Chapter - 6

Continuous Speech Recognition using DNN-HMM Classifier

In the previous chapters, implementation of optimized model parameters was performed using MFCC approach which provides marginal improvement in the modeling stage. The combinations of heterogeneous or hybrid feature vectors alone or with multiple taper approaches, and PCA based filter bank feature vectors were analyzed in consecutive chapters. It would improve the performance of the front and back end of the system in training and testing phase. The study is made on isolated words individually using GA+HMM or DE+HMM modeling classifiers. In this chapter, we demonstrate the use of GMM and DNN on HMM using two modeling units: monophone, and triphone. LDA+MLLT and fMLLR+SAT are further integrated to study the benefits of feature projection and speaker adaptation for continuous and connected Punjabi dataset. fMLLR+SAT on MFCC and GFCC feature vector with DNN+HMM modeling stage produce better results as compared to other modeling techniques.

6.1 Introduction

Typical ASR systems process the input speech signal into its phones in two steps - feature extraction and storage of this information in modeling classifiers. Literature (Rabiner, 1989) shows that most powerful section of research in ASR is based on HMM approach for tackling of instability of an input speech utterance. The researchers process the speech signal by refining, optimizing or integrating them to avoid performance decay in later modeling classifiers (Aggarwal, R. K., & Dave, 2012; Kwong et al. 2001; Rabiner, 1989). It has the issue of mandatory parameter evaluation, proper modeling of

speech unit and estimation of minimum number of observations. These factors are tackled by processing them on large corpora using HMM technique. Subsequently, researchers used formation of hybrid classifiers - GMM-HMM, DNN-HMM and ANN-HMM (Liu, S., & Sim, 2014; Maas et al. 2017; Mohamed, A., & Nair, 2012)**.** GMM is connected with HMM for building a bridge between its states and acoustic signal. It terminates the problem of overfitting for training dataspace (Hinton et al. 2012). GMM introduced the challenge of improper modeling of its information on or near to a nonlinear manifold for available dataspace. The functionality of GMM is replayed using DNN trying to attain maximum accuracy as analyzed earlier (Schmidhuber, 2015).

The studies discussed in (Chen, X., & Cheng, 2014; Hinton et al. 2012) showcased the benefits of DNN-HMM when employed at modeling stage on TIMIT and Mandrian speech dataset. In this study, the integration of two different combinations are used to handle the mismatch conditions, to implement the low rank feature projection and supporting speaker adaptability at front end of the system. An attempt is made to process the front end features on two modeling units - monophone and triphone. It can be combined with speaker adaptive and feature projection approaches. Later feature classification is performed with GMM-HMM, and DNN-HMM at the back end. Rest of the chapter is organized as follows: Section 6.2 outlines the functionality of various hybrid acoustic modeling classifiers. The description of an overview of heterogeneous system architecture strategy is presented in Section 6.3. Punjabi dataset design strategy is discussed in Section 6.4. Section 6.5 consists of comparative analysis of different sub-system along with performance evaluation and discussion. The summary of proposed system is describes in Section 6.6.

6.2 An overview of acoustic modeling classifier

6.2.1 GMM-HMM model

In real practice, the observation probabilities are generated in acoustic modeling

approach, based on traditional HMM technique. It helps in acquiring of words that are stored in the basic unit called phoneme. Combination of these phonemes helps in formation of a particular word which is employed as a sentence or phrase. The modeling of observation probabilities is made possible through multivariate GMM technique (Povey et al. 2011)**.** The estimation of observation vector for D dimension is possible through multivariate Gaussian density function with Eq. (6.1).

$$\mathcal{N}(O_l, \emptyset_{pn}, C_{pn}) = \frac{1}{\sqrt{(2\pi)^D |C_{pn}|}} \exp\{-\frac{1}{2}(O_l - \emptyset_{pn})^T C_{pn}^{-1} (O_l - \emptyset_{pn})\} \quad (6.1)$$

Where O_l represents the observation of symbol at time l, $\emptyset_{pn}$ derives the value of mean and C_{pn} depicts its covariance matrix for n^{th} Gaussian component of a p^{th} state subsequently.

A number of additional model parameters of HMM (such as weight of mixtures as well as state transition probabilities) are trained with the help of Baum-Welch re-estimation algorithm (Rabiner & Juang, 1993).

6.2.2 Deep Neural Network model

Earlier a GMM technique is adopted for statistical training of an acoustic model. It is generated on or near the non-linear manifold problem in an input data space. In neural network approach, this parameter is tackled using single hidden layer which is not a good practice. Improving one parameter and keeping other parameter with a constant value is mostly performed while training of each stage of an acoustic model. DNN is currently employed using a series of fully interconnected hidden layers. It helps in transformation of an input speech vector (x). DNN tries to estimate the output class (N) from a probability distribution $\hat{y}$. It also acts as a function approximation, helping in calculation of conditional distribution $\rho(y|x)$. It processes the function through j layers each of which is connected to its output layer. These hidden layers followed by output

layer helps in resolving the issue of an acoustic variability. Each layer is connected with a weight *(w)* and a bias vector (b) using a logistic function in Eq. (6.2):

$$y_j = logistic(x_j) = \frac{1}{1+e^{-xj}},\ x_j = b_j + \sum_i y_i w_{ij} \quad (6.2)$$

The output layer of network is finally required to generate a proper formation of its probability distribution for output sections**.** These output layers act as an input to various number of HMM states. Each state helps in modeling of phone on both sides with the help of a triphone modeling unit. A large number of tied states are produced as an output from its triphone HMM while processing a large set of training dataset. Instead of processing the whole training dataset, it is segregated into small chunks that are operated before updating their weight and bias as per gradient. The gradient can be enhanced using a momentum coefficient point lying in the range of $0 < \alpha < 1$ for smoothening of its gradient. It helps in the estimation of a minibatch (t). DNN tries to reduce over-fitting over large training corpora. The variation in weight parameters are needed by discontinuing the learning activity before it reaches to a unacceptable performance condition. Initially small varied values are allocated to weight of an initial layer, later joining them to its hidden layer. It avoids in obtaining same gradient points for its entire hidden layer. This DNN formation is found to be beneficial in speech recognition application through setting of their parameters: number of layers and the number of associated units. In this chapter, building an acoustic model on DNN-HMM and GMM-HMM have been described to analyze the improvement performance for Punjabi language.

6.2.3 DNN-HMM model

In earlier years, (Deng et al. 2013) DNN-HMM studies focused on building of a speech recognition technology. The HMM paradigm implements the DNN model that estimates the posterior probabilities with processing of a Context Independent (CI) tried state model (senone). A non linear transformation is applied using various numbers of hidden

layers for a feature vector (x_t) corresponding to a CD window frame. Furthermore, a gain in labeling of senone is obtained through force-alignment technique. It is implemented on a large set of training dataset using traditional GMM-HMM system. Parameters of DNN are processed through gradient descent function that employs back propagation algorithm. It focuses on softmax output layer, framed from a number of nodes of different classes, equal to their corresponding senones. The classification of multi-class in DNN is achievable with a relation of softmax non-linearity using Eq. (6.3).

$$p(x_t) = \frac{\exp(x_t)}{\Sigma_k \exp(x_p)} \tag{6.3}$$

Where p denotes the index value for each class. The likelihood of senone $p(x_t)$ is determined in HMM through a chain of characteristics of information of the speech model. Certainly, the output of scaled likelihood is estimated from posteriors on the test dataset in decoding state. It evaluates recognized hypothesis on the basis of scaled likelihood.

6.3 An overview of heterogeneous system architecture strategy

In this section, Punjabi ASR system is demonstrated for continuous sentences or connected words using an efficient approach of GFCC and MFCC at front end. These two feature extraction techniques plays a key role in generation of robust features that needs to be employed in subsequent stages of training and testing phase of the proposed system. These approaches are mainly responsible for better training for generation of corresponding text in decoding phase. To quantify the affect of mismatch among train & test conditions and to generate the refined feature vectors for improved classification, two feature and two context modeling unit (monophone and triphone) are used. The monophone (M1) is employed as an input to other modeling unit such as triphone. For better contrast, the output of triphone modeling is explored by delta-delta training (M2)

that act as an input to LDA+MLLT (M3) method, and finally employed for fMLLR+SAT (M4) implementation. These methods are developed with two acoustic modeling approaches like GMM+HMM and DNN+HMM as represented in Figure 6.1 having two sub-systems.

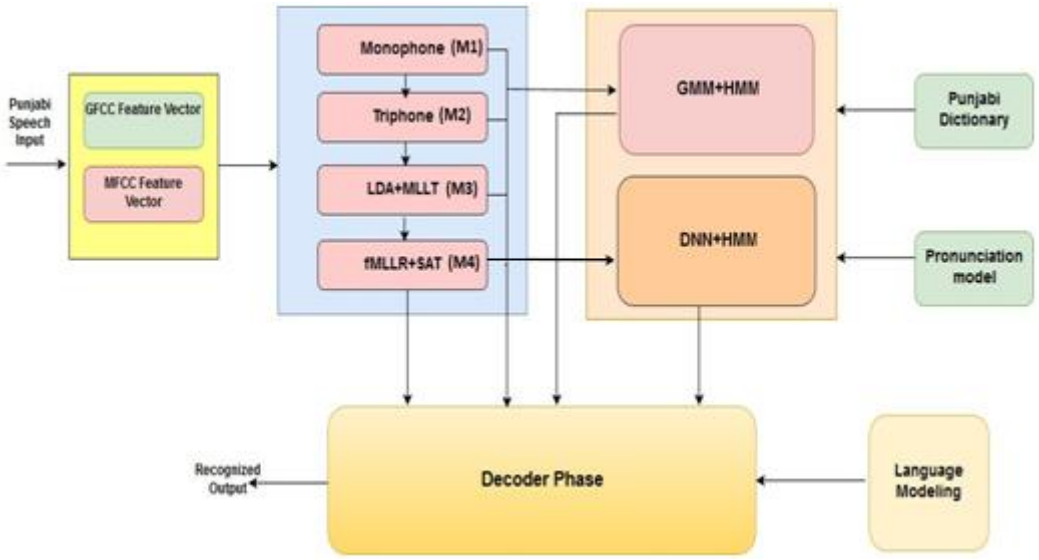


Figure 6.1: Flow diagram of GMM+HMM and DNN+HMM modeling classifiers integrated with monophone and triphone modeling units

6.3.1 Baseline system implementation using GMM-HMM and DNN-HMM based acoustic modeling

For the development of GMM-HMM and DNN-HMM system, 39 features are employed for training and testing of the system using MFCC or GFCC. It involves 13 static features with energy of their first and second changing derivatives. The implementation is performed on 25 ms window with a frame shift of 10 ms. These techniques discard huge information of extracted feature vectors from the input waveform. The extracted information of feature vectors is sliced from 9 frames with combination of $+-4$ frame on both sides of current frame. To further process this information low rank features reduction method is used to project them into 40 feature vectors. The projected vectors are processed with MLLT method to produce

decorrelation. The training of these vectors is integrated with GMM-HMM classifiers. The training is started with state tying of a decision tree for triphone model, framed from the output of monophone model (M1). The triphone model works on 3 states of HMM model. Its diagonal covariance matrix contains either a value of 16 or 32 thus helping in storing silence and short pauses. The observation sequence (O) of processed information is estimated using Baye's theorem. It calculates the maximum probability corresponding to observations in hidden layers. These probabilities are determined over whole trained model. The output information is evaluated in decoding phase with the help of maximum probability. These probabilities are calculated over hidden states to generate desired observation sequence using Viterbi algorithm.

In another acoustic model DNN-HMM, the process of normalization is applied with feature space MLLR (fMLLR) approach. It works on $40 * 41$ parameter dimension estimated in speaker adaptive training approach and is employed on GMM based system. During training, DNN model is implemented with a varied range of hidden layer from 1 to 7. System attains maximum word accuracy at a value of 5 hidden layers. Additional parameters like learning rate is fixed at a matrix–value of 0.015 and number of epochs are kept upto 20. The process is initiated on two minibatch such as 512 and 1024. In testing phase, similar process of fMLLR transformation is applied which is used in decoding module. At the decoding module, it can be analyzed that fMLLR is found to be very influential in mismatched test and train conditions. The system simulation is performed on similar size of dictionary adopted in GMM-HMM based system.

6.4 Punjabi dataset preparation

In this study, a self recorded corpus of Punjabi language is constructed for connected and continuous Punjabi sentences or words. The data comprising of sentences of two types: phonetically rich sentences and simple sentences. Accordingly dataset is broadly

categorized into sets: dataset1 and dataset2. The dataset1 is framed of connected words. It is generated by 13 speakers, who produced a total of 21,764 speech sample of 3 second in duration. Each dataset1 is recorded at $44kHz$ in clean or real environment. The dataset2 consists of utterances of 28 speakers. The phonetically rich sentences are produced by 13 speakers speaking 422 unique sentences thus building a corpus of 3611 sentences. Each sentence in this category is of $3-7$ second duration. Rest of the data is recorded by 15 speakers who spoke collected 5000 words training simple sentences. The dataset2 is recorded at a sampling rate of $16\,kHz$ through Sony Dictaphone recorder. The system performance is tested in form of WER.

6.5 Results and discussion

The system implementation is done using Kaldi toolkit version 4.3.11 for development of Punjabi connected and continuous ASR system. During development stage a number of experiments are performed with the variations in the values of feature dimension, hidden layers, and number of Gaussian mixture. These parameters are varied on different modeling classifiers (DNN-HMM and GMM-HMM) and front end approaches (GFCC, MFCC) on various modeling units (M1-M4). These parameters are varied to obtain an optimal value on linux platform (Ubuntu 14.04).

Firstly the training dataset is prepared from the corpus. Transcriptions of spoken utterances are stored in a text editor files. Phonelist, lexicon, dictionary and filler files are also created before training of the system. After collection of raw input files, training of the system is initiated. Feature extraction stage is common in both phases. Initially, different feature vector are extracted using two approaches - GFCC and MFCC. This information is further processed using monophone, and triphone models. The output is applied to LDA, and fMLLR-SAT either alone or in their combinations to generate fused feature vectors. The test results are applied in a similar way using 10 fold cross validation method where 10% data is used for training and rest is stored for testing

purpose.

6.5.1 Performance evaluation on different modeling units

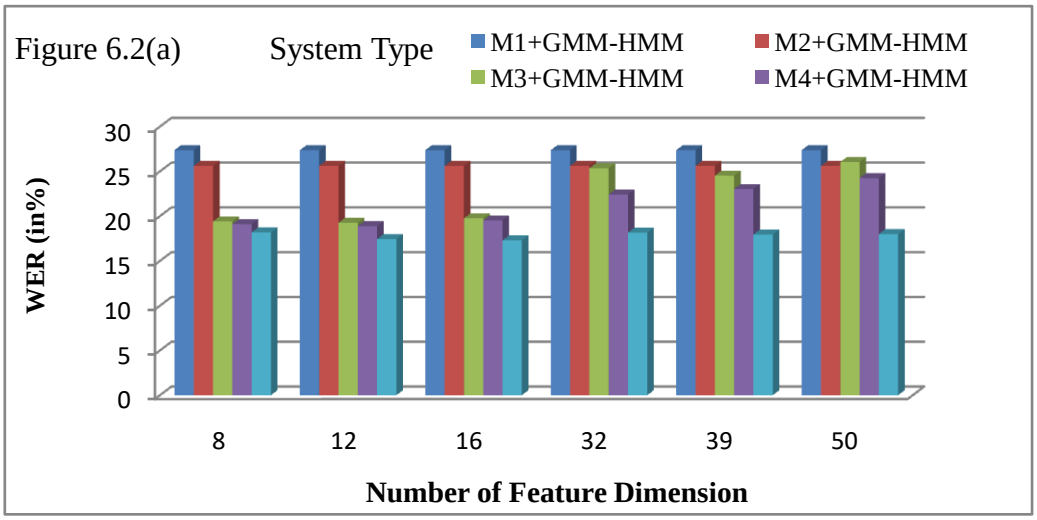
In this section, five different training models are used with two different modeling classifiers. These training models are based on two major modeling unit - context dependent and context independent. When an input speech signal is processed in initial stage it passes through two different front end approaches - MFCC and GFCC using M1 model with GMM-HMM classifier. The performance was not found to be acceptable and so needs to be improved with CI model. This model uses 120 num pdf (probability density function) for formation of a triphone model on the same front end and classifier approach. It boosts up the performance in both dataset1 and dataset2. The experiments are performed on both datasets. To reduce the WER, CI models are used after processing using M3 and M4 models along with varying value of num pdf. System performance is enhanced with speaker adaptive method and mismatches approach in M4 model and is examined using DNN-HMM classifier. As the size of corpus increases, DNN-HMM model outperforms as shown in Table 6.1. The proposed system achieves very high value of WER with M1 model and very low with M4 model.

Table 6.1: Comparison of WER (%) obtained by various modeling unit (M1 to M4) for different acoustic classifiers

System Type	**Feature Type**		
	GFCC	**MFCC**	**MFCC**
	dataset2		**dataset1**
M1+GMM-HMM	46.3	14.11	46.74
M2+GMM-HMM	82.16	19.14	46.71
M3+GMM-HMM	73.39	13.08	46.09
M4+GMM-HMM	34.4	7.01	44.94
M4+DNN-HMM	24.67	5.22	39.73

6.5.2 Performance evaluation on varying range of feature dimension with LDA approach

In the Table 6.1, it is observed that the DNN-HMM based Punjabi ASR system is found better in comparison to GMM-HMM based classifier. To reduce the value of retrieved WER the feature dimension component is analyzed to study the effect on default feature vector output. Feature dimensions are varied from a range $(8, 12, 16, 32$ and $39)$ to analyze their effect on WER. The analysis is shown in Figure 6.2 (a) and (b).



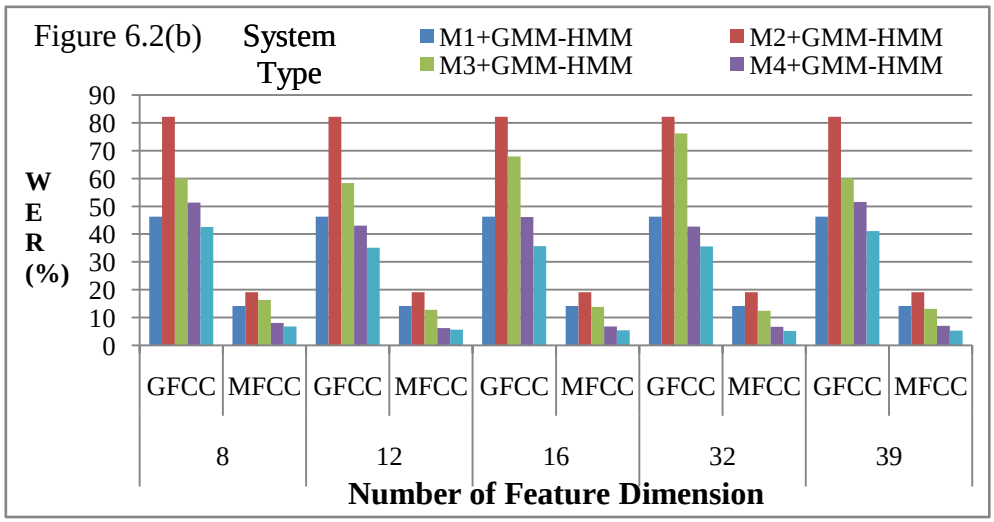


Figure 6.2(a) and (b): Comparison of WER (%) for varying feature dimension with different system type

The time slicing of MFCC is found to be beneficial in comparison to front end feature vector approaches. It is made possible through introduction of feature projection using LDA approach in M3 model. It helps in time-splicement of feature information, extracted using MFCC or GFCC approach. It also increases class separability among various classes. It makes this separation possible through projection of 117 entries into 40 feature dimension decreasing its WER. It is clear from Figure 6.2 (a) and (b)that feature projection scheme showcased significant performance improvement. A low WER for a feature value of 16 is examined using MFCC technique on dataset1 corpora. In testing dataset2, least WER is obtained on a feature value of 32 in case of MFCC and 16 in case of GFCC.

6.5.3 Performance evaluation with varying number of hidden layer

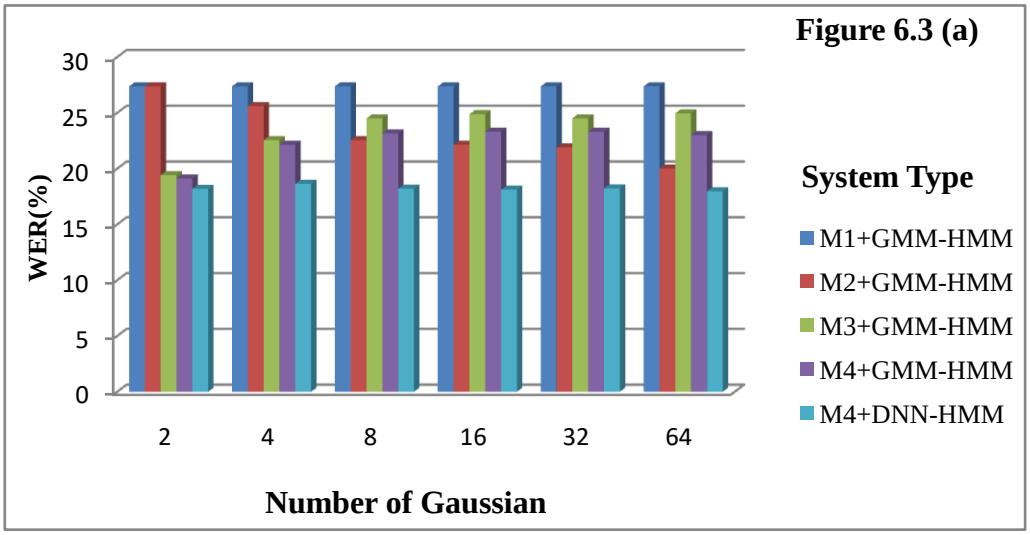
In case of matched train and test conditions an improvement of $4-5\%$ WA and $1-3\%$ WA is obtained in two different datasets using DNN-HMM and GMM-HMM classifiers. Other researchers have constructed other ASR engines (Mitra et al. 2014). Subsequently investigation is done with varying number of hidden layers and their associated hidden units in two minibatches. Tanh function is employed to process the non-linear hidden information actuated from normalized feature data. WER profiles for different hidden layer range from 1 to 7. These layers are associated with different ranges of units, related to each layer having values $512, 1024, 2048,$ and 3074. These components are processed on various number of frames $(7, 11, 15, 27,$ and $37)$. It investigates the proposed system from its shallow area to its broad area as showcased in Table 6.2.

Table 6.2: Comparison of WER(%) profile for varying number of hidden layer for DNN-HMM classifier

Dataset Type	**Number of Hidden Layer**						
	1	2	3	4	5	6	7
Dataset1	18.01	17.97	17.85	17.58	17.66	**17.53**	17.54
Dataset2	5.32	5.41	5.32	5.41	**5.22**	5.29	5.51

The results analysis on dataset1 found minimum WER at layer value 6 and at layer value 5 in case of dataset2.

6.5.4 Performance evaluation with varying Gaussian mixture



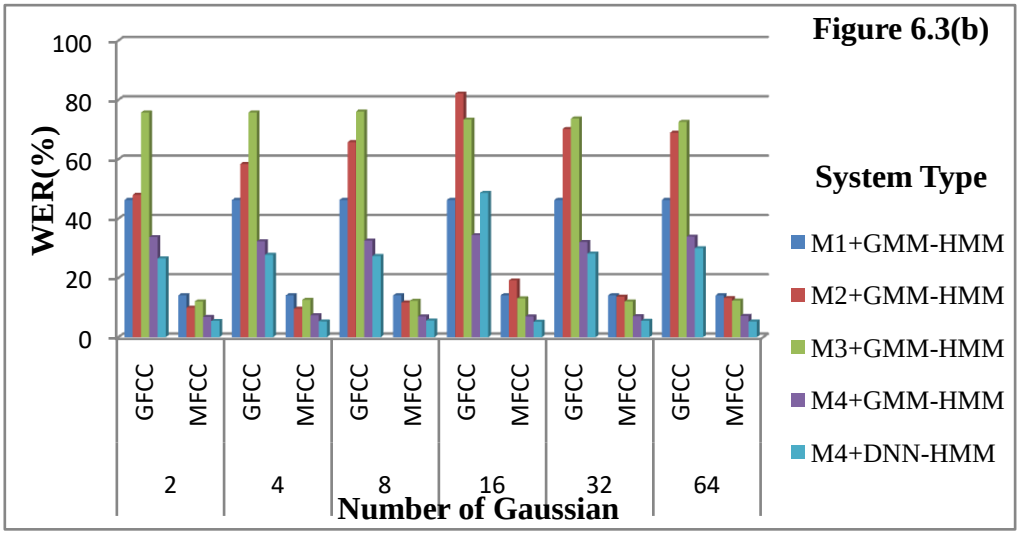


Figure 6.3 (a) and (b): Comparison of WER(%) profile for varying Gaussian mixture obtained for different acoustic modeling classifiers

The decay in performance recognition of baseline ASR system occurs due to improper selection of its Gaussian value. These parameters are associated with various GMM to depict a single Gaussian holding similar probability density. The system performance is

analyzed on varying ranges of Gaussian and with MLLR+SAT approaches showing into performance improvement with various front end back end approaches as shown in Figure 6.3 (a) and (b).

The investigation is performed initially on M1 and M2 models that didn't represent any significant change. But system shows better performance improvement with M3, M4 model on DNN and GMM based modeling classifiers.

Using dataset1, minimum WER is achieved with a mixture value of 64 in case of MFCC approach. In dataset2, GFCC attains low WER at a value of 64 Gaussian mixtures but in case of MFCC, it achieves at a value of 16 Gaussian mixtures on DNN-HMM classifier.

Finally optimal values of all above parameters are selected such as number of feature dimensions are set to 16, number of Gaussian attains maximum accuracy at a value of 16 and numbers of hidden layers are tuned to 6.

6.6 Summary

In this chapter two front end noise robust approaches (GFCC and MFCC) were introduced with DNN-HMM modeling classifier in context of Punjabi continuous and connected speech corpora. To evaluate the performance, several parameters are varied to avoid the degradation due to its default values. These experiments are conducted on different modeling units - context dependent and context independent. Removal of mismatch condition barrier, decorelation of vector information in practical speech sample and their low rank feature projection has been implemented. During formation of a real ASR engine none of the approaches individually produces better results in all the circumstances. In this chapter different combinations are analyzed to overcome the speaker variation, overfitting, mismatch condition and decorelation components. Results show that DNN based ASR engine produce improved word recognition performance if they are combined with complex front end feature processing. The use of proper

training method and extraction of efficient feature vectors in testing phase outperforms the baseline model. To enhance the performance of speech recognition, work can be extended with RNN and LSTM approaches in acoustic and language modeling stages.