International Journal of Innovative Research in Advanced Engineering (IJIRAE) ISSN: 2349-2163
Issue 2, Volume 2 (February 2015) www.ijirae.com

Hidden Markov Model Based Robust Speech Recognition

Vikas Mulik* Vikram Mane Imran Jamadar
JCEM,K.M.Gad,E&Tc,&Shivaji University, ADCET,ASHTA,E&Tc&Shivaji university ADCET,ASHTA,Automobile&Shivaji

Abstract— ***Automatic Speech Recognition applications such as voice command and control, audio indexing, speech-to-speech translation, do not usually work well in noisy environments. This paper proposes a new model for a noise robust Automatic Speech Recognition (ASR) based on Hidden Markov Model (HMM) structure with a novel approach for robust speech recognition. A robust and practical speech recognition and Hidden Markov Model (HMM) was proposed aiming at improving speech recognition rate in noise environmental conditions. The system is comprised of three main sections, a pre-processing section, a feature extracting section and a HMM processing section. The proposed estimation methods are applied in combination with oracle masks, which provide an upper performance bound, as well as masks derived from speech presence probability, which represent a more realistic scenario.***

Keywords— ***Automatic speech recognition (ASR), Hidden Markov Model (HMM), robust, feature extracting, oracle masks, speech presence probability.***

1. Introduction

In real-world, in automatic speech recognition (ASR) applications, speech signals generally suffers, degradation by acoustic noise, resulting in decreased system performance. Traditionally the problem of noise robust ASR has been approached in front end feature extraction by reducing variability due to noise while retaining important discriminative information. As an alternative to the previous studies, proposed method explores the missing features approach to speech recognition, in which unreliable spectral components are detected and compensated for accordingly. Proposed Hidden Markov Model [HMM] based approach is for the reconstruction of spectral speech components degraded by acoustic noise. The basic HMM theory was published in a set of papers by Baum et al. around 1967. The HMMs derive from the (simpler) Markov chains. Markov chain is a discrete (discrete-time) random process with the Markov property. It is a discrete-time random process because it is in a discrete state (from among a finite number of possible states) at each "step". The Markov property states that the probability of being in any particular state only depends on the previous state it was at before.

By implicitly quantizing spectrographic data, feature trajectories can be interpreted as transitioning through a HMM-defined trellis, with respect to time or with respect to frequency. The proposed method utilizes observed speech together with a local noise estimate to compute observation statistics. With the mentioned transitional and observation information, the proposed HMM-based missing data algorithm uses the traditional forward–backward algorithm to obtain optimal spectral estimates in the MMSE sense. A major component of missing feature approaches for robust speech recognition systems is mask estimation, which detects the spectral location of reliable features. Many missing feature studies include results based upon oracle masks, for which exact knowledge of a clean version of the input speech signal is known. Knowledge of oracle masks provides an upper performance bound for data imputation techniques. In Order to convey realistic results, mask estimation must be performed. It is proposed to develop spectral masks based on speech presence probability (SPP), in which the probability of active speech is based upon the statistical distribution of speech and noise spectral magnitudes. It is proposed to obtain clean speech & identify the speaker.

2. Relevance & Literature Survey

Relevance:

Most speech analysis techniques encode speech parameters in the frequency domain, a domain which enables most speech signals to be discriminated accurately. Conversion from time to frequency domain is based on three basic methods: Fourier transforms, digital filter-banks, and linear prediction. The Fourier transform allows the passage of a signal from the time to frequency domain, and vice versa (Inverse Fourier Transform). This transform has also been extended for use with discrete time signals, sampled at regular intervals, known as the Discrete Fourier Transform (DFT). This uses a set Window length of samples for analysis which is proportional to the frequency resolution. A large window produces greater frequency resolution, but is naturally at the cost of temporal resolution. This is one of the essential tools of speech processing, along with its more efficient counterpart that of Fast Fourier Transform (FFT), used in most applications where spectrum estimates are required. The second method for estimating the spectral envelope is via a filter-bank which separates the signal frequency bandwidth in a number of frequency bands where the signal energy is measured. This method offers two main advantages over DFT,that of the small number of parameters used to represent the spectrum envelope, and the possibility of having different frequency resolutions for each envelope. This last advantage, along with the characteristics of the filters and the spacing of their central frequency has been found to be important, for instance, in the simulation of cochlea like filtering in auditory periphery modeling.

These are several possible choices for what type of signal model is used for characterizing the properties of a given signal. Broadly one can dichotomize the types of signal models into the class of deterministic models, and the class of statistical models. Deterministic models generally exploit some known specific properties of the signal, e.g., the signal is a sine wave, a sum of exponentials, etc.

In these cases, specification of the signal model is generally straight- forward; all that is require is to determine(estimate) values of the parameters of the signal model (e.g., amplitude, frequency, phase of a sine wave, amplitudes and rates of exponentials, etc.). The second broad class of signal models is the set of statistical models in which one tries to characterize only the statistical properties of the signal. Examples of such statistical models include Gaussian processes, Poisson processes, Markov processes, and hidden Markov processes, among others. The underlying assumption of the statistical model is that the signal can be well characterized as a parametric random process, and that the parameters of the stochastic process can be determined (estimated) in a precise, well-defined manner.[7][2]

The concealment or mitigation of the errors introduced by a wireless channel is a necessary step to ensure the quality of the received signals. Minimum mean square-error (MMSE) estimation can be applied for this purpose. The MMSE-based techniques can be meaningfully improved by introducing some source a priori knowledge in their formulation by means of a source model. This modeling makes it possible to exploit the high degree of correlation among consecutive signal samples. Speech and images are usually modeled as first-order Markov processes. Examples of this can be found in and in under a formulation based on Hidden Markov models (HMMs) (adopted in this work). The main drawback of the HMM-based MMSE techniques is the large computational burden that they involve. HMM-based MMSE estimation involves a recursive probability computation that requires operation.[3][4]

Most modern automatic speech recognition (ASR) systems model the speech as a non-stationary stochastic process by statistically characterizing a sequence of spectral estimations. The common technique for spectral estimation includes an approximation of auditory filtering, a compressive nonlinearity (usually the logarithm), and de-correlation of the spectral estimation through an approximate Karhunen–Lo` eve (KL) transform (the discrete cosine transform). These steps represent rough approximations of the most fundamental aspects of auditory processing, frequency selectivity and magnitude compression. In the last five to ten years, the frequency selectivity for ASR front-ends has migrated from a linear to a perceptually based frequency scale. This progress, toward a better auditory model for ASR, has improved robustness [6].

The proposed work consist of a data imputation framework utilizing minimum mean square error (MMSE) estimation of missing features for noise robust speech recognition. Hidden Markov models (HMMs) have been extensively used in signal processing and communications due to their elegant framework which captures steady-state, transitional, and observation statistics. Recently, HMMs have been successfully used to model feature trajectories during the estimation of missing data due to lost packets during the transmission of digital information.[1]

Literature Review:

There are various authentication methods which are summarized as follows.

In the paper "a tutorial on hidden Markov models and selected applications in speech recognition " Lawrence r. Rabiner, fellow, IEEE*' proceedings of the IEEE,* vol. 77, no. 2, *February 1989,* discussed the theoretical aspects of this type of statistical modeling and show how they have been applied to selected problems in machine recognition of at speech. [7]

Finally the most important reason why signal model are important is that they often work extremely well in practice, and enable us to realize important practical system-e.g., prediction system, recognition system, identification system etc., in a very efficient manner.[7]

In the paper "A model of dynamic auditory perception and its application to robust word recognition, *IEEE transactions on speech and audio processing*, vol. 5, no. 5, September 1997" by *Brian Strope and Abeer alwan,member, IEEE Brian strope and Abeer alwan,member, IEEE,* it describes two mechanisms that augment the common automatic speech recognition (ASR) front end and provide adaptation and isolation of local spectral peaks. A dynamic model consisting of a linear filter bank with a novel additive logarithmic adaptation stage after each filter output is proposed.[6]

It is also discuss that due to environmental noise mismatch in level of background noise between training and recognition phase. The probability distribution which obtained at training is not valid for testing phase. According to component on which recognizer work they classify in three category (1) augmenting the front end by a statistical estimator to estimate the clean speech parameter from the noisy signal; (2) adaptation of the HMM output probability distribution to the presence of noise; (3) modifying the front end such that the acoustic feature are more robust to noise.

Both the estimation and adaptation methods rely on a priori knowledge of the statistical properties of noise, and on these properties being easily modeled. Another approach that does not depends on such assumption is to modify the front end such that the acoustic representation (the feature vector) will be less corrupted by noise.[6]

In the paper "*Noise power spectral density estimation based on optimal smoothing and minimum statistics*" "*IEEE transactions on speech and audio processing,* vol. 9, no. 5, july 2001" by *Rainer martin, senior member, IEEE,* describe a method to estimate the power spectral density of non-stationary noise when a noisy speech signal is given.

The method can be combined with any speech enhancement algorithm which requires a noise power spectral density estimate by minimizing a conditional mean square estimation error criterion in each time step derive the optimal smoothing parameter for recursive smoothing of the power spectral density of the noisy speech signal. Based on the optimally smoothed power spectral density estimate and the analysis of the statistics of spectral minima an unbiased noise estimator is developed. [5]. The minimum statistics algorithm does not use any explicit threshold to distinguish between speech activity and speech pause and is therefore more closely related to soft-decision methods than to the traditional voice activity detection methods.

Similar to soft-decision methods it can also update the estimated noise power spectral density during speech activity.The minimum statistics method rests on two observations namely that the speech and the disturbing noise are usually statistically independent and that the power of a noisy speech signal frequently decays to the power level of the disturbing noise. It is therefore possible to derive an accurate noise power spectral density estimate by tracking the minimum of the noisy signal power spectral density. Since the minimum is smaller than (or in trivial cases equal to) the average value the minimum tracking method requires a bias compensation. In this paper, the bias is a function of the variance of the smoothed signal power spectral density and as such depends on the smoothing parameter of the power spectral density estimator. [5]

In the paper"likelihood-maximizing beam forming for robust hands-free speech recognition"by Michael l. Seltzer, member, IEEE, Bhiksha raj, member, IEEE, and Richard m. Stern, member, IEEE, *IEEE transactions on speech and audio processing,* vol. 12, no. 5, *September 2004*, a new approach to microphone-array processing is proposed in which the goal of the array processing is not to generate an enhanced output waveform but rather to generate a sequence of features which maximizes the likelihood of generating the correct hypothesis.

In this approach, called likelihood-maximizing beam forming, information from the speech recognition system itself is used to optimize a filter-and-sum beam former. Speech recognition experiments performed in a real distant-talking environment confirm the efficiency of the proposed approach. [4]

In the paper "efficient MMSE-based channel error mitigation techniques. Application to distributed speech recognition over wireless channels" Hui' 1536-1276 , 2005 IEEE" by Antonio m. Peinado, member, IEEE, Victoria Sánchez, member, IEEE, José l. Pérez-córdoba, member, IEEE, and Antonio j. Rubio, senior member, IEEE, this work addresses the mitigation of channel errors by means of efficient minimum mean-square-error (MMSE) estimation. Although powerful model-based implementations have been recently proposed, the computational burden involved can make them impractical. We propose two new approaches that maintain a good level of performance with a low computational complexity.

These approaches keep the simple structure and complexity of a raw MMSE estimation, although they enhance it with additional source a priori knowledge. The proposed techniques are built on a distributed speech recognition system. Different degrees of tradeoff between recognition performance and computational complexity are obtained.[3]

In the paper "Band-independent mask estimation for missing-feature reconstruction in the presence of unknown background noise" *1-4244-0469-x/06,2006 IEEE*" by Wooil kim and Richard m. Stern an effective mask estimation scheme for missing-feature reconstruction is described that achieves robust speech recognition in the presence of unknown noise. [2]

In the paper "HMM-based reconstruction of unreliable spectrographic data for noise robust speech recognition*" IEEE transactions on audio, speech, and language processing*, vol. 18, no. 6, August 2010" by *Bengt j. Borgström, student member, IEEE, and Abeer alwan, fellow, IEEE***,** presents a framework for efficient HMM-based estimation of unreliable spectrographic speech data. It discusses the role of hidden Markov models (HMM) during minimum mean-square error (MMSE) spectral reconstruction. [1]

3. SYSTEM DESIGN & DEVELOPMENT

3.1 Proposed Block Diagram:

To accomplish this objective, the following techniques have been proposed as shown in given figure.

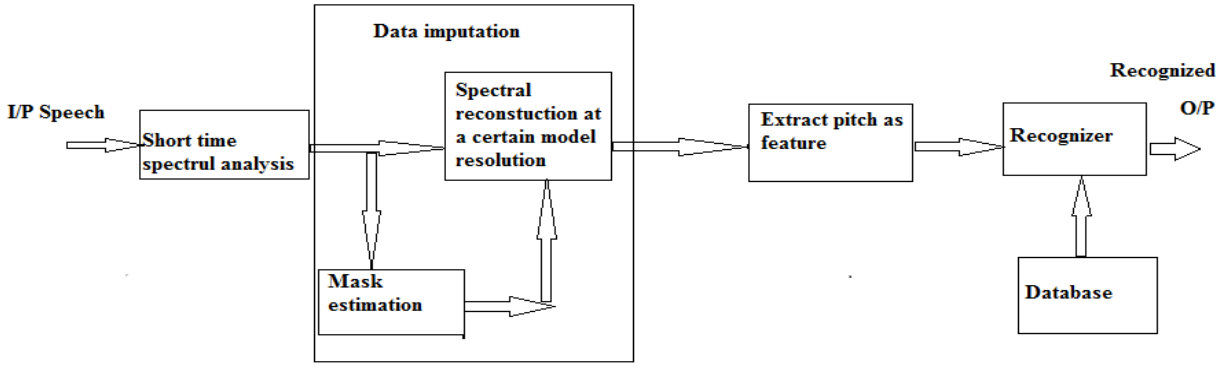

Figure 3.1 Proposed Block Diagram

Level 1:

With help of microphone speech is recognized in MATLAB. This signal is processed using short time spectral analysis. Microphone is one of the transducer which convert audio signal in to the electrical signal.

Level 2:

Decompose an input signal in the frequency domain by applying Mel-filter short time Fourier transform [STFT] to construct input speech signal in sampled frequency frame. The aim of quantization in this our work is not to compression data, but rather to designate each decoded feature element a corresponding HMM state.

Level 3:

This frequency frame is given to the data imputation block. Imputation means the process of replacing substituted value. This block contains two sub-blocks, mask estimation and spectral reconstruction of certain model resolution as explained below.

3.1] Mask estimation- It is major component for identifying missing feature approach, which detects the spectral location of missing features. It use spectral mask based speech presence probability [SPP] algorithm.

3.2] Spectral reconstruction of certain model resolution- This use the Hidden Markov Model [HMM] based spectral construction. In speech recognition HMM is used to model non-stationary signal. HMM derive from Markov chain. Markov chain is a discrete random process. The Markov property states that the probability of being in any particular state only depends on the previous state it was at before.

While in a Markov chain the output in each state is known, in an HMM each state incorporates a probabilistic function to generate the output. An HMM can be thought of a double stochastic process state sequence and output in each state where the state sequence being not directly observable. It is therefore called Hidden Markov Model.

HMM By implicitly quantizing spectrographic data, feature trajectories can be interpreted as transitioning through a HMM with respect to frequency. Here applying HMM during minimum mean square error [MMSE] for spectral reconstruction. MMSE can be used for to mitigation the error in the channel. This modeling makes it possible to exploit the high degree of correlation among consecutive signal samples. HMM based approach for reconstruction of spectral speech components degraded by acoustic noise.

Level 4:

The speech feature like pitch is then extracted and compared with standard speeches stored in data base. The speaker is identified more accurately by recognizer block. The recognizer block will compare the corresponding speech features.

Level 5:

Examine the performance of system like accuracy, false rate etc. and compare with existing system is also included.

Level 6:

Work is proposed to carry out by using MATLAB software.

3.2 Theory of Hidden Markov Model:

The Hidden Markov Model (HMM) is a popular statistical tool for modeling a wide range of time series data. The data samples in the time series can be discretely or continuously distributed scalars or vectors. The basic HMM theory was published in a set of papers by Baum et al. around 1967 [7]. The HMMs derive from the (simpler) Markov chains. The principal of Hidden Markov model is based on modeling speech pattern as a sequence of observation vector derived from a probabilistic function of a first order Markov chain. The state in such a model is connected by probabilistic transition and each state is identified with an appropriate output probability density function.

While in a Markov chain the output in each state is known, in an HMM each state incorporates a probabilistic function to generate the output. An HMM can be thought of a double stochastic process state sequence and output in each state, where the state sequence being not directly observable so it is called hidden. And this model is called hidden Markov Model.

Below shows the basic mathematical theory of Markov process, which shows that mathematics has a role to play in speech recognition.

3.2.1 Discrete Markov Processes:

Consider a system which may be described at any time as being in one of a set of N distinct state, $s_1, s_2, s_3, \ldots\ldots s_N$, as illustrated in Fig. 3.1 (where N=5 for simplicity). At regularly spaced discrete times, the system undergoes a change of state (possibly back to the same state) according to a set of probability associated with the state. Here denote the time instant associated with state changes as t = 1,2,3, ..., and denote the actual state at time t as q_t .

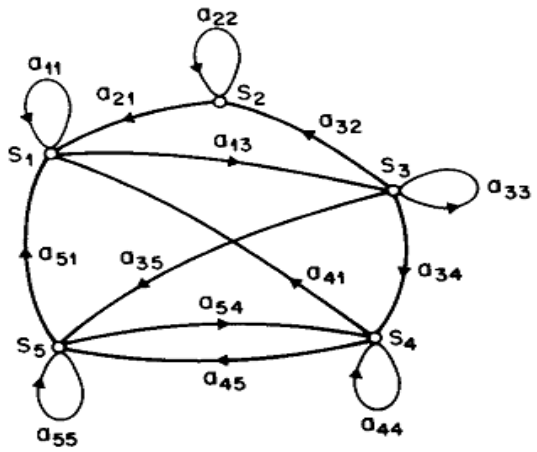

Fig.3.2 A Markov chain with 5 state (labeled s_1 to s_5) with selected state transition.

A full probabilistic description descript of the above system would, in general, require specification of the current state (at time), as well as all the predecessor states. For the special case of a discrete, first order, Markov chain, this probabilistic description is truncated to just the current and the predecessor state, i.e.,

$$P[q_t = s_j | q_{t-1} = s_i, q_{t-2} = s_{k,\ldots\ldots}] \qquad [1]$$
$$= P[q_t = s_j | q_{t-1} = s_i].$$

Furthermore only consider those processes in which the right hand side of (1) is independent of time, there by leading to the set of state transition probability a_{ij} of the form

$$a_{ij} = P[q_t = s_j | q_{t-1} = s_i]. \quad 1 \leq i, j \leq N \qquad [2]$$

With the state transition coefficient having the properties

$$a_{ij} \geq 0 \qquad [3]$$

$$\sum_{j=1}^{N} a_{ij} = 1 \qquad [4]$$

Since then obey standard stochastic constraints.

The above stochastic process could be called an observable Markov model since the output of the process is the set of states at each instant of time, where each state corresponds to a physical (observable) event.

3.2.2 Discrete Markov Processes:

Assuming the following scenario. You are in a room with a barrier (e.g.,a curtain) through which you cannot see what is happening. On the other side of the barrier is another person who is performing a coin (or multiple coins) tossing experiment. The other person will not tell you anything about what he is doing exactly, he will only tell you the result of each coin flip. Thus a sequence of Hidden coin tossing experiment is performed, with the observation sequence consisting of a series of heads and tail; e.g., a typical observation sequence would be

$$O = O_1 O_2 O_3 \ldots\ldots\ldots O_T$$
$$= H, H, T, T, T, H \ldots.$$

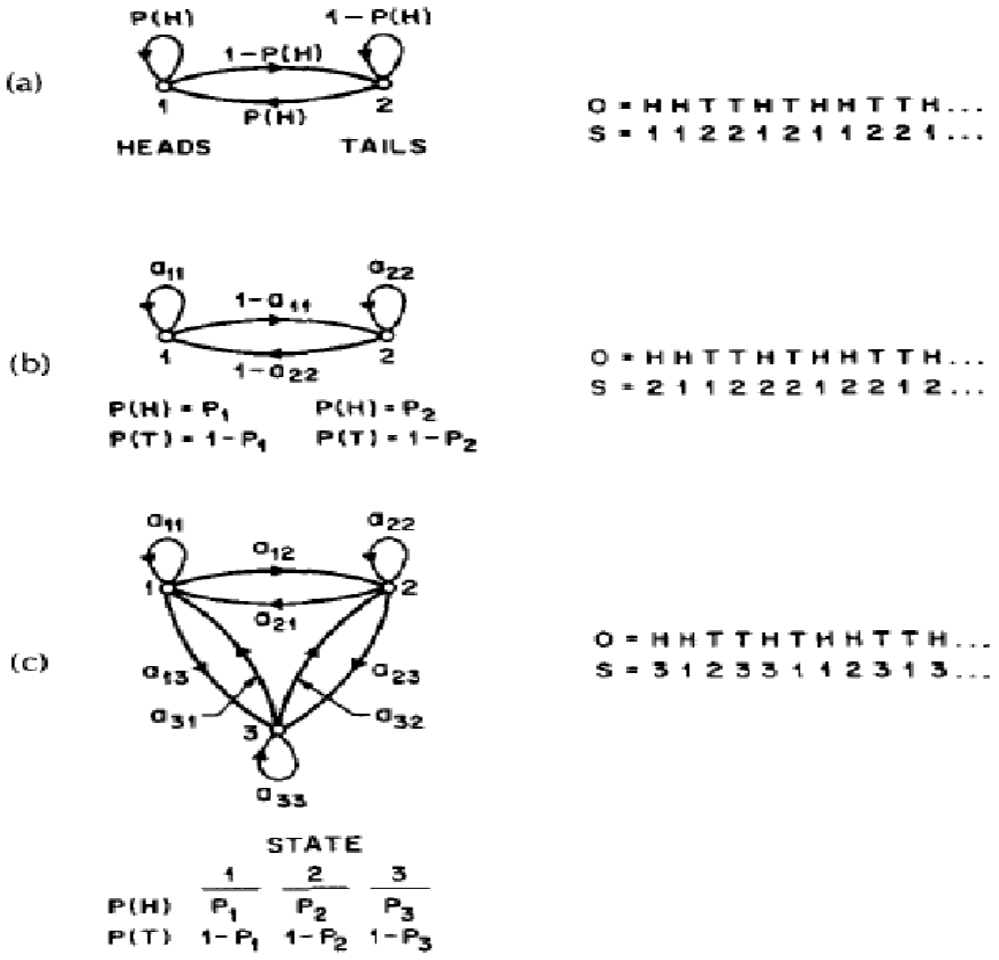

Fig.3.3 Three possible Markov models (a) I-coin model. (b) 2-coin model. (c) 3-coin model.

Where H stands for heads and T stands for tails. Given the above scenario, the above scenario, the problem of interest is how do build an HMM to explain (model) the observed sequence of heads and tail. The first problem one faces is deciding what the states in the model correspond to, and then deciding how many states should be in the model. One possible choice would be to assume that only a single biased coin was being tossed. In this case we could model the situation with a 2-state model where each state corresponds to a side of the coin (i.e., head or tail). This model is depicted in Fig. 3.3(a). In this case the Markov model is observable, and the only issue for complete specification of the model would be to decide on the best value for the bias (i.e., the probability of, say, head). Interestingly, an equivalent HMM to that of Fig. 3.3(a) would be a degenerate 1-state model, where the state corresponds to the signal biased coin, and the unknown parameter is the bias of the coin.

A second form of HMM for explaining the observed sequence of coin then toss outcome is given in Fig. 3.3(b). In this case there are 2 states in the model and each state corresponds to a different, biased, coin being tossed. Each state is characterized by a probability distribution of heads and tails, and transitions between states are characterized by a state transition matrix. The physical mechanism which accounts for how state transition are selected could itself be a set of independent coin tosses, or some other probabilistic event.

A third form of HMM for explaining the observed sequence of coin toss outcomes is given in Fig. 3.3(c). This model corresponds to using 3 biased coins, and choosing from among the three, based on some probabilistic event.

3.2.2 Elements of HMM:

Here now formally define the element of an HMM, and explain how the model generates observation sequence. An HMM is characterized by the following.

1) N, the number of states in the model. Although the states are hidden, for many practical applications there is often some physical significance attached to the states or to sets of states of the model. Generally the states are interconnected in such a way that any states can be reached from any other state (e.g., an ergodic model). We denote the individual states as $= \{S_1, S_2, \dots \dots \dots . S_N\}$, and the state at time t as q_t.

2) M, the number of distinct observation symbol per state, i.e., the discrete alphabets size. The observation symbol corresponds to the physical output of the system being modeled. Here denote the individual symbol as

$$V = \{V_1 V_1 V_1 \dots \dots \dots . V_M\}$$

3) The state transition probability distribution A= { a_{ij}} where

$$a_{ij} = P[q_t = s_j | q_{t-1} = s_i]. \quad 1 \leq i, j \leq N$$

4. RESULT:

4.1 Linear Predictive Coding (LPC)

Linear predictive coding (LPC) is used in audio signal processing and speech processing for representing the spectral envelop of a digital signal of speech in compressed form, using the information of a linear predictive model. It is a way of encoding the information in a speech signal into a small space for transmission over restrictive channel. It is one of the most powerful speech analysis techniques, and one of the most powerful methods for encoding good quality speech at a low bit rate and provides extremely accurate estimates of speech parameter. LPC has application in filter design and speech coding.

LPC encodes a signal by finding a set of weight on earlier signal values that can predict the next signal value. It determine the coefficient of a forward linear predictor by minimizing the prediction error in the least squares sense. For a P^{th}- order linear predictor, the current values of the real-valued time series X(n) is based on past samples.

$$\boldsymbol{X_P(n) = -A(2) * X(n-1) - A(3) * X(n-2) - \cdots - A(N+1) * X(n-N)}$$

Such that the sum of the squares of the errors

$$\boldsymbol{err(n) = X(n) - X_P(n)}$$

is minimized. And the LPC coefficient are given by

$$\boldsymbol{A(1), A(2), \dots . . A(N+1)}$$

4.2 Training procedure

Isolated words such as one, two ,three and four were spoken by four different speakers, each uttering a word 20 times. These words were recorded in the 16 bit-mono PCM format at a sampling rate of 10,000 samples per second. The acquired speech files were processed after noise reduction in the voice file. This edited speech file shown in fig below fig. On the edited speech samples, LPC was carried out using the Durbin's method to extract LPC coefficients frame by frame. The speech samples are framed with a frame size of N = 420 samples (=42 milliseconds). Consecutive frames are spaced M = 180 samples apart (= 8 milliseconds), corresponding to a frame overlap of 240 samples (=24 milliseconds). Then each frame is multiplied by a N sample Hamming Window W(n), where

$$W(n) = 0.54 - 0.46\cos(2\Pi n/(N-1))$$

Hamming window is very useful in speech like waveforms to smoothen the ends of the frame. Each windowed set of speech samples is auto-correlated to give a set of 'p+1' coefficients, where' p' is the order of the desired LPC vector. We have chosen p=5. One such LPC coefficients for each is shown in fig

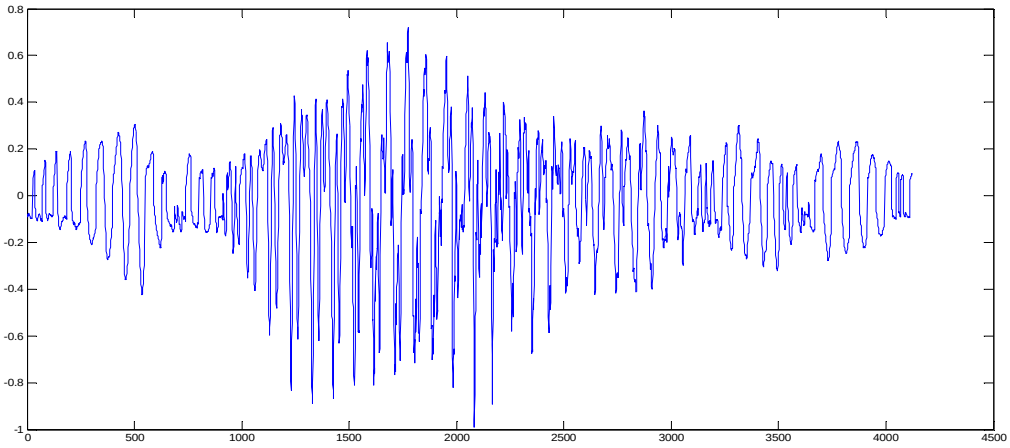

Figure 4.1. Edited speech waveform for the word 'one

S

1	2	3	4	5	6	7	8	9
1.0000	-1.0081	0.1176	-0.2242	0.2166	-0.0399	0.1240	-0.1912	0.0067
1.0000	-0.8460	-0.0680	-0.2245	0.2058	-0.0041	0.1604	-0.3407	0.1189
1.0000	-0.9253	-0.0158	-0.1539	0.0805	-0.0190	0.1878	-0.1584	0.0091
1.0000	-1.0377	0.1967	-0.3690	0.3603	-0.1564	0.2542	-0.2766	0.0331
1.0000	-1.0187	0.0887	-0.2063	0.1861	-0.1223	0.1845	-0.0985	-0.0099
1.0000	-0.9689	0.0243	-0.2120	0.2898	-0.2192	0.2272	-0.0335	-0.1051
1.0000	-1.0754	0.1297	-0.1839	0.1294	-0.0347	0.3290	-0.2437	-0.0472
1.0000	-0.9085	-0.1233	-0.0546	0.1430	-0.0027	-0.0451	-0.2015	0.2041
1.0000	-1.0806	-0.0858	0.0490	0.0574	0.1371	-0.0633	-0.4167	0.4399
1.0000	-1.2548	0.1361	-0.2088	0.3285	0.2123	-0.2315	-0.2697	0.3171
1.0000	-1.3713	0.2217	-0.1063	0.2110	0.2121	-0.1522	-0.2215	0.2262
1.0000	-1.5834	0.3684	0.1862	-0.0037	0.0042	0.0462	-0.0536	0.0462
1.0000	-1.5111	0.1589	0.1096	0.2967	-0.0399	0.1739	-0.2595	0.0787
1.0000	-1.4026	0.1097	0.0274	0.2287	-0.0201	0.2336	-0.1702	0.0025
1.0000	-1.4307	0.1769	0.2395	0.0141	-0.2036	0.4155	-0.3006	0.1041
1.0000	-1.2429	0.0397	0.2353	0.0565	-0.2167	0.0825	-0.2548	0.3261
1.0000	-1.2139	0.0105	0.1751	0.0398	-0.1126	0.0667	-0.2208	0.2750
1.0000	-1.3797	0.0356	0.5006	-0.1628	-0.1105	0.3743	-0.5570	0.3206
1.0000	-1.5817	0.2885	0.5182	-0.5418	0.1531	0.7495	-0.7210	0.1658
1.0000	-1.8209	0.6033	0.6073	-0.8638	0.2532	1.0711	-1.1342	0.3026
1.0000	-1.9137	0.9313	0.1340	-0.5436	0.2845	0.8541	-1.0155	0.2889
1.0000	-1.9914	1.2688	-0.2358	-0.3604	0.2586	0.6405	-0.8127	0.2602
1.0000	-2.1543	1.8034	-0.9064	0.0690	0.3113	0.2213	-0.5079	0.1819
1.0000	-1.9880	1.3401	-0.3076	-0.4732	0.6868	0.0828	-0.5582	0.2361
1.0000	-1.8539	0.9555	0.2281	-0.8566	0.6951	0.3807	-0.9256	0.4088
1.0000	-1.9058	1.0055	0.3514	-1.0349	0.8279	0.3611	-1.0918	0.5426
1.0000	-1.9276	1.0825	0.2636	-0.9502	0.7332	0.4047	-1.1414	0.5803
1.0000	-1.9708	1.1609	0.1753	-0.8558	0.6608	0.4191	-1.1530	0.5964
1.0000	-1.9389	1.0008	0.3797	-0.8282	0.4039	0.5668	-1.0631	0.5057
1.0000	-2.0496	1.3585	-0.1302	-0.4464	0.3030	0.2950	-0.6287	0.3244
1.0000	-1.9993	1.2004	-0.0369	-0.3657	0.2649	0.1108	-0.3702	0.2169
1.0000	-2.0915	1.5488	-0.5783	0.0629	0.1994	-0.1262	-0.1462	0.1480

Figure 4.2. LPC coefficient for word 'one'

Once the LPC coefficients are obtained, we quantize them using a scheme known as vector quantization.

4.3 Vector quantizatio (VQ)

Vector quantization (VQ) is a lossy data compression method based on the principle of block coding. It is a fixed-to-fixed length algorithm. It is a process of mapping vectors from a large vector space to a finite number of regions in that space. Each region is called a cluster and can be represented by a center called as code word. The collection of all code word is called codebook.

Design problem-

The VQ design problem can be stated as follows. Given a vector source with its statistical properties known, given a distortion measure, and given the number of code-vectors, find a codebook and a partition which result in the smallest average distortion. Assuming that there is a training sequence consisting of M source vector:

$$T = \{x_1, x_2 \dots \dots \dots x_M\}.$$

M is assumed to be sufficiently large so that all the statistical properties of the source are captured by the training sequence. The source vector is K dimensional, e.g.

$$X_m = (x_{m,1}, x_{m,2}, \dots \dots \dots, x_{m,k}), \quad m = 1, 2, \dots \dots M$$

Let N be the number of code-vector and let,

$$C = \{c_1, c_2 \dots \dots \dots, c_N\},$$

Represent the codebook. Each code-vector is K dimensional, e.g.

$$c_n = (c_{n,1}, c_{n,2}, \dots \dots \dots, c_{n,k}), \quad n = 1, 2, \dots \dots N.$$

Let S_n be the encoded region associated with code vector C_n and let,

$$p = \{S_1, S_2, \dots \dots \dots \dots, S_N\}$$

Denote the partition of the space. If the source vector X_m is in encoding region S_n , then its approximation (denoted by $Q(x_m)$) is c_n :

$$Q(x_m) = c_n \qquad \text{if } x_m \in S_n.$$

Assuming a squared-error distortion measure, the average distortion is given by:

$$Dave = \frac{1}{Mk} \sum_{m=1}^{M} ||x_m - Q(x_m)||^2$$

Where,

$$||e|| = e_1^2 + e_2^2 + \cdots + e_k^2$$

The design problem can be succinctly stated as follows: Given T & N, find C and P such that D_{ave} minimized.

Optimality criteria

If C and P are a solution to the above minimization problem, then it must satisfy the following two criteria.

Nearest neighbor condition:

$$S_n = X: ||X - C_n||^2 \leq ||X - C_{n'}||^2 \quad ¥_n = 1, 2, \dots \dots N$$

This condion say that encoding region S_n should consist of all vector that are closer to C_n than any other of the other code vector.

Centroid condition:

$$C_n = \frac{\sum_{x_m \in x_n} x_m}{\sum_{x_m \in x_n} 1} \quad n = 1, 2, \dots . N$$

This condition says that the code-vector C_n should be average of those training vector that are encoding region S_n. In implementation, one should ensure that at least one training vector belongs to each encoding region (so that the denominator in the above equation never zero)

LBG Design Algorithm:

Given T. Let $\mathcal{E} > 0$ be a small number.

1) Let $N = 1$ and

$$C_1^* = \frac{1}{Mk} \sum_{m=1}^{M} x_m$$

Calculate

$$C_1^* = \frac{1}{Mk} \sum_{m=1}^{M} ||x_m - C_1^*||^2$$

2)**splitting:** For $i = 1, 2, \dots \dots . N$ set

$$\mathbf{C_i^{(0)}} = (\mathbf{1} + \epsilon)\mathbf{C_i^*}$$
$$\mathbf{C_{N+i}^{(0)}} = (\mathbf{1} - \epsilon)\mathbf{C_i^*}$$

Set ***N=2N***

3)**Iteration:** Let

$$\boldsymbol{D_{ave}^{(0)} = D_{ave}^*}$$

4) Set the iteration index i=0.

i. For m=1,2,…..,M, find the minimum value of

$$\| \boldsymbol{x_m}\text{-}\boldsymbol{c_n^{(i)}} \|^2,$$

over all n=1,2 …. , N. Let n* be the index which achieves the minimum.

Set

$$Q(x_m) = c_n^{(i)}$$

ii. For n=1,2 …. , N, updates the code-vector

$$\boldsymbol{c_n^{(i+1)}} = \frac{\sum_{Q(x_m)=c_n^{(i)}} x_m}{\sum_{Q(x_m)=c_n^{(i)}} 1}$$

iii. Set i=i+1

iv. Calculate

$$\boldsymbol{D_{ave}^{(i)} = \frac{1}{Mk}\sum_{m=1}^{M} ||x_m - Q(x_m)||^2}$$

v. If

$$\boldsymbol{D_{ave}^{(i-1)} - D_{ave}^{(i)} / D_{ave}^{(i-1)} > \epsilon}$$

Go back to step (i)

Set

$$\boldsymbol{D_{ave}^* = D_{ave}^{(i)}}$$

For $n = 1,2, \ldots \ldots, N$ set

$$\boldsymbol{C_n^*}=\boldsymbol{C_n^{(i)}}$$

As the final code vector.

4) Repeat the step 3 & 4 until the desire number of code vector is obtained.

4.4 Baum-Welch algorithm

To determine the parameters of a HMM it is first necessary to make a rough guess .Afterword's in the maximum likelihood sense more accurate parameters can be found by applying the so-called Baum-Welch re-estimation formulae.

Forword-Backword algorithm: Let the forward probability $\boldsymbol{\alpha_j}$ **(t)** for some model ***M*** with ***N*** states be defined as

$$\boldsymbol{\alpha_t(t) = P(o_1, \ldots, o_t, x(t) = j|M}$$

That is, $\boldsymbol{\alpha_j}$ **(t)** is the probability of observing the first ***t*** speech vector and being in ***j*** at time ***t***. This forward probability can be efficiently calculated by the following recursion.

$$\boldsymbol{\alpha_j(t) = \sum_{i-1}^{N-1} \alpha_i\,(t-1)a_{ij}]b_j(o_t)}$$

\

This recursion depends on the fact that the probability of being in state ***j*** at time ***t*** and seeing observation o_t can be deduced by summing the forward probability for all possibility for all possible predecessor states ***i*** weighted by the transition probability. The initial conditions for the above recursion are

$$\boldsymbol{\alpha_1(1)=1}$$

$$\boldsymbol{\alpha_i(1)= a_{1i}b_i(\,o_1)}$$

For **1**< ***j*** < ***N*** and the final condition is given by

$$\boldsymbol{\alpha_N(T) = \sum_{i-2}^{N-1} \alpha_i\,(T)a_{iN}}$$

The forward probability $\boldsymbol{\beta_i(t)}$ is defined as

$$\beta_i(t) = P(o_{i+1},\ldots o_T|x(t) = j, M).$$

As in forward case, this backward probability can be computed efficiently using the following recursion.

$$\beta_i(t) = \sum_{i=2}^{N-1} a_{ij} b_j (o_{i+1}) \beta_j(t+1)$$

With initial condition given by

$$\beta_i(T) = a_{iN}$$

For $1 < i < N$ and final condition given by

$$\beta_1(1) = \sum_{i=2}^{N-1} a_{ij} b_j (o_1) \beta_j(1)$$

$$\alpha_i(t)\beta_i(t) = P(O, x(t) = j|M)$$

SOFTWARE TEST RESULTS

The following test were performed by taking 20 speech sample spoken by four different speaker for each of the word "One", "two", "three", "four".

Recognition Accuracy (%)

Word Spoken As	word recognized as "One"	"Two"	"Three"	"Four
"One"	55	5	15	25
"Two"	30	35	15	20
"Three"	20	10	50	20
"Four"	20	10	15	55

5. CONCLUSION-

In this paper, we have presented a novel HMM-based framework for estimation of unreliable spectrographic data. We utilize hidden Markov models to reconstruct corrupted spectral components based on reliable information, unreliable observations model, to improve noise robust speech recognition.

ACKNOWLEDGEMENT-

Gracious help and guidance from various sources contributed much towards successful completion of this Dissertation work. I owe my sincere thanks towards my Project Guide Prof. Vikram A. Mane for his constant guidance during my Dissertation and for encouraging me to do Selective dissertation work.

REFRENCES-

Journal papers

[1] Bengt J. Borgström "HMM-Based Reconstruction of Unreliable Spectrographic Data for Noise Robust Speech Recognition*" IEEE transaction on audio, speech, and language processing*, vol. 18, no. 6, August 2010

[2] W. Kim and R. Stern, "Band-independent mask estimation for missing feature reconstruction in the presence of unknown background noise"*in Proc. ICASSP*, 2006, pp. 305–308.

[3] B. Raj and R. Stern, "Missing feature approaches in speech recognition," *IEEE Signal Process.* Mag., vol. 22, no. 5, pp. 101–116, Sep.2005.

[4] B. Raj, M. L. Seltzer, and R. M. Stern, "Reconstruction of missing features for robust speech recognition,"*Speech Commun*., vol. 43, pp.275–296, 2004.

[5] R. Martin, "Noise power spectral density estimation based on op-timal smoothing and minimum statistics," *IEEE Trans. Speech Audio Process*., vol. 9, no. 5, pp. 504–512, Jul. 2001.

[6] B. Strope and A. Alwan, "A model of dynamic auditory perception and its application to robust word recognition," *IEEE Trans. Speech Audio Process*., vol. 5, no. 5, pp. 451–464, Sep. 1997.

[7] L. R. Rabiner, "A tutorial on hidden markov models and selected applications in speech recognition,"*Proc. IEEE*, vol. 77, no. 2, pp. 257–286,Feb. 1989.

BOOKS-

[1]Chris Rowden "Speech Processing" ,1991

