corresponding to $O = \{o_t\}$ is
known. Since $S$ is not known, $P(O|M)$ is computed as the sum of likelihood $P(O,S|M)$, overall possible state sequence:

$$P(O|M) = \sum_{s} a_{s(0)s(1)} \prod_{t=1}^{T} b_{s(t)}(o_t) a_{s(t)s(t+1)} \quad (2.8)$$

Although we can compute the likelihood of an observation sequence (i.e., sequence of feature vectors extracted from speech) using equation 2.8, we can’t decipher the actual state sequence of model $M$ that generated $O$. In other words, the identity of state sequence is hidden from us, hence the name hidden Markov model.

Using equation 2.8, one can compute forward probability, $\alpha_t(s(t))$, the probability of a partial leading observation sequence $o_1, o_2, \cdots o_t$. Similarly, we can compute backward probability, $\beta_t(s(t))$, the probability of a partial trailing observation sequence $o_{t+1}, o_{t+2}, \cdots o_T$.

Baum and his colleagues proposed an efficient, iterative algorithm for estimating the optimal parameter set, $\lambda = \{\pi, A, B\}$, of a HMM using the above mentioned forward and backward probabilities. This HMM training algorithm is known as Baum-Welch (also forward-backward) algorithm. Details of this algorithm can be found in [60]. Generally (and in this current research work), each and every basic sound e.g., phoneme) is modeled by a HMM. Baum-Welch algorithm can be used to train all phoneme models simultaneously when speech data and the corresponding transcription (description of utterance in terms of phonemes is given).

### 2.4.4 Speech recognition using HMMs

Speech recognition involves finding the best word (equivalently state) sequence, given an utterance and a set of trained HMMs (each HMM modeling a phoneme). In other words, the goal is to compute

$$\arg\max_{i} \{P(w_i|O)\} \approx a_{s(0)s(1)} \prod_{t=1}^{T} b_{s(t)}(o_t) a_{s(t)s(t+1)} \quad (2.9)$$
\( P(w_i) \) is the probability of a particular word sequence. This can be estimated using linguistic constraints such as grammar. The set \( \{ P(w_i) \} \) constitutes a language model. Since computation of \( P(O|w_i) \) using equation (2.8) is computationally expensive, Viterbi proposed that the likelihood can approximated by considering the most likely state sequence.

\[
\hat{P}(O|M) = \max_S \left\{ a_{s(0)s(1)} \prod_{t=1}^{T} b_{s(t)}(o_t) a_{s(t)s(t+1)} \right\} \quad (2.10)
\]

A HMM modelling of a phoneme is a sequence of states. A word HMM can be constructed by concatenating relevant phoneme HMMs. Similarly a sentence HMM can be composed of the corresponding word HMMs. Thus, a sentence HMM, \( M \), is a long sequence of states.

To recognize a test (unknown) utterance, the sequence of feature vectors, \( O \), is computed from the speech signal. Recognition is performed by computing \( \hat{P}(O|M) \) using equation (2.10) and maximizing \( P(w_i|O) \) using equation (2.9). Computation of \( \hat{P}(O|M) \) also gives the best state sequence. From this best state sequence, we can deduce the best phoneme sequence and consequently the best word sequence. Recognition of a sentence not only involves acoustic evidence \( (\hat{P}(O|M)) \) but also linguistic evidence \( (P(w_i)) \). So, both acoustic and language models have to be trained for recognition of spoken sentences.

### 2.5 HTK - Toolkit for speech recognition

HTK is a software toolkit for building and testing HMM based speech recognizer[^82]. It consists of a number of library modules and a number of tools. Functions include speech analysis, training tools, recognition tools, result analysis and tools for labeling. HTK can perform isolated word or connected word speech recognition. It can model whole words or sub-word units.

There are four main phases involved in designing a continuous speech recognizer system.

[^82]: Source of information.
2.5.1 Data preparation tools

HMM models are trained using speech data and their associated transcription. Speech files have to be converted into the appropriate parametric form such as MFCC and any associated transcriptions (description of speech terms of labels such as phones) must be converted to have the correct format and use the required phone or word labels. There are several tools that help in recording speech, compute features such as MFCCs and facilitate prescription of grammatical constraints in the form of a finite state machine.

2.5.2 Training tools

In order to train a HMM, its topology has to be defined. This involves stating the number states and setting initial values of probabilities of transition between states. Normally speech data and the associated sentence (sequence of words) is known. One manually writes a pronunciation dictionary that describes how each word in the vocabulary has to be pronounced in terms of phonemes. Using this pronunciation dictionary and the word sequence, an utterance can be described as a sequence of phonemes; this is known as transcription. This transcription and waveform can be used to train phoneme HMMs as shown in figure 2.1. This training process is described below. All of the phone HMMs are initialized to be identical and have means variances equal to the speech mean and variance. HCompV tool is used for this purpose. Then HERest tool is used to perform embedded training using the entire training set. HERest performs a single Baum-Welch re-estimation of the whole set of HMM phone models simultaneously. For each training utterance, the
corresponding phone models are concatenated and then the forward-backward algorithm is used to accumulate the statistics of state occupation, means, variances, etc., for each HMM in the sequence. Accumulated statistics are used to compute re-estimates of the HMM parameters after processing the training data.

HMMs are refined incrementally using HTK. Hence, a typical progression starts with a simple set of Gaussian context-independent phone models and then iteratively refine them by expanding them to include context-dependency and use multiple mixture component Gaussian distributions. The tool HHEd is a HMM definition editor which clones models into context-dependent sense and applies variety of parameters incrementing the number of mixture components. The usual process is to modify a set of HMMs in stages using HHEd and then re-estimate the parameters of the modified set using HERest after each stage. In case of speaker or accent adapted system, the tools HERest and HVite can be used to adapt HMMs to better model the characteristics of a particular speaker or accent.

One of the hurdles in designing context-dependent HMM systems is data insuf-
iciency. The more complex the model set, the more data is needed to make robust estimates of its parameters and since data is usually limited, a balance must be struck between complexity and the available data. In case of continuous systems, the balance is achieved by tying parameters. Parameter tying allows data to be pooled so that the shared parameters can be robustly estimated. In addition, fully tied mixture systems and discrete probability systems also supported in HTK. In these cases, the data insufficiency problem is usually addressed by smoothing the distributions and the tool HSmooth is used for this.

2.5.3 Recognition tools

A recognition tool, HVite, uses token passing algorithm to perform Viterbi-based speech recognition. HVite takes a network describing the allowable word sequences, a dictionary how each word is pronounced and a set of HMMs as the inputs. It converts a word network to a phone network and then attaching the appropriate HMM definition to each phone instance. Recognition can then be performed on either a list of stored speech files or on direct audio input. The word networks are simple word loops in which any word can follow any word or they are directed graphs representing a finite-state task grammar. In the former case, bigram probability are normally attached to the word transitions.

2.5.4 Analysis tool

Performance of the HMM-based recognizer can be evaluate using the tool HResults. Performance evaluation is done by transcribing some pre-record test sentences and match the recognizer output with the correct reference transcriptions. The tool HResults uses dynamic programming to align the two transcriptions and then count substitution, deletion and insertion errors and compute recognition accuracy.


2.6 Summary

In this chapter, an account of computing useful features from digital speech signals and training HMMs using supervised training approach is given. A brief description of HTK toolkit used in this work for training and evaluation of ASR systems is also given. In the next chapter, we will describe implementation and performance evaluation of a Kannada ASR system.