MACHINE LEARNING FOR IDENTIFICATION OF REGIONAL LANGUAGES OF PAKISTAN FROM SHORT UTTERANCES

by

Ammara Imtiaz

Supervised By

Dr. Hanif Zauq

Co-Supervised By

Dr. Sajid Saleem

Submitted for partial fulfilment of the requirements of the degree of MSCS to the Faculty of Engineering and Computer Science

NATIONAL UNIVERSITY OF MODERN LANGUAGES,

ISLAMABAD

MARCH 2019

MACHINE LEARNING FOR IDENTIFICATION OF REGIONAL LANGUAGES OF PAKISTAN FROM SHORT UTTERANCES

by
Ammara Imtiaz

Supervised By
Dr. Hanif Zauq

Co-Supervised By
Dr. Sajid Saleem

Submitted for partial fulfilment of the requirements of the degree of MSCS to the Faculty of Engineering and Computer Science

NATIONAL UNIVERSITY OF MODERN LANGUAGES,

ISLAMABAD

MARCH 2019

NATIONAL UNIVERSITY OF MODERN LANGUAGES

FACULTY OF ENGINEERING AND COMPUTER SCIENCE

THESIS AND DEFENSE APPROVAL FORM

The undersigned certify that they have read the following thesis, examined the defence, are satisfied with overall exam performance, and recommend the thesis to the Faculty of Engineering and Computer Sciences.

THESIS TITLE: Machine Learning for Identification of Regional Languages of Pakistan from Short Utterances

Submitted By: Ammara Imtiaz Registration #: MSCS-S-16-009

Master of Science
MSCS

Computer Science
Name of Discipline

Dr. Hanif Zauq
Name of Research Supervisor
Signature: ____________________________

Dr. Sajid Saleem
Name of Co-Supervisor
Signature: ____________________________

Dr. Muhammad Akbar
Name of Dean (FE&CS)
Signature: ____________________________

Brig. Muhammad Ibrahim
Name of Director General (NUML)
Signature: ____________________________

22nd March, 2019

CANDIDATE DECLARATION

I declare that this thesis entitled "*Machine Learning for Identification of Regional Languages of Pakistan from Short Utterances*" is the result of my own research except as cited in the references. The thesis has not been accepted for any degree and is not concurrently submitted in candidature of any other degree.

Signature:	
Name:	Ammara Imtiaz
Date:	March 22, 2019

ABSTRACT

An interesting problem in speech analysis is automatic identification of languages from short utterances. Language Identification (LID) related research is gaining importance. It tries to overcome communication barrier among the speakers in sharing information with each other in their native languages. LID has wide range of applications in spoken languages such as language to language translation, language understanding, telephone based system, voice dialling, tourism, e-health, and distance learning.

The thesis focuses on application of LID in classifying major regional languages of Pakistan. These languages are Urdu, Balochi, Punjabi, Pashto and Sindhi. Urdu is national language of Pakistan whereas other four are regional and provincial languages. The thesis proposes a new method for LID, which is referred to as Nearest Neighbour Feature Matching (NNFM) strategy to efficiently classify the languages of Pakistan in recordings.

To identify languages with NNFM, a three step process is implemented. In the first step, Mel Frequency Cepstral Coefficients (MFCC) algorithm is applied to the speech samples of training and test set to extract speech features. The extracted features are then normalized such that the magnitude of each feature becomes equal to unity. In the second step, the normalized features of a test speech samples are matched with features of all the speech samples of the training set using dot product. The dot product produces maximum values where a test feature perfectly matches with its Nearest Neighbour (NN) feature in a speech sample of the training set. Then the maximum dot product values are obtained. The maximum values are averaged over all the features of the test speech sample. The average value quantifies the similarity of the test sample with the samples of the training set. The training sample that gives maximum average value is selected and its features, which are referred to as NN features are used to replace the features of the test samples.

In the third step, Gaussian Mixture Model-Universal Background Model (GMM-UBM) is trained on the training samples. The GMM-UBM computes a General Language model and a specific language model. The NN features are then provided to GMM-UBM for prediction of a language in the test sample. Based on the two models GMM-UBM computes log-likelihood. The language category of the training set that gives the maximum log likelihood is selected as a predicted language for the test sample.

Experiments are performed on Corpus of Regional Languages (CRL) of Pakistan. The experimental results show that GMM-UBM classifier with proposed NN-FM method gives better results than GMM-UBM without NNFM method. The experimental results show that GMM-UBM without NNFM achieves average 48%, 50%, 52% and 53.3% accuracies on test utterances of duration three, five, ten and fifteen seconds, respectively. Whereas with NNFM, GMM-UBM achieves average 56.7%, 60.7%, 63.3% and 65.3% accuracies, on three, five, ten and fifteen seconds test utterances, respectively. The proposed NNFM efficiently improves the accuracy of GMM-UBM by almost 8.7% to 12%. Experiments on a Call friend corpus consisting of six different international languages are also performed the experimental results show that NNFM also significantly improves the performance of GMM-UBM.

Keywords: Language Identification, Nearest Neighbour Feature Matching, Speech Signal, Speech Features, Gaussian Mixture Model

DEDICATION

This thesis work is dedicated to my parents and my teachers throughout my education career who have not only loved me unconditionally but whose good examples have taught me to work hard for the things that I aspire to achieve

ACKNOWLEGEMENT

First of all, I wish to express my gratitude and deep appreciation to Almighty Allah, who made this study possible and successful. This study would not be accomplished unless the honest espousal that was extended from several sources for which I would like to express my sincere thankfulness and gratitude. Yet, there were significant contributors for my attained success and I cannot forget their input, especially my research supervisors, Dr. Hanif Zauq and Dr. Sajid Saleem and my teacher Dr. Fazli Subhan who did not leave any stone unturned to guide me during my research journey.

I shall also acknowledge the extended assistance from the administrations of Department of Computer Sciences who supported me all through my research experience and simplified the challenges I faced. For all whom I did not mention but I shall not neglect their significant contribution, thanks for everything.

Table of Content

List of Tables

List of Figures

LIST OF ABBREVIATIONS

LPC	Linear Predictive Coding
LPCC	Linear Predictive Cepstral Coefficient
MFCC	Mel-frequency Cepstral Coefficient
SDC	Shifted Delta Coefficient
KNN	K-Nearest Neighbour
SVM	Support Vector Machine
I-vector	Identity Vector
GMM	Gaussian Mixture Model
GMM-UBM	Gaussian Mixture Model-Universal Background Model
NNFM	Nearest Neighbour Feature Matching
EER	Equal Error Rate

CHAPTER 1

INTRODUCTION

1.1 Overview

This chapter presents an introduction of the research topic. It contains aims and objectives of the research. Applications of Language identification (LID) are also discussed in this section. Different implementation stages of identification LID system is also presented and along with LID framework.

1.2 Language Identification System

LID refers to the process of automatically identifying the language spoken in a speech sample [1]. LID is considered as an important bridge between human to human and human to machine communication [2]. The human intelligence is the most accurate identification system of the language compared to machines, because humans are born with the capability of differentiating different spoken languages. Within few seconds human can identify the language they already know. Even though they are not familiar with the language they can predict the languages in the speech sample to its similar language they know. In case of machines automatic LID aims to replicate this human ability into digital computers [3]. Figure 1.1 illustrates the main components used in implementation of LID.

1.3 Language Identification Components

The process of LID consists of three major components which are database collection, features extraction and machine learning. Database collection plays a vital role to achieve better efficiency and performance of the system. Informative database leads the system to achieve better results. Database consists of speech signals and samples collected from different speakers. Process of feature extraction is also the major component of LID. Selection of feature algorithm type to detect distinct features from the recorded samples and improves the accuracy rate of LID.

The last step is machine learning. Different machine algorithms can be used the one which gives the best results is used. All the components contribute to efficiency and performance of LID system. In Figure 1.1 process of LID is discussed.

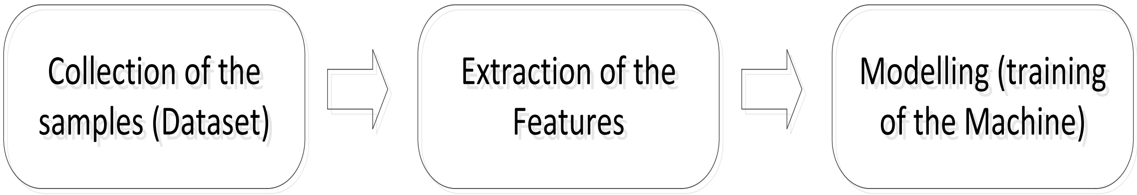

Collection of the samples (Dataset)
Extraction of the Features
Modelling (training of the Machine)

Figure 1.1: Main Components of LID

1.4 Language Identification Applications

During the past three decades LID has become the main technology in many speeches related applications. This includes biometric verification [4], voice dialling [5], travel information & automated dialogue system [3], multilingual speech recognition &speech retrieval [5], spoken language translation system [6], call routing (language routing) & emergency assistance [7], language interpretation, buying services from international markets & tourism [8], weather forecasting [9]. Language is a natural mode of communication between human and machine [10]. Exchange of information between man and machine is held through devices such as keyboard, mouse, printer, scanner, monitor, microphone, webcam and plotter [5].

Use of these devices some time is not easy to operate by a common person without any assistant. Interfacing the spoken language with computers provides an easy tool for common person as compared to computer devices [11]. Speech interfacing provides a user friendly interface for communication between man and machine because speech is more expensive and effective than text. Table 1.1 summarizes few applications of LID and corpuses used for these applications.

Table 1.1: Applications of LID

Sr. No	Application	Corpus
1.	Commerce and emergency [7]	English, Mandarin, Spanish, Japanese and German etc
2.	Classification of music [8]	Chinese and Mandarin
3.	Commercial speech translation service [12]	English, Vietnamese and Korean
4.	Language identification [13]	English and Mandarin
5.	Multilanguage capability (Travel information, buying trades) [14]	English, German, Spanish, Japanese and Mandarin
6.	Speech synthesis [15]	Japanese and English
7.	Symbolic plan recognition [16]	Human activities
8.	Spoken language translation [17]	English, Spanish is used for translation
9.	Semantic learning for mobile robot [18]	Human sound in European languages
10.	Sensor integration [19]	Audio-Video music is collected in American English an British language

In educational sector LID is used in the distance learning and for teaching of foreign languages to students and also to handicapped people (i.e blind people). LID is also used in the robotics [20] medical sector (health care, medical transcription) [14], intelligence and security [21]. This thesis focuses on LID applications in classifying the majorly spoken languages of Pakistan from short utterances and speech samples. It can be used in telephone based system, tourism, retrieval of audio files in certain languages.

1.5 Language Identification Process

Usually LID system delivers two types of information: Speaker content and Voice recognition [9]. Speaker recognition focuses on only on speaker identification. Content in the voice samples carry the information about language he/she is speaking. This thesis focuses on speaker content rather than speaker recognition. For this purpose, machine learning approach plays a vital role for identification of language from the speaker voice content. Machine learning approaches has become an important component of LID [22]. These approaches consists of two sections(i) training section and (ii) testing section [20]. In the training step features are extracted from the speech signal and the machine is trained. The training phase of machine depends on training samples, speech length, speech quality and types of speech features [23]. In the testing phase the features of the test speech signal are provided to the trained machine for decision making i.e LID [24]. Then accuracy of the machine is analyzed through number of speech samples it correctly identifies. Normally the training set and the test set includes recordings from different speakers so the primary issue is to extract the differences of the languages rather than content, speaker and environment [25].

Exploring the speech information from the test speech signal is also an important factor that machine have to take care and also it is required that machine remain flexible to variations in the speech signals due to different speakers, speaking styles and noise [26]. Similarly, the computation time, number of languages and duration of the sample also affect the machine performance [27]. Machine is trained in two different levels for LID. In case of low level training, the speech features are used for training purpose. At high level training morphology and sentence syntax are also used along with features [23].The spoken languages differ from each other by words, structure and grammar [28]. This information is required to be incorporated into machine to differentiate between different languages and to provide better performance on test samples [29]. This thesis only focuses on feature based training of machine for LID. In Figure 1.2 framework of LID is shown. The first step in the framework is collection of speech samples, then working on the training samples using different feature extractor techniques. Classifiers are trained and tested on the speech samples. The speech step includes the selection of the vocabulary either the speech is text dependent or text independent and duration of the speech samples. In text dependent there is fixed vocabulary sentences or words spoken by different speakers. And in the text independent scenario there is no restriction for the speakers to choose different words and sentences for recording. This thesis focuses on implementation of LID using speech samples, which are text independent and the speaker's recordings include the sentences as per their own choice. Duration of the speech samples is defined. In the training phase feature extraction techniques such as Mel-Frequency Cepstral Coefficients (MFCC), Linear Perceptual Coefficients (LPC), Linear Predictive Cepstral Coefficients (LPCC) to extract the features from the speech samples. The features are then used in the training of classifiers such as Gaussian Mixture Model-Universal Background Model (GMM-UBM), Support Vector Machine (SVM), K-Nearest Neighbour (KNN) and tested on test samples.

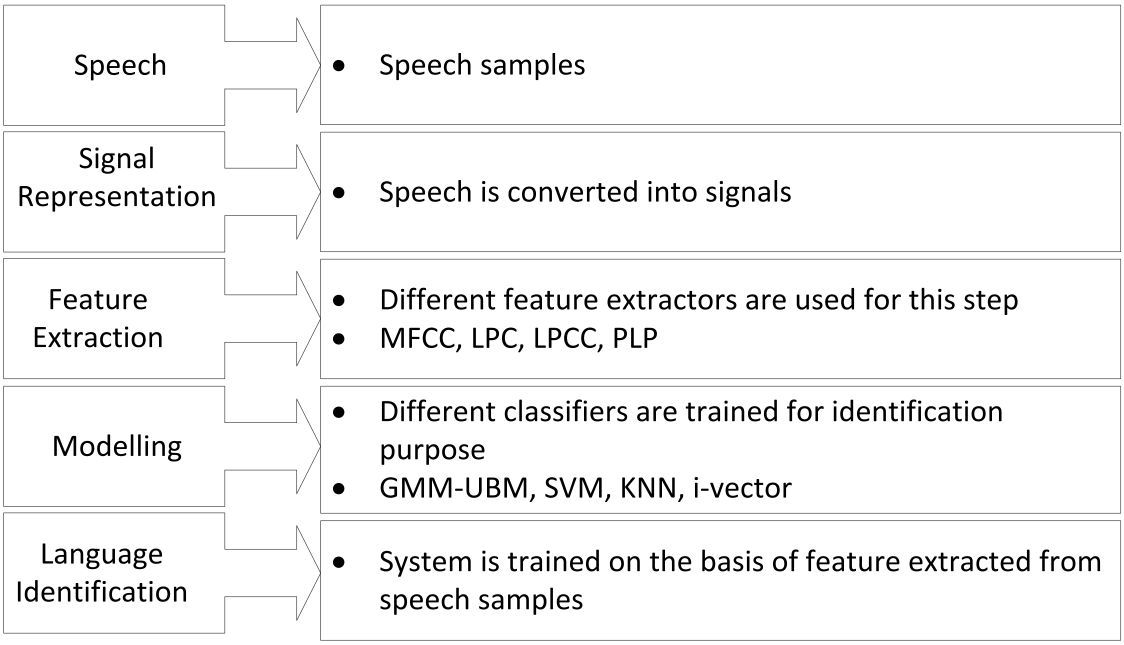

Speech
• Speech samples
Signal Representation
• Speech is converted into signals
Feature Extraction
• Different feature extractors are used for this step
• MFCC, LPC, LPCC, PLP
Modelling
• Different classifiers are trained for identification purpose
• GMM-UBM, SVM, KNN, i-vector
Language Identification
• System is trained on the basis of feature extracted from speech samples

Figure 1.2: Description of LID Process

1.6 Problem Statement

In literature, the focus of LID is on international languages like English, German, Chinese, Arabic, Japanese, Iraqi, Turkish and Indian [17, 30, 31, 32]. There are some studies that focus on local languages like Indian (Marathi, Gojri, Gujrati, Bengaliand Hindi) [17] and Afghani (Dari, Farsi) [32].There are few studies that focus on the local languages of Pakistan like Punjabi, Sindhi, Balochi and Pashto [33, 34, 35]. Some focuses on Urdu and Sindhi languages[36] . In case of Pakistani languages such as the work presented in [36] a fixed vocabulary sentences are spoken by both female and male speakers. These sentences are asked in local languages (Urdu and Sindhi), the sentences are what is your name? Where do you study etc. Audio recordings are carried out through microphone. No fixed time limit is in this research for spoken the sentences. Mel-frequency Cepstral Coefficients (MFCC) is used for feature extraction from speech samples. Support Vector Machines (SVM) is used as a classifier. It is shown that system identifies the 90% accurate result for Urdu language and 75% result for Sindhi language.

There is a need to identify the regional languages of Pakistan in text and speaker independent way and in uncontrolled environment. The regional languages are Punjabi, Pashto, Balochi, Sindhi and Urdu. The identification of languages is required to be done on short utterances from 3 sec to 15 sec. Such identification has applications in telephone based system, voice dialling and tourism.

1.7 Research questions

i. Is it possible to identify regional languages of Pakistan from short utterances?

ii. Which speech features and machine learning techniques can be used for identification of regional languages?

1.8 Thesis Contribution

The main contributions of this thesis are as follows:

i. A performance evaluation of different speech features and classifiers for identification of regional languages of Pakistan from short utterances.

ii. A new method, which is referred to as Nearest Neighbour Feature Matching (NNFM) to improve the performance of GMM-UBM based LID.

iii. A new speech corpus of regional languages of Pakistan.

1.9 Thesis Outline

Chapter 1 presented an objective of the thesis and research questions. Introduction of the LID which includes identification process, explanation of the LID and all the phases in the process of identification are discussed in this section. Different applications of LID are discussed in chapter 1. To enhance the efficiency and performance of the system, the factors effects the system performance is also explained. Chapter 2 includes the past related work to this thesis. This chapter reviews briefly language identification task and its techniques step by step from extraction of feature to modelling. Feature extractors and classifiers are discussed in the chapter 3. Different techniques of feature extraction, collection of samples, recording techniques with or without noise, controlled and uncontrolled environment is discussed in this section. Proposed system, framework and the methodology are explained in the chapter 4. Step by step explanation of the proposed system is discussed. Values of different parameters used to extract the features from the samples are shown in tabular form in this section. Chapter 5 presented the experimental results of the experiments done for the thesis. In chapter 6 future conclusion of entire thesis is explained.

1.10 Summary

Chapter 1 presents an introduction to the LID system. It explains the use of LID and features and classifiers commonly used for language research process. Aim of the research and the problem statement is also a part of this chapter. It also discusses the research questions and contribution of this thesis. Finally, an outline of the thesis was presented.

CHAPTER 2

RELATED WORK

2.1 Overview

This chapter presents literature review. It gives an overview of classical and state of the art methods for LID. All phases, methods and technologies used from time to time for LID are discussed. Different stages of LID such as Language corpus, feature extraction methods and machine learning techniques are explored in this chapter.

2.2 Related Work

A process of LID aims at identifying languages in speech signals. LID has made it possible for computers to detect human voice and understand the human language [37]. Detecting and understanding the human language is required to identify the different utterances. Primary goal of LID is to develop approaches and systems for input to machine [38]. The process of LID surrounds a wide range of challenges including recognition of sentences and words in the speech sample, extraction of authentic information and understanding of speech for machine to interfere [39]. LID incorporates three main stages namely, corpus collection, feature extraction, training [40]. All these three stages are important and selected for LID on experimental basis. Features extraction is primary and crucial step. Feature extraction is performed in two steps that is speech analysis and compilation of features in LID [38]. Machine learning is the last step where classifier is trained on extracted features, and the test utterances are validated.

Performance of LID system depends on parameters in which selection of features is indispensable [6]. In order to discover relevant information form input speech, it is important to have approaches, techniques and methods for capturing the required data or features. Different approaches are there to capture the useful and relevant information from speech samples. The elementary difficulty for LID is the variability in the speech signals due to different speakers [4]. Recorded samples vary from speaker to speaker. Many feature extraction methods have been proposed for LID. These methods provide distinct features robust to noise and are quite successful [41]. Remarkable endeavours have been made to further improve the extraction of speech features [42]. And also on the modelling techniques [43, 44, 45]. The success of all such modelling techniques sprawls on extraction of distinct features from the speech signal. The features that have been used for LID are acoustic [43], prosodic [23], spectral [23] and cepstral [45] features.

2.2.1Text and Speaker Independent corpuses

Text independency means there is no restriction of the words or sentences for speaker. In text independent databases recordings are collected randomly. There is no proper text, words and sentences like the recordings such as talk shows, interviews, broadcast, random conversations and telephone calls etc. some of the corpuses with independent text and speakers are discussed in this section.

In [46] authors use Hidden Markov Model (HMM) to differentiate between two languages (English and French). The same model is also applied on Oregon Graduate Institute (OGI) Telephone Speech [47] corpus for LID. The OGI corpus comprises over ten languages (English, Farsi, French and German etc) [47]. Perceptual Linear Predictive (PLP) makes use of OGI Database to extract features and train neural network for LID [48]. First neural network is trained for ten languages and then for five languages (Japanese, Mandarin, American English, Chinese and Tamil). Recordings from the first 70 valid calls (from 342 males and 158 females) in each language were used for experiments. Recordings from the first 25 valid calls (from 151 males and 41 females) in each language were used in training phase. Average duration of each utterance was 13.4 secs. It is shown that neural networks demonstrate 47.7% accuracy on ten languages and 70% accuracy on four languages.

In [45] LID performance is evaluated on four different datasets OGI [47] which consists of European languages (English, French, German, Spanish) Asian languages (Japanese, Korean, Mandarin Chinese, Farsi, Tamil and Vietnamese languages), CCITT [49] dataset which consists of 20 languages (English, French etc), Rome laboratory database [50] which consists of three languages (Russian, German and Chinese) and Spoken language library (three languages). Duration of the sample was 10 minutes for each utterance from Rome laboratory database and from OGI database average duration of the utterance was 17 secs long. For CCITT database average duration of each utterance was 8 secs long. LPC is pre-owned for extraction of Cepstral features from the speech. HMM is trained on features and transition probabilities of HMM are obtained with Gaussian Mixture Model (GMM). Experimental results show that LID performance is encouraging on OGI corpus as compared to others.

Two applications of the LID system is defined in [12], first system determines which identification model should be loaded and run and the second system routes the incoming speech to a corresponding language operator. For this purpose, HMM is used. Parallel Phone Recognition and Language Modelling (PPRLM) are used for extraction of phonemes as features from the speech samples. Features are extracted from twelve different languages of the OGI corpus (English, Vietnamese and Korean etc). Experiments are performed in two phases: first phase includes 11 languages and second phase includes two languages. LID is implemented on two sets of speech (45 secs and 10 secs). It is shown that LID achieves 89% accuracy on 45secs and 79% accuracy on 10secs evaluated on 11 languages and for two languages LID achieves 98% accuracy on 45secs and 95% accuracy on 10 secs.

In [51] the performance of GMM based approaches are evaluated on Call-friend and OGI corpus. Shifted-Delta Cepstral (SDC) is used for feature extraction. The Call-friend corpus consists of conversations captured over telephone lines in twelve different languages with duration of 30 minutes per language. In case of OGI corpus 11 different languages are used duration about 90 minutes per language. For the testing phase there are two subsets one includes length of the utterance about 45 secs and second includes 10 secs long. It is shown that GMM on OGI gives best performance as compared to Call-friend corpus.

Five languages from IDA (Integrated Deductive Approach) language database (three European and two indo-Asian) was recorded by approximately 82 male speakers, three Asian language from Spoken language library includes 32 different male and female speakers and from OGI corpus 900 speakers includes from 10 different languages are used in [52]. Average length of each utterance was 25 to 50 secs long. Spectral features are extracted by first automatically marking vocalic centres and then encoded by spectral frames. This process trained the system to extract the features from the language without any high level knowledge of that specific language. HMM is trained for LID. It is shown that system achieves 95% identification accuracy.

MFCC, PLP and SDC methods are combined for feature extraction in [53]. Ten different languages of the OGI corpus are used as a speech corpus. GMM is trained for identification of the languages. A successful system is obtained by combining of PLP with SDC with an accuracy of 76%. MFCC and formant frequencies are extracted features from the speech samples. The speech samples belong to OGI corpus [6]. For training, samples of 25secs duration for each language are used. Testing is performed on 1sec, 2secs and 3secs duration speech samples. To remove the noise effect from the speech feature wrapping technique is used during feature extraction process. A new feature vector is created by combining the strength of MFCC and formants. Different number of mixtures (8, 16 and 32) is evaluated with GMM using features. It is shown that performance of LID for 32 mixtures is better than 8 and 16 mixtures. With 32 mixtures system achieves 98% accuracy.

Nearest neighbour based classifier is used for LID in [54]. Linear Prediction Coefficients (LPC) is used for feature extraction. LPC collects features like energy from the speech and are found robust to noise. Corpus consisting of four hours' speech is recorded in Slavic and indo-Asian languages. It is shown that nearest neighbour achieves 64% accuracy. HMM model is also used in [55] to identify five different language (two Asian and three Indo-European languages). Acoustic features like loudness, zero-crossing and power of the samples recorded by ten different speakers are used.

A segment based approach is applied for LID in [56]. Features are extracted from segments and Neural Networks are trained. A corpus consisting of four different languages are used i.e American English, Japanese, Mandarin Chinese and Tamil. Corpus is collected according to the age range of the speakers. Twelve native speakers (6 males and 6 females) for every language record 15 conversational sentences of their own choice. Age range of the female speakers was 15 to 71 years and range of male speakers was 18 to 71 years. All the selected speakers had spent their major part of the life (childhood and youth) in their native countries. Duration of the utterances was 3.7 secs for the training phase and 4.0 secs for the testing phase. Samples are recorded with the help of microphone sampled at 16 kHz. Accuracy rate for identifying Mandarin Chinese, Japanese, Tamil and American English languages are 85%, 83%, 82% and 74% respectively.

HMM is also used for LID in [44]. Sentences of 10 minutes in four different languages (English, Spanish, Hindi and Mandarin) are recorded in a controlled environment. There was no restriction for fixed vocabulary; speakers were free to choose sentences of their own choice from the literature of each language. LPC is used for feature extraction. It is shown that the parameters of the features such as pitch, loudness, volume etc are different for every language. But despite that HMM identifies all languages of small database with least error rate.

MFCC and PLP are combined for feature extraction [3]. GMM is used for language identification. Samples of three South-Asian languages (Bengali, Hindi and Telugu) are recorded by different speakers and different samples of foreign languages (Dutch, Italian, Russian etc) are downloaded from internet sources. Seven different speakers from ten languages were participated in this research and each speaker utterance was about 1-minute long. All the recordings were recorded in controlled environment. Average length of the utterance was 35sec to 70 secs. For each model the identification performance is check on 2 secs, 4 secs and 10 secs duration. It is shows the system achieves on average an accuracy of 88.7%.

Acoustic features are collected from five different languages (English, Spanish, German, Japanese and Mandarin) [14]. Each language has different words length and different words sequence. Features are extracted with the help of LPC model. To utilize these features HMM is trained as a classifier. Experimental results show that system obtains 88% accuracy on recording of 50 secs length and 81% accuracy rate for the duration of 10sec.Similarly an Acoustic Segment Model (ASM) based LID framework is deployed for LID in [57]. This model trained acoustic and language model for three languages (English, Korean and Mandarin). Average duration of the utterance was 30 secs long. Objective is to collect the cues from the speech for human beings to identify the known and unknown languages on the basis of little knowledge of that specific language. It is shown that ASM achieves 86% accuracy.

For language identification task, recordings consisting of five targeted languages (Arabic, Dari, Farsi, Pashto and Urdu) are employed in [17]. Duration of the utterances was 10 secs and 3 secs. MFCC and PLP are used for feature extraction. Three approaches are used for training:

Single Gaussian model (SGM), Linear Discriminative analysis (LDA) and Neural Networks. Experimental results show the improvement in the performance of the system upto 10% on 3 secs and 10 secs utterances. It is shown that performance of nearest neighbour classifier is comparatively but better than SGM and LDA[17].

Audio features are extracted through MFCC in [58]. Bag-of-words approach is used for LID. In their previous work they only considered the two languages (English and Mandarin) but for this task 25000 videos of twenty-five different languages are used which are Nepali, Khmer, Arabic, Pashto, Punjabi, Russian, Sinhala, Spanish, Tagalog and Tamil etc. Main task of this research is to combine the audio-visual features from the songs of each language. Duration of the recording varied for each language but the frame of each feature is fixed. For each language a linear SVM is trained to separate the audios that associated to certain language. The experimental results show that SVM based LID is 48% accurate. GMM based approach is applied in [59]. Recordings of 15 different languages of European and Asian areas (English, Italian, Farsi, Arabic, Polish, German and Romanian etc) are recorded through TV channel. Length of the utterances is 5 secs, 15 secs, 30 secs and 60 secs. Linear Prediction Cepstral Coefficients (LPCC) parameters are extracted from the speech. It is shown that system achieves approximately 38%, 46%, 51% and 57% accuracy on 5 secs, 15 secs, 30 secs and 60 secs utterances respectively. To increase in the utterance length and used of higher level information (pronunciation, vocabulary and accent etc) allows the system to achieve better results.

A new approach to language identification is based on SVM is proposed for recognition of the languages in [60]. SVM is trained for twelve different languages (English, Farsi, Spanish, Mandarin and Japanese etc). Features are extracted with the help of SDC approach. Fusion of three techniques (GMM, PPRLM and SVM) is evaluated for LID. Fusion of three systems shows approximately 2.7%, 7.8% and 20.3% error rate on the utterances of 30 secs, 10 secs and 3 secs respectively. System shows 4.8% error rate using GMM technique, 6.1% error rate using SVM and 6.6% using PPRLM.

Spectral features are identified to get the specific information in speech [61]. MFCC and LPCC are used to extract the spectral features. Corpus consists of 27 different Indian languages (Gojri, Gujrati, Marathi, Rajasthani, Sanskrit and Bhojpuri etc) are used for analyzing specific information in the speech. Recording of the broadcast television channel is used as a database. In case of some languages where recording is not available on the channel radio broadcast is used for collecting the corpus. Recordings of the speech includes one hour for every language with the sampling rate of 16 kHz. About 80% data is utilized for training and remaining for testing purpose. GMMs are trained on the basis of spectral features for LID. Accuracy of the system depends on different parameters such as number of features and dimension of the features. Performance of LPCC is better than the MFCC on the basis of their experimental results.

Acoustic model is trained on 12 languages of the Call-friend corpus [7]. Recordings of 3secs, 10secs and 30secs duration are used. PPRLM is used as classifier. It is shown that the equal error rate for 30secs is 0.8% and for 10secs is 3.0%. SVM and GMM are used with features extracted through Shifted Delta Cepstral (SDC) method in [62]. Speech corpus consists of twelve languages (English, Farsi, German, Mandarin etc) is used. For training, twenty complete conversations (30 minutes) are collected from each language from both male and female. Test data consists of different speech segments of duration 3 secs, 10 secs and 15 secs. They combine both classifiers to obtain better results.

A unified Deep Bottle-neck (DBN) based i-vector framework is implemented for LID in [31]. The main goal of this framework is to performs both tasks feature extraction and modelling. Phonetic and spectral feature are extracted from two languages English and Mandarin. Corpus includes recordings of 1000 hours for Mandarin language and 300 hours of English language. Duration of the utterances was 30 secs, 10 secs and 3 secs. DBN is trained for both languages separately. Framework is shows improvement of 46%, 7% and 13% for 30 secs, 10 secs and 3 secs respectively for Mandarin language and for English language it shows 22% and 11% for 30 secs and 10 secs respectively as compared to previous framework.

HMM and Neural Networks are trained for LID in [63]. PLP, MFCC and Split Temporal Context (STC) are used to extract phonetic features. Wall-street Journal (WSJ) based corpus [64] and Speech-Dat [65] corpus are used for training and testing. These corpuses include four languages (English, Czech, Hungarian and Russian). They are used as phone decoded sequences instead of words for training the machine. Collection of English language recording from WSJ corpus and recordings of non-English languages are taken from Speech-Dat corpus. Idea is to develop a phone recognizer instead of Automatic Speech Recognition (ASR). It is good to have a single phone recognizer which summarizes each language instead of single phone recognizer individually for each language such as English language phone recognizer only summarizes the English language.

Two different dataset (Google 5M and NIST LRE 09) are used in [32]. Google 5M is generated to collects the data from Google speech recognition services. Corpus of Google 5M consist of 34 languages (Turkish, Sweden, Russian, Italian, Bulgarian etc). NIST LRE 09 [66]includes eight different languages (English, Spanish, Dari, French etc). DNN is use for LID. Performance of this technique is checked on the length of the speech sample. According to their experiments result samples with the length of 2 sec are to reach the accuracy rate of 90% as compared to length of 0.5 sec.

SVM classifiers are used for machine learning [25]. The proposed identification system is evaluated on 2003 NIST language database. This database includes 12 different languages such as English, Spanish, French and Mandarin etc. Length of the speech sample was 30secs for each recording. Performance of SVM, PPRLM and GMM is evaluated in this research. To extract the features from the samples SDC is used as a feature extractor. SVM framework obtains an equal error rate of 4.0%. Performance of the GMM and PPRLM obtains an equal error rate of 5.1% and 5.0% respectively in the 30secs task. DNN technique is also use in [67] where PLP is use for feature extraction. Broadcast data is obtained from Voice of America (VOA) includes eight different languages (Spanish, French, Farsi, Pashto, Russian, Urdu, Chinese, Mandarin and English). Data is the mixture of telephone and non-telephone speech. Sample with the length of 3 sec is focused in their research. Comparison and fusion of DNN technique and i-vector is also evaluated. Fusion of both techniques leads to the better improvement of the system.

Efficiency of the LID system in controlled and open environment is computed in [68]. Indian LID system which includes 27 different languages (Rajasthani, Sanskrit, Telugu, Dogri and Konkani etc) was analyzed. Recordings of the sixteen languages were recorded through news channels, talk shows and interviews. For the development of the LID system fifty minutes' recordings were recorded and for testing or evaluating the system sixty utterances were used. Duration of the sixty utterances was 5 sec of each language. MFCC feature were used to extract the features. GMM was designed to develop the system for identification. System shows that comparison of open and controlled environment, controlled environment shows 63% more accuracy rate than open environment.

PLP based features are extracted from the speech data in two different languages (English and Japanese) from a telephone channel [69]. Features are used to train a neural network for LID. For this purpose, from English language 50 recordings are selected for training and 20 are selected for testing phase and for Japanese language 35 recordings are selected for training and 10 recordings are selected for testing phase. Duration of the recordings varies from 1sec to 49secs with an average of 13 secs. Results show that neural network demonstrates an accuracy of 70%. Support Vector Machine (SVM) is classifier used for pattern classification.

Three different features are extracted in [21] namely MFCC-SDC, MFCC and Gamma tone frequency cepstral coefficients (GFCC). Speech sample consists of DARPA RATS database (languages are Bengali, Korean and Urdu, Arabic, Farsi, Dari etc). Speech samples are recorded in noisy and uncontrolled environment. I-vector modelling approach is used for efficient language identification. Experimental results show that fusion of the three feature extractor technique gives better result than individual

2.2.2 Text and Speaker dependent corpus

In text and speaker dependent database fixed vocabulary sentences, words or sentences are recorded. In this type specific fixed number of speakers is there for recordings. Samples are recorded by speakers in different environment such as controlled environment, lab, class room and open environment. Some of the corpuses with text and speaker dependent corpus for language identification are discussed in this section. HMM is developed for LID system in [11]. System is trained for fixed vocabulary of 102 words of Hindi language from twelve different speakers of age between 18 and 23. Twelve speakers include seven males and five females used for training. Speech is recorded at the sampling rate of 16 KHz. For testing a dataset of five different speakers is recorded separately. To evaluate the system performance test data was recorded in a controlled environment, class room environment, lab and open space. Their system consisted of three phases: (i) acoustic analysis (ii) training phase and (iii) testing phase. Their system was developed in Ubuntu 10.04 operating system environment it is a platform of Linux. Mel-Frequency Cepstrum Coefficients (MFCC) is used for feature extraction. It is shown that the system achieves an accuracy of 87%.

A novel technique is used in [70] for SVM based identification system. Effect of sequence is also discussed in which training and testing is performed for LID. It is speaker dependent and text dependent (fixed digits) research. Fixed digits are spoken by different speakers. There are two techniques used to extract the features which are LPC and MFCC. SVM is used as classifier to compute the accuracy rate. Identification accuracy is 97% achieved by SVM classifier with LPC feature extractor. And for MFCC features system achieves 98% accuracy.

Extraction and selection of the features is an important task in identification system, it directly affects the performance of the system. Therefore MFCC and PLP are used for feature extraction in [34], GMM is trained for acoustic modelling and HMM for sequence modelling. System is tested in a room and also in open environment. Eight distinct speakers are asked to speak 115 words and six speakers used for testing phase in controlled environment. Three samples of each speaker were recorded. For the developing of the system Linux operating system Ubuntu 11.0 was used. Six speakers were asked to record 35-40 words for the testing phase. Average performance of the system is 94% to 96% and error rate is 4% to 6%.

Automatic syllable repetition in the speech is identified in [71] . Syllable repetition is important parameter of stuttered speech. Stuttered is a speech disorder known as stammering in United Kingdom. Dataset is consisted of 150 words of American English; its mean it was text dependent. Fifteen speakers (both male and female) were there for recordings around the age range of 25 to 30 years. Features are extracted through MFCC at sampling rate of 16 KHz with the 16 number of bits. 80% recorded data was used for training the system and remaining 20% for evaluating the machine. Accuracy achieved by the system around 93%. It was better than their previous work in which HMM used as a classifier and accuracy rate was 78%.

2.2.3Text dependent and Speaker Independent corpus

In this section databases with dependent text and independent speaker are discussed. In this kind of corpuses text is restricted to some specific words, sentences or a paragraph (dependent), on the other hand there is no restriction for the speakers. Any speaker recording is selected as a vocabulary for corpus. Recordings are recorded in any type of environment. Features are obtained from speech using PLP in [19]. Corpus is taken from M2VTS database [72], which consists of 185 recording of 37 different speakers with fixed vocabulary. Five recordings are collected from every speaker. Sampling rate for audio stream was 8 kHz and sampling of the frame was 10 ms and 30ms. RASTA algorithm is used to remove the noise from the speech sample. HMM model is trained for LID. Experimental results show 2.8% error rate.

2.2.4Text independent and speaker dependent corpus

This type of database has dependency in speaker but there is no restriction in speech vocabulary. Specific speakers are selected to speak the sentences or utterances according to their choice. Same speakers are used for both purposes training of the machine and testing the ability of system. For more understanding some of the corpuses with such characteristics are discussed with their systematic tools (feature extractors and classifiers) and their other attributes.

In [73] a class dependent technique for text independent speaker identification on very short utterances is used. In this paper a method for spotting speakers in independent text is introduced. This method is based on GMM maximum likelihood for speaker identity. Mixture modelling is used with GMM for identifying the language from any utterance. It is a text independent model with 12 restricted speakers. About 89% accuracy is measured for an average of 10 second test speech length. This model is also estimated for a telephone speech database consists of 20 speakers[73]. A comparison of two approaches for text-independent speaker verification task using GMM is presented [74]. Bayesian adaptation is used with GMM-UBM for language identification models. It provides a framework to save calculations during recognition. Results are obtained from the experiment performed on the NIST Speaker Recognition Evaluation corpus. Results shows that this method gives better results from the results taken when the speaker is dependent. In both scenario text will remain independent[74].

2.3 Summary of Corpuses and Applications

Many corpuses are used for language identification process. These corpuses are used for different applications and have different attributes. Some commonly used corpuses are discussed below in Table 2.1. In table some corpuses and techniques are shown which are used for extracting language information. It also discusses the speaker and text dependency of dataset.

Table 2.1: Summary of state of the art methods for LID

Database	**Languages**	**Features**	**Text / Speaker Dependent**	**Classifiers**	**Accuracy**
Two- languages Corpus [54]	Slavic and indo-Asian language	LPC	Speaker = Yes Text = No	Neural Network	64%
Four languages Corpus [56]	English, Japanese, Chinese and Tamil	MFCC	Speaker = Yes Text = No	Neural Network	85%
OGI-corpus [48]	English, Japanese, Spanish and German etc.	PLP	Speaker = Yes Text = No	Neural Network	70%
Telephone Speech [69]	English, German, Japanese and Chinese etc.	PLP	Speaker = No Text = No	Neural Network	70%
OGI- corpus [12]	English, Vietnamese and Korean	MFCC	Speaker = Yes Text = No	HMM and PPRLM	79%
OGI- Corpus [14]	English, German, Spanish, Japanese and Mandarin	LPC	Speaker = Yes Text = No	HMM	80%
Twelve-languages Corpus [60]	English, Farsi, Mandarin and Japanese	SDC	Speaker = No Text = No	GMM, PPRLM and SVM	GMM + SDC = 81% SVM + SDC = 73% PPRLM + SDC = 75%
Call-friend corpus [7]	English, Mandarin, Spanish, Japanese and German etc	MFCC	Speaker = No Text = No	PPRLM	70%
Fifteen-languages Of European and Asian [59]	English, Farsi, Arabic, Italian, Romanian and Polish etc.	LPCC	Speaker = No Text = No	GMM	60%

Indian Languages Corpus [3]	Bengali, Hindi and Telugu	MFCC and PLP	Speaker = Yes Text = No	GMM	88.7%
Five- languages [17]	English, Spanish etc.	MFCC and PLP	Speaker = No Text =No	SGM and LDA	51%
OGI-corpus [6]	English, Mandarin, Spanish, Japanese and German etc.	MFCC	Speaker = No Text = No	GMM	98%
Broadcast data [67]	Spanish, Pashto, French, Urdu, Chinese and Russian	PLP	Speaker = No Text = No	Neural network and i-vector	80%
Google 5M and NIST LRE 09 [32]	Turkish, Sweden, Russian, Italian, Spanish and French	MFCC	Speaker = No Text = No	DNN	90%

Accuracy percentage is also given in this table for comparison with previous and latest work on language identification. Different corpuses are discussed; the most commonly used corpus is OGI due to its wide range of sample of languages. OGI corpus consists of 11 different languages. Samples of OGI corpus are independent in both cases text and speaker. Another corpus is NIST which consists of more than 10 languages.

Speaker and text dependent and independent models are used in different corpuses. Speaker dependent corpuses are in which speaker is restricted. Some bounded speakers are used for obtaining recordings and making a dataset. While in speaker independent corpuses, speakers are not restricted. Recordings can be taken from different speakers. In the same way text dependent datasets are in which text is restricted and every speaker has to utter the same text.

While in text independent there is no limitation of text. Any utterance can be taken as input. OGI corpus is speaker dependent while it is text independent [48]. In a same way call friend corpus discussed in [7] is speaker and text independent. One more corpus Google 5M and NIST LRE 09 in[32] is both speaker and text independent.

In [36] an approach is discussed to identify the people of different regions based on their languages. Different people from different backgrounds communicate to each other through languages. The languages vary from region to region, it creates a language barrier or an obstacle for speaker to communicate with each other. As a solution to this problem many techniques are presented. [36] Discusses a regional language (Sindhi) and national language (Urdu) of Pakistan. This paper uses an audio feature extraction approach and vector quantization. Support vector machine (SVM) is used as a classifier.

2.4 Summary

Different techniques and frameworks are discussed that are used in LID field in the 2ndchapter. Chapter 2 includes numbers of languages, collection of data, format of data, saving method of data, duration of the speech samples and collecting informative samples. Feature extraction and modelling techniques upto date used in LID are also discussed and explored. At the end a latest work is discussed with its extraction techniques and modelling methods based on the future work of previous papers.

CHAPTER 3

SPEECH FEATURES AND MACHINE LEARNING TECHNIQUES

3.1 Overview

In this chapter different feature extraction algorithms are discussed. Some of the machines learning techniques are also described. These features and techniques are briefly explained and elaborated in this chapter. Various steps involved in each feature extraction algorithm and machine learning techniques are explained.

3.2 Speech Features

Feature extraction is the first processing step in LID after collecting the database. Feature extraction plays an important role in LID and collects the informative data from speech sample it becomes the crucial part of the research for many years and performance of the system is heavily depends on extraction of features. Feature extractor task to discard the raw and irrelevant information from the speech and saved the useful information. One of the basic and main steps is the extraction of features from the raw data. To collect information from the speech is a crucial step for LID. Efforts have been made to extract useful features from the speech signals. Feature extraction phase helps in differentiate among one speech to another. It transformed the raw form of speech signal into processed speech signal which is more informative, stable and reliable than the original one [9].

Audio is initially processed to find the speech (words or sentences without interruptions) in an audio recording and then extract the features that show the information about language or speaker. This information (extracted features) is used in data training phase. Data is trained on some specific extracted feature to identify the language from given speech. Features are divided into training and testing data set. Trained data set is then tested on different features, if the achieved result is useful and according to desire it means system is trained well. There are different feature extraction algorithms that have been used. Performance of LID depends on extraction and selection of the features. Some of the feature extraction algorithms described in the following section. The most popular one is Mel-Frequency Cepstral Coefficient (MFCC) which shows better performance in the task of speech processing. Apart from MFCC many other features such as Linear Predictive Coefficient (LPC), Linear Predictive Cepstral Coefficients (LPCC), Linear Frequency Cepstral Coefficient (LFCC) and Perceptual Linear Predictive (PLP) are used as well.

3.2.1 Mel-frequency Cepstral Coefficient

Mel-frequency Cepstral Coefficient (MFCC) is widely used as spectral feature for LID. These features are motivated by human auditory system. It is the short form of sound spectrum. There are few reasons to choose MFCC for LID [75, 76,77] that is first of all it is based on the perception of human hearing, secondly less complexity and fast computation. And lastly it gives high accuracy. Frequency of sound perceive by human does not follow the linear scale [78]. But it follows Mel-frequency scale. Mel scale is calculated with the formula given in Equation 3.1

$$mel\ frequency = 2595 \times log_{10}\left(1 + \frac{f}{700}\right) \tag{3.1}$$

where f is perceptual frequency expressed in Hertz (Hz).This is incorporated in MFCC algorithm. It is designed to capture short-term features. Success of MFCC is the ability to present the spectrum in a compact form. Computation and perception is considered at every step in the process of MFCC. MFCC features are calculated in different steps as explained in Figure 3.1.

Speech samples are typically varying from each other in length, so easy way is to make a feature with fixed size regardless of its length. First step is division of speech signals into fixed size frames by applying a windowing function (Hamming window). Purpose of hamming window is to removes the edge effects, avoid interruption in the speech. Length of the frames is about 25ms. Signals are projected into frequency domain with the help DFT. The phase information is discarded and the amplitude of the spectrum is retained because it carries much important features than phase. After that there is a step of smoothing the spectrum into meaningful frequencies. Mel frequency scale is followed with the bin spacing concept which takes only lower frequencies. Figure 3.1 discusses framework of Mel-frequency cepstral coefficients (MFCC).

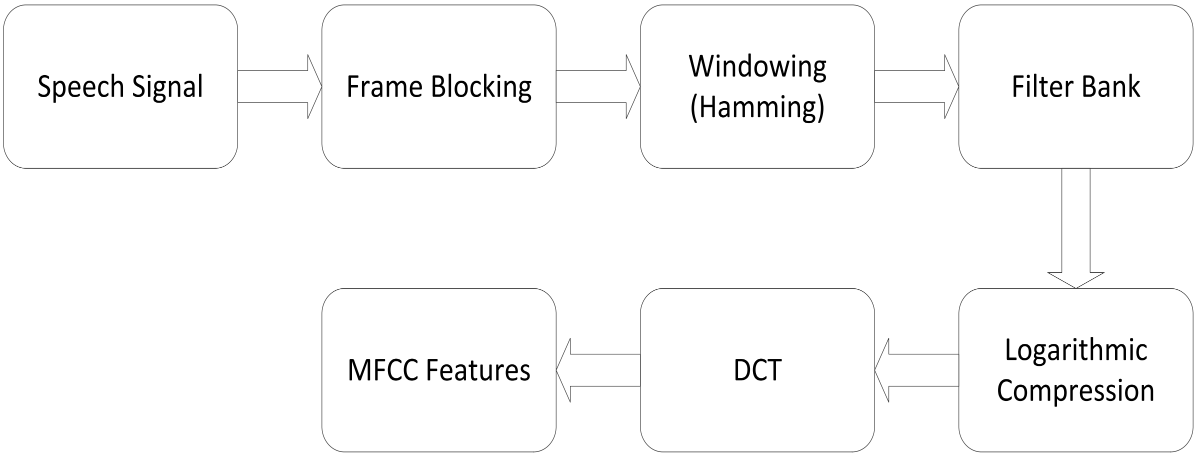


Figure 3.1: Framework of Mel-frequency Cepstral Coefficient

An audio signal is changing constantly but on short time scales the audio signals slightly changed, does not change much. This is the reason to frame the signals into the range of 20-40 ms. Typically; MFCC features are computed for the short frame of speech. If the length of the frame is shorter than we don't get the reliable spectral, signals change constantly if the frame length is high. Presence of the frequencies in the frame is used to calculate the power spectrum of the frame. To check the different variations of the frequencies on different phase filter-bank is used, which is narrow and indicates the variation of frequency on different regions. Filter-bank is used to calculate and compute the average energy. Strength of the filter-bank is scaled by the Mel scale. Different filter-bank energies make a logarithm. Logarithmic form of filter-bank variations is motivated by the human hearing. On a linear scale human don't hear loudness.

The last step is to calculate and compute the Discrete Cosine Transform (DCT) of the filter-bank variations. MFCC's are defined as DCT of the logarithms of the energies of the filter-bank. To compute the DCT there are two basic reasons first is filter-banks are overlapping and the energies of filter-bank are associated with each other. The DCT décor-relates the filter-bank energies to maintain the performance of the LID process.

3.2.2Linear Prediction Coding Coefficient

Linear Prediction Coding Coefficients (LPCC) idea is directly derived from Linear Prediction coding (LPC) [79]. They are more reliable and efficient than LPC. LPCC inherits the characteristics and advantages of LPC. Most commonly used parameters of speech signals are pitch period, speech frame, frame energy and formant, to calculate and estimate these parameters. Linear Predictive Coding Coefficients framework is diagrammatically explained in Figure 3.2.

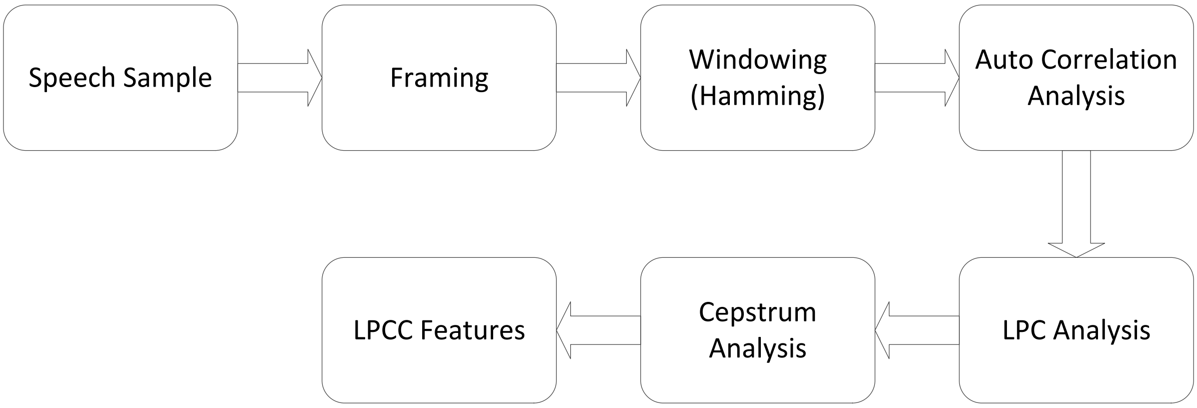


Figure 3.2: Framework of Linear Predictive Coding Coefficient

LPCC has becomes the most important and reliable features. It is a less computationally expensive feature extraction logarithm because it is computed without Fourier transformation to covert the signals from time domain into frequency domain like MFCC. LPC coefficients are transformed into LPCC which are robust to noise and distinct. The method to obtain LPCC is called auto correlation. LPCC features are calculated through Cepstrum coefficients in the LPC parameters. The easy way to calculate Cepstrum coefficients is to find the predicator coefficient vector. Recursion of the LPC parameters helps to convert the parameters into Cepstrum coefficients. N is presented as sample index and k is the time shift. LPC coefficients are calculated using Equation 3.2.

$$E^{(0)} = R(0) \tag{3.2}$$

where R is presented as known values and E is a variable. The value of $x$is computed in Equation 3.3 as:

$$K_x = R(x) - \sum_{y=1}^{x-1} \propto_y^{(x-1)} R(x-y) \propto_x^{(x)} \frac{K_x}{E^{(x-1)}} \tag{3.3}$$

where x is an input speech signal, and$(x-1)$ is previous speech signal. In the following Equation 3.4 the value of$y = 1 : x = 1$

$$\propto_y^{(x)} = \propto_y^{(x-1)} - K_x \propto_{(x-y)}^{(x-1)} \tag{3.4}$$

Here K_x are referred to as reflection coefficients. Now LPC coefficients will be written as in Equation 3.5

$$E^{(x)} = (1 - K_x^2) E^{(x-1)} \tag{3.5}$$

LPC coefficients are explained as $a_x = \propto_x^{(p)}$, where $1 \leq x \leq P$. To obtain the LPCC coefficients after obtaining the LPC coefficients the Equation 3.6 is computed as follow:

$$C_m = a_m + \sum_{k=1}^{m-1} \left(\frac{k}{m}\right) C_k a_{m-k} \qquad 1 \leq m \leq p \tag{3.6}$$

Here a_m & C_m are the nth-order linear predication coefficients and cepstrum coefficients respectively, and p is the prediction order in the above Equation 3.6.

3.2.3 Linear Predictive Coding

One of the most powerful feature extraction techniques is Linear Predictive Coding (LPC). Idea of LPC is based on that a speech signal can resemble as a linear combination of past speech samples. LPC is used in most of the recognition processes especially in language and speaker identification. It is used in audio and speech processing to represent the compressed form of audio signals. LPC analysis any signal from the speech LPC, and removes the interruptions and noise distortions from the sound format. This process is called filtering. It enhances the sound quality of sound and helps to improve system accuracy. LPC has various applications, some of which are voice compression that is used by phone companies; it is also used to encrypt data so that data remains secure and protected. But main application is speech analysis used for the identification of a language or a speaker. A diagrammatical representation of LPC extractor is expressed in Figure 3.3.

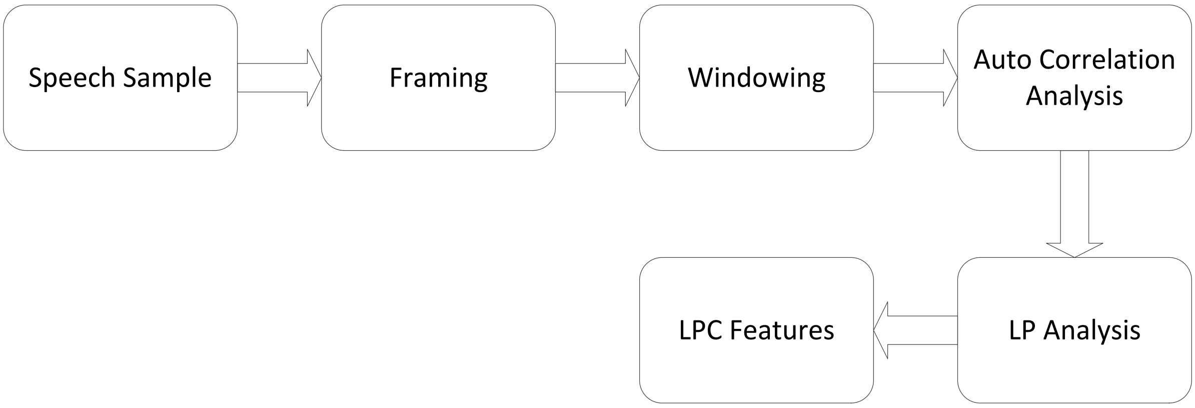


Figure 3.3: Framework of Linear Predictive Coding

Let the discrete time representation of signals $x(t)$ be x_t. The linear equation between input samples x_t and the preceding p samples is as derived in Equation 3.7 as follows.

$$x_t + a_1 x_{t-1} + \ldots\ldots + a_p x_{t-p} = \varepsilon_t \tag{3.7}$$

Here, ε_t is an uncorrelated statistical variable [80]. After that each sample is divided into frames and then each frame is multiplied with hamming window [81]. To remove the leakage effect and smooth of edge the Equation 3.8 is used as:

$$w(n) = 0.54 - 0.46 \cos\left(\frac{2\pi n}{N} - 1\right) \tag{3.8}$$

where N is presented as sample index. Then Cepstrum coefficients are computed. To obtain the coefficients Equation 3.9 is used.

$$r(m) = \sum_{m=0}^{N-1-m} x(n)x\ (n+m) \tag{3.9}$$

where m=12 and N=330. After that LPC values are obtained using the Equation 3.9 Equation

3.10 is derived using the values of Equation 3.9.

For $f = 0$ $\quad f = 0 \quad E(f) = r(f)$

For $f = 1$ $\quad f = 1 \quad k(f) = \frac{r(f)}{E(f-1)}$

$$E(f) = E(f-1)\{ 1 - k(f)2\} \tag{3.10}$$

$$a_f^f = k(f)$$

$$k(f) = \frac{1}{E(f-1)}\big(r(f)\big) \tag{3.11}$$

$$a_m^f = a_m^{f-1} - k(f)a_{f-m}^{f-1}$$

$$a_f^f = k(f)$$

3.2.4 Shifted Delta Cepstral

Shifted Data Coefficients (SDC) is extension of delta coefficients. SDC features are widely used to present the better and useful information in the speech signals for LID. SDC features are particularly acceptable for language identification because they collect wide range of features over a wide range of time. One of the most commonly used acoustic features in LID is Shifted Delta Cepstral [82] appraised an extension of the MFCC and PLP [83]. It has become the most popular feature extractor in the acoustic model approach. SDC features capture additional information in speech and improves the system performance [84] . Basic set of MFCC coefficients is explained in Equation 3.12 as follow

$$\{ C_j(t), j = 0,1, \dots \dots, N-1\} \tag{3.12}$$

Here j is dimension index, t is the frame and N is the number of coefficient. Features are expressed in Equation 3.13 as follow

$$(t)_{S(iN+j)} = C_j(t + iP + d) - C_j(t + iP - d), \ \ i = 0,1, \dots \dots \dots, k-1 \tag{3.13}$$

where *c(t, i)* represents the *ith* block of features, time is represented by t. SDC features are identified by four parameter, $N - d - P - k$. N is the number of coefficients calculated at each frame, d represents the delay in time, k is the number of blocks and p represents the shift in time within the consecutive blocks. Usually the values of $N - d - P - k$are 7-1-3-7.

3.3 Speech Modelling Methods

After feature extraction features are trained using some classifiers. Collection of information from the speech dataset is the main step in language identification. Useful features are extracted from the speech signals and trained on different modelling methods. Training and testing extracted speech signals helps in differentiating the language from one another. After audio is processed from audio recording extracted features are trained using different classifiers. Classifiers performance is directly proportional to the accuracy rate of the language identification system. Some classifiers (modelling method) are discussed in this section. Most commonly used for language identification are GMM, GMM-UBM, SVM and i-vector. These classifiers give better results as compare to other modelling schemes of techniques. Different features are paired with different classifiers for improving results. Apart from these some other classifiers are also used for language identification purpose. Those are discussed in literature review section. Basically used modelling schemes (GMM, SVM, I-vector) are discussed in this section.

3.3.1 Gaussian Mixture Model

In pattern recognition GMM classifier has gained the attention. In this model a function of density probability is expressed as linear combination of functions. Performance of GMM based system is good but there is a problem that affects the use of GMM in real time application. GMM models require large amount of memory and complex computation in exponential calculations. GMM is widely used in different applications such as acoustic modelling for speech identification, speaker identification and verification [85]. Framework of GMM is diagrammatically explained in Figure 3.4.

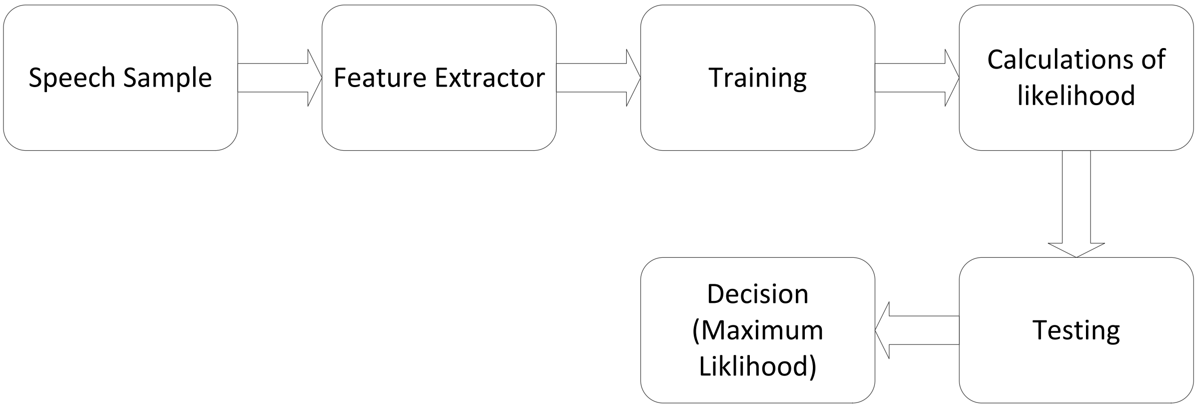


Figure 3.4: Framework of GMM

GMM is works as front end and back end modelling. In the front end feature extraction vectors are used. Back end modelling includes training and testing of the machine. Likelihood is calculated and selected in the modelling stage as explained in Equation 3.14.

$$\rho(x|\leftthreetimes) = \sum_{k=1}^{v} \omega_k g\ (x|\mu_k,\Sigma_k) \tag{3.14}$$

where Gaussian components are represented by v and prior probability is represented with $x\epsilon R^i, \omega_k > 0\ , (k = 1, \dots \dots v)$. Component Gaussian densities are $g\ (x|\mu_k,\Sigma_k)$ where (k= 1,. … . .v). Gaussian densities are measured with the formula used in Equation 3.15.

$$g(x|\mu_k,\Sigma_k) = (2\pi)^{-d/2}|\Sigma_k|^{-1/2} \exp\{-\tfrac{1}{2}(x-\mu_k)^t\,\Sigma_k^{-1}(x-\mu_k)\} \tag{3.15}$$

where μ_k presented as mean vector and Σ_k presented as covariance matrix. Mixture weights are presented as in Equation 3.16 and their densities of all these three components are represented as in Equation 3.17.

$$\sum_{k=1}^{v} \omega_k = 1 \tag{3.16}$$

$$\leftthreetimes = \{\omega_k,\mu_k,\Sigma_k\}_{k=1}^{v} \tag{3.17}$$

Maximum likelihood is available for estimation of the parameter $\leftthreetimes$ from the training dataset $X = \{x_1, \dots \dots, x_v\}$.

3.3.2 I-Vector

I-vector has become the state-of-art in the text independent language recognition. I-vector is the popular technique for verification and recognition due to its brilliant performance. Over the last few years i-vector representation has gained remarkable interest by researchers in both language identification and speaker verification due to the ability of achieving better performance. Classical i-vector approach is based on GMM. I-vector framework covers the front-end feature extraction and back-end modelling stages.

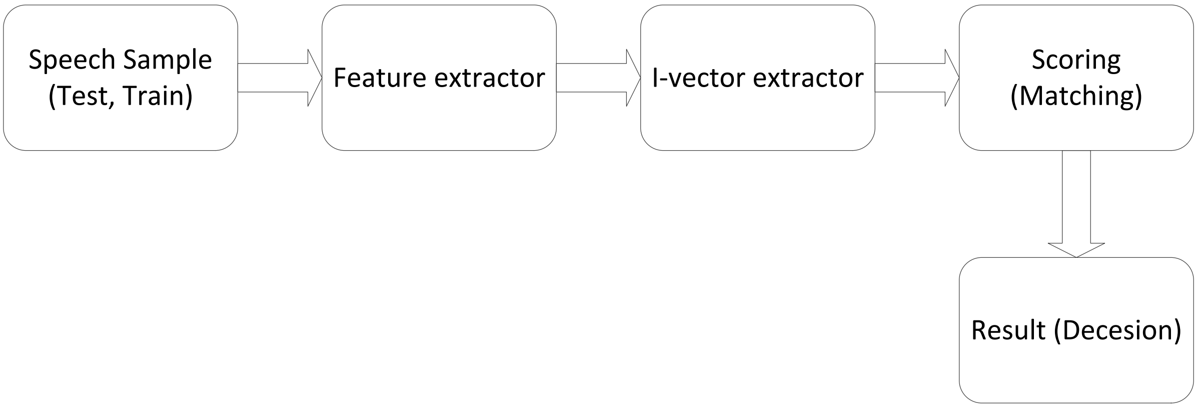


Figure 3.5: Framework of I-vector

A known and unknown speech sample is collects for identification. Features are extracted from the speech samples. Classifier is worked on that features to rain the machine for decision. Let $Z = \{z_1, \dots \dots .., z_F\}$ is an utterance with F frames of speech, $z_f \in R^d$is the extraction of features on the $f-th$ frame. Let $\leftthreetimes = \{\leftthreetimes_1, \dots \dots \leftthreetimes_T\}$ is the function of probabilistic density that analyze the process of z_f where $\leftthreetimes_T$ are the parameters of the $t-th$ component. N_t and E_t are calculated for zero-order and first order for $t-th$ component are as follows in Equation 3.18, Equation 3.19 and Equation 3.20 respectively.

$$\Upsilon_{f,t} = P(\leftthreetimes_t | z_f) \tag{3.18}$$

$$N_t = \sum_{f=1}^{F} \Upsilon_{f,t} \tag{3.19}$$

$$E_t = \sum_{f=1}^{F} \Upsilon_{f,t} \, z_f \tag{3.20}$$

$\Upsilon_{f,t}$ is presented as a posterior probability for $z_f$on the$t-th$ component.Ê is used for super vector that is achieved by concatenating all E_t together. After the implementation of the factor analysis on the Ê super-vector total variability space is calculated as follows in Equation 3.21

$$\hat{E} = ms + R_d \tag{3.22}$$

where d is presented as a q-dimensional i-vector with$N(0, I), q \ll d$ (N is the normal distribution). $\Upsilon_{f,t}$ is the Gaussian posterior probability and ms is the mean super vector [31].

3.3.3 Support Vector Machine

SVM is powerful and strong state-of-art modeling technique for LID which implements a discriminative approach. SVM is used linear and non-linear technique for classification of data. SVM can only classify the fixed length data. SVM is particularly used for binary classification. By controlling the VC dimensions SVM controls the complexity of the model. SVM classifiers separate the regions of two classes in an organized way through a non-linear decision boundary. There are main two ways of classification first is linear and the other is Non-linear classification.

Figure 3.6 illustrates the steps of SVM for classifying the audio speech signals.

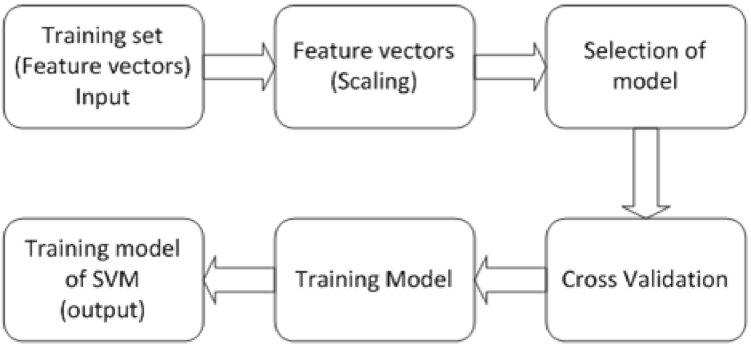

Training set (Feature vectors) Input
Feature vectors (Scaling)
Selection of model
Cross Validation
Training Model
Training model of SVM (output)

Figure 3.6: Framework for SVM

3.3.3.1 Linear SVM classification

Algorithm for linear classification on SVM was proposed in. In this algorithm maximum margin hyper plane is calculated from given training database D as described in Equation 3.22.

$$D = \{(a_i, b_i)|a_i \in R^p, b_i \in \{-1,1\}\}_{i=1}^{n} \tag{3.22}$$

where value of b_i is either -1 or 1 and n is the number of training samples. Each $a_i$is a p-*d*imensional vector with feature quantity *R*. Equation 3.23 gives the product of hyper plane and vector as:

$$w.a - x = 0 \tag{3.23}$$

where w is a hyper plane vector. If the training data values are linearly separable, the value of hyper plane can be described as in Equation 3.24.

$$w.a - x = 1 \; and \; w.a - x = -1 \tag{3.24}$$

The distance between hyper plane is describe as $2/||w||$, the purpose of this is to minimize $||w||$. Therefore, the algorithm can be expressed as in Equation 3.25. And Formula is changeable even without changing the solution as shown in Equation 3.26.

Minimize $/||w||$ under the specific condition of $b_i(w.a_i - b) \geq 1$,for any $1 \leq i \leq n$ (3.25)

$Min_{w,x}\frac{1}{2}\ /||w||^{\ 2}$ under the specific condition of $b_i(w.a_i - b) \geq 1$,for any $1 \leq i \leq n$ (3.26)

In linear SVM set of hyper plane or the single hyper plane can be expressed as separate line in classification. Margin of the separation between classes is directly proportional to the performance of linear SVM.

3.3.3.2 Non-linear SVM

Non-linear SVM classifiers can be expressed by using kernel trick. Kernel function is expressed as in the following Equation 3.27.

$$\text{Polynomial: } k(y_i, y_j) = (y_i, y_j + 1)^d \tag{3.27}$$

where value of i is varies from 1 to n. Accuracy of the classifier is directly depends on kernel, parameters and cost factor.

3.3.4 K-Nearest Neighbour

Idea of K-nearest neighbour (KNN) prediction of query instance is simply based on majority of classes of nearest neighbour. The basic principle is based on minimum distance from the unidentified samples to the training samples to calculate the k-nearest neighbour. Euclidean distance is one of the mostly used distance measures. Calculations of distance from unknown pattern in testing dataset are used. Samples are used in training dataset with known class for calculations. It can be expressed in Equation 3.28.

$$d_{ij}^2(a) = \sum_{b-1}^{B} \{x_i(a,b) - x_j(a,b)\}^{\ 2} \tag{3.28}$$

where value of $a = 1,2,3, \ldots .A$, and $x_i(a,b)$ and $x_j(a,b)$are represented as training dataset and testing dataset $a-th$ sample and $b-th$ dimensions of features. Size of sample is represented as A and dimensions of the samples are represented as B . the algorithm about working of K-nearest neighbour is summarized in flow chart as in Figure 3.7.

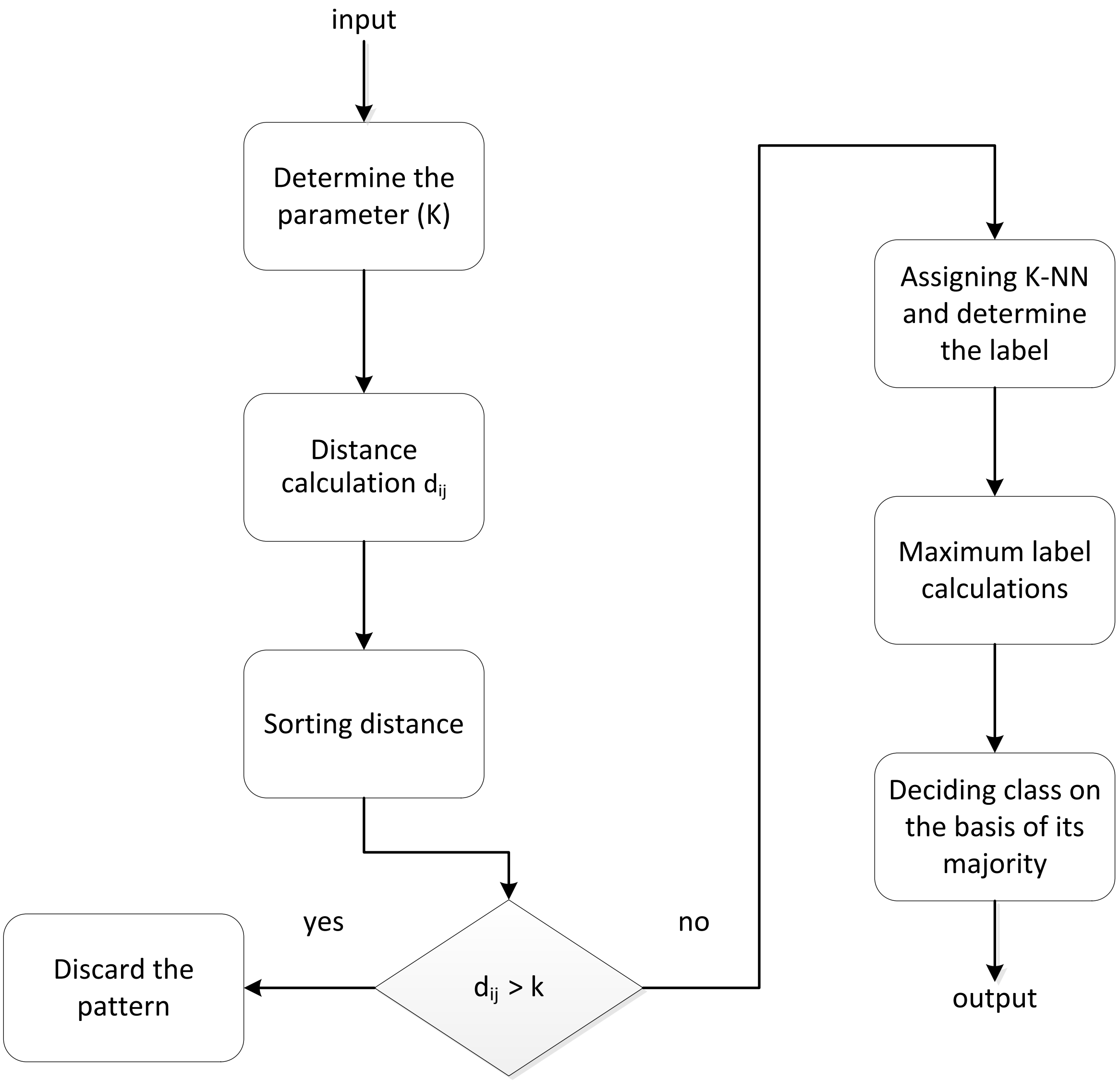

input
Determine the parameter (K)
Distance calculation d_{ij}
Sorting distance
yes
no
Discard the pattern
$d_{ij} > k$
Assigning K-NN and determine the label
Maximum label calculations
Deciding class on the basis of its majority
output

Figure 3.7 Framework of KNN

3.4 Summary

This chapter includes the different feature extractor and classifiers techniques for language identification. Ability and performance of the classifiers and feature extractors are discussed in this chapter. Mathematically explanation of different feature extractor and classifiers is described in this section. Process of techniques is described diagrammatically. Starting of the paragraph consist of features and abilities of the feature extractor and classifiers. Then diagram, and mathematically explained. After discussing the techniques, computational cost, reliability MFCC is the most reliable and efficient extractor used by different researchers. In the classifier GMM get more attention of the researchers than others.

CHAPTER 4

MEHTODOLOGY

4.1 Overview

This chapter includes the methodology of our research. Framework of our system is defined in the following paragraphs. Selection of corpus, collection of corpus samples, and extraction of features from the corpus are then analysis on the specific classifier are the part of the proposed system. Experiments are performed on the corpus samples and all the results are generated on the basis of experimental procedure. Language identification process completely depends on collection of corpus; to get the better result collection of corpus must be efficient and accurate.

4.2 Proposed system

It consists of speech corpus. The speech corpus is further divided into training and test samples. Trained set has different languages category that are trained on the basis of their speech features. All the trained speech features are combined in GMM-UBM classifier for computation, after classification results are shown based on the performance criteria (Accuracy/ Equal Error Rate). In a same way these trained speech features are passed through proposed method Nearest Neighbour Feature Matching (NNFM) that will give NN train speech features. These NN features are trained on the classifier and results will be stored in the form of performance criteria. On the other hand, speech features are extracted from test samples and trained on proposed methodology that is Nearest Neighbour Feature Matching (NNFM) strategy. NNFM and performance evaluation metrics will give NN test speech features which are classified and results will be shown is measured on the basis of accuracy and error rate of system. All this process is briefly diagrammatically explained in Figure 4.1 shown below:

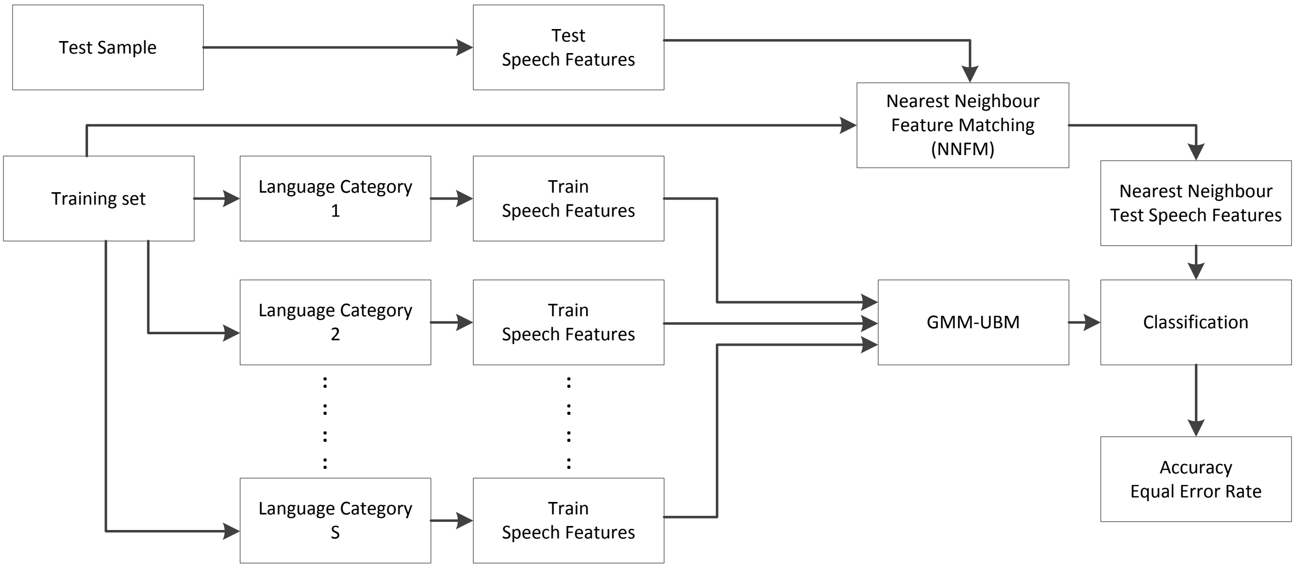


Figure 4.1: Block diagram for proposed Nearest Neighbour Feature Matching method

4.3 Corpus of Regional Languages

Speech corpus plays vital role for LID. The accuracy of LID is directly connected with the quality of speech corpus. Useful and informative corpus gives better result than the useless corpus that contains huge amount of data. In this thesis, a corpus comprises of five different languages of Pakistan is constructed. The languages are Urdu, Punjabi, Pashto, Sindhi and Balochi. There are 150 speakers per language category. The audio data i.e, speech samples are collected from different internet sources. The corpus is referred to as Corpus of Regional Languages (CRL).

Each language category contains text independent speech samples recorded by different male and female speakers. Each sample is 15 secs long. The text in each sample is different from all other samples of the corpus. Each speaker is also different from other speakers of the same Language category. The reason for this is an implementation of LID in text and speaker independent way. Features are extracted from the recordings with the help of different feature extractor like MFCC and Linear Predictive Cepstral Coefficients (LPCC). The features are used to train different classifiers like SVM and GMM. All speech samples are in .wav format. The sampling rate is 16 kHz. Duration of each recording is 15 secs. Selection of the sentences and phrases are not fixed.

There are 150 different speakers (both male and female) per language category group. Total number of speakers are 5 (languages) x 150 = 750. The corpus is, therefore, speaker independent. It is also text independent because each speaker speaks totally different text from all other speakers. Table 4.1 summarizes the speech corpus. The corpus is referred to as Corpus of Regional Languages (CRL). The corpus is randomly divided into two disjoint sets: (i) Training set and (ii) Test set.

Table 4.1: Summary of Corpus of Regional Languages (CRL)

Language	**Training Samples**	**Duration**
Punjabi	150	15secs
Sindhi	150	15secs
Pashto	150	15secs
Balochi	150	15secs
Urdu	150	15secs

4.3.1 Training and Testing set

The training set consists of 100 samples per category. These samples are randomly selected from each category. Total number of samples in the training set is 100 (samples) x 5 (languages) = 500. Whereas the test set consists of remaining 50 samples per category. Total number of samples in the test set is 50 (samples) x 5 (languages) = 250.

4.4 Speech features

MFCC, LPCC and SDC are feature algorithms applied on the speech samples of the training and test sets for features extraction. If MFCC is used as feature for training then MFCC is also used as feature for testing purpose, Similarly, for other features if LPCC features is used for training the dataset, it is also used in testing that dataset. And the same rules applied to SDC.

4.5 Gaussian Mixture Model-Universal Background Model

In the proposed method GMM-UBM classifier is used. In-fact the performance of GMM-UBM is enhanced with proposed Nearest Neighbour Feature Matching (NNFM) strategy in the proposed method. The speech features of the training set are provided to GMM-UBM. The GMM-UBM computes different Gaussian mixture components from the features and then uses the Gaussian components to extracts two language models, which are (i) General Language model (ii) Specific Language model

4.6 General Language Model

To compute General Language model (L_g), GMM-UBM combines the speech features of all the language categories and form a single language category, which is referred to as general language category and also it is call the background model. Then it computes R - Gaussian mixture components from the general language category features. These components are represented with different mixture weights (ω_i), probability (p_i) and covariance matrices (Σ_i). These all parameters define the L_g model. To make use of this model, let X be a set of speech features. Let the set be represented as X = {x_1, x_2, x_3, ……., x_m}. the set consist of m speech features. Each speech feature is N-dimensional feature vector of real numbers. The Log-likelihood for X is then computed with L_g as defined in Equation 4.1:

$$\log p(\mathrm{X} \mid L_g) = \sum_{i=1}^{m} \log p(\mathrm{x}_i \mid L_g) \tag{4.1}$$

Values of Equation 4.1 variables are extended as shown in Equation 4.2 and Equation 4.3,

$$p(x \mid L_g) = \sum_{i=1}^{R} \omega_i p_i(x) \tag{4.2}$$

$$p_i(x) = \frac{1}{\sqrt{(2\pi)^N \mid \Sigma_i \mid}} \exp\left(-\frac{1}{2}(x - \mu_i)^T \Sigma_i^{-1} (x - \mu_i) \right) \tag{4.3}$$

4.7 Specific Language Model

GMM-UBM uses the Bayesian adaption process [86] to adapt the Specific Language Model (*Ls*) from the General Language Model (*Lg*). For this purpose, GMM-UBM processes the speech features of a specific language separately as compared to *Lg*. In case of *Lg*, GMM-UBM combined all the speech features of all the language categories. On the specific language *s*, the GMM-UBM utilizes all its set of features i.e, $X_s = [x_{s_1}, x_{s_2}, x_{s_3}, ..., x_{s_T}]$ for the adaptation process, and adapts the *i*th Gaussian mixture component for *L*s from the *i*th Gaussian mixture component of *Lg* as shown in Equation 4.4:

$$\Pr(i \mid x_{s_t}) = \frac{\omega_i p_i(x_{s_t})}{\sum_{j=1}^{R} \omega_j p_j(x_{s_t})} \tag{4.4}$$

where Pr represents probability. This probability is used to compute the sufficient statistics for the adaptation of weights (ω_i), probability (p_i) and covariance matrices (Σ_i) for the language category *s* and *t* = *{1,2,3,.....T}*, where *T* represents total number of speech features in the language category, *s*, in this thesis *s*= *{1,2,3,4,5}* is used because total number of regional languages are five. The sufficient statistics are computed in Equation 4.5, Equation 4.6 and Equation 4.7 respectively:

$$s_i = \sum_{t=1}^{T} \Pr(i \mid x_{s_t}) \tag{4.5}$$

$$E_i(x) = \frac{1}{s_i} \sum_{t=1}^{T} \Pr(i \mid x_{s_t}) x_{s_t} \tag{4.6}$$

$$E_i(x^2) = \frac{1}{s_i} \sum_{t=1}^{T} \Pr(i \mid x_{s_t}) x_{s_t}^2 \tag{4.7}$$

Then ith mixture for specific language model *Ls* is adapted from the ith mixture of *Lg* as follows form Equation 4.8 to Equation 4.10:

$$\overline{\omega}_i = [\alpha_i^{\omega} n_i / T + (1 - \alpha_i^{\omega})\omega_i]\gamma \tag{4.8}$$

$$\overline{\mu}_i = \alpha_i^{m} E_i(x) + (1 - \alpha_i^{m})\mu_i \tag{4.9}$$

$$\overline{\sigma}_i^2 = \alpha_i^{v} E_i(x^2) + (1 - \alpha_i^{v})(\sigma_i^2 + \mu_i^2) - \overline{\mu}_i^2 \tag{4.10}$$

where the adaptation coefficients are denoted as $\{\alpha_i^{\omega}, \alpha_i^{m}, \alpha_i^{v}\}$, the scale vector is denoted as γ and the adapted mixture weights $\overline{\omega}_i$ are computed in such a manner that they sum to unity, here $\alpha_i^{p}, p \in \{\text{weights } (\omega),\ \textit{mean}\ (m),\ \text{variance } (v)\}$ as defined in Equation 4.11:

$$\alpha_i^{p} = \frac{s_i}{s_i + r^{p}} \tag{4.11}$$

The relevance factor is denoted as r^p and r equals to 16 is used.

4.8 Nearest Neighbour Feature Matching

This thesis proposes Nearest Neighbour Feature Matching (NNFM) strategy to enhance the performance of GMM-UBM for LID. For NNFM we simply take a dataset and divide it into train and test data. Now this trained is categorized into languages given in CRL corpus. At next step these categorized trained samples are trained on classifiers.

Figure 4.3 illustrates the NNFM strategy, and then passed through proposed system that is NNFM. Now the results are stored. In the case of test data, test samples are passed through feature extractor and these extracted features of test data are passed through NNFM for classification and identifying the category of that test feature. If the answer is accurate, that it recognizes the right category then this system accuracy will increase and if does not identify the correct category its means its error rate is greater than accuracy. Figure 4.2 brief description of NNFM diagrammatically.

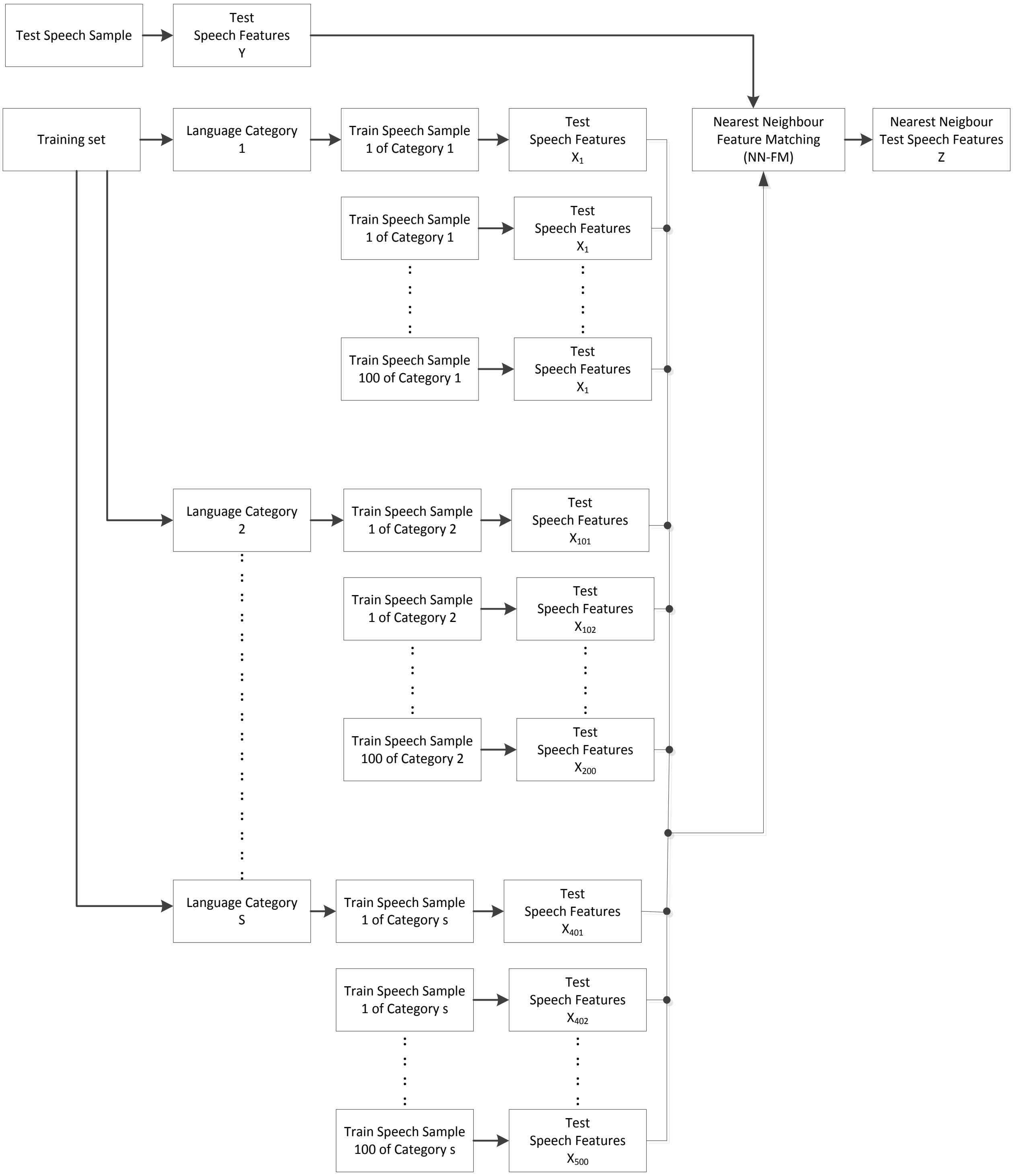

Test Speech Sample
Test Speech Features Y
Training set
Language Category 1
Train Speech Sample 1 of Category 1
Test Speech Features X1
Nearest Neighbour Feature Matching (NN-FM)
Nearest Neigbour Test Speech Features Z
Train Speech Sample 1 of Category 1
Test Speech Features X1
Train Speech Sample 100 of Category 1
Test Speech Features X1
Language Category 2
Train Speech Sample 1 of Category 2
Test Speech Features X101
Train Speech Sample 1 of Category 2
Test Speech Features X102
Train Speech Sample 100 of Category 2
Test Speech Features X200
Language Category S
Train Speech Sample 1 of Category s
Test Speech Features X401
Train Speech Sample 1 of Category s
Test Speech Features X402
Train Speech Sample 100 of Category s
Test Speech Features X500

Figure 4.2: Illustration of proposed Nearest Neighbour Feature Matching strategy

To understand the functioning of NN_FM, let x_i be N-dimensional feature vector. The feature vector x_i is normalized to obtain a unit vector $\bar{x}_i$ as shown in Equation 4.12.

$$\overline{x}_i = \frac{x_i}{\sqrt{x_{i_1}^2 + x_{i_2}^2 + \dots\dots + x_{i_N}^2}} \tag{4.12}$$

This process is repeated on each and every speech feature vectors of both training and test sets. Let Y be a test speech sample, its normalized set of feature vectors are denoted as $\overline{Y} = \{\overline{y}_1, \overline{y}_2, \overline{y}_3, \overline{y}_4, \dots\dots, \overline{y}_k\}$. The set consists of k number of feature vectors. Let $\overline{X}_i = \{\overline{x}_{i_1}, \overline{x}_{i_2}, \overline{x}_{i_3}, \overline{x}_{i_4}, \dots\dots, \overline{x}_{i_m}\}$ represents a set of normalized feature vectors of *i*th training sample of the training set. For this purpose, the training samples of all the language categories are combined. For example, each language category consists of 100 training samples, so total number of sets are *i=1, 2.... S,* where *S = 5 (languages) x 100 = 500.*

Let say there are *m* number of feature vectors in $\overline{X}_i$. Let *di,rp* represents the value of dot product between *r*th feature vector of $\overline{Y}$ and *p*th feature vector of $\overline{X}_i$. The dot product is computed as follows in Equation 4.13.

$$d_{i,rp} = \overline{y}_{r_1}\overline{x}_{i,p_1} + \overline{y}_{r_2}\overline{x}_{i,p_2} + \overline{y}_{r_3}\overline{x}_{i,p_3} + \dots . + \overline{y}_{r_N}\overline{x}_{i,p_N} \tag{4.13}$$

The dimension of $d_{i,rp}$ is S x k x m. It is converted into *di,* r of size S x k as follows in Equation 4.14.

$$d_{i,r} = \arg \max_p d_{i,rp} \tag{4.14}$$

where *di,r* is obtained by computing the maximum value along the third dimension of $d_{i,rp}$ (i.e, *m*). In fact the maximum dot product represents the nearest neighbour of *r*th feature vector of $\overline{Y}$ in the set of feature vectors of $\overline{X}_i$. Then d_i is computed from $d_{i,r}$ as follows in Equation 4.15.

$$d_i = \frac{1}{k}\sum_{r=1}^{k} d_{i,r} \tag{4.15}$$

where d_i represents the mean dot product, computed over all the k feature vectors of $\overline{Y}$ that quantify the similarity between $\overline{Y}$ and $\overline{X}_i$. The size of d_i is S x 1. The maximum value in d_i is identified, let Z be a set of training feature vectors that belong to $\overline{X}_i$ and gives the maximum value. Then all the feature vectors of $\overline{Y}$ are replaced with features vectors of Z. In fact Z is the nearest neighbour sample in the training set for $\overline{Y}$. In the proposed method Z is used to as test features, which are given to GMM-UBM for LID instead of $\overline{Y}$

4.9 Performance criteria

In performance criteria performance of is measured on the basis of accuracy rate and equal error rate. Accuracy measures the performance of system and equal error rate (EER) measures the error in system. The accuracy and error rate are indirectly proportional to each other. As accuracy increases the error rate will automatically decrease.

4.9.1 Accuracy rate

Accuracy is computed from a confusion matrix. The confusion matrix is used to analyze the system performance. Correct and false classification is counted from the confusion matrix. There are four possibilities in confusion matrix for values as illustrated in Table 4.2.

Table 4.2: Possibilities of positive and negative from confusion matrix

T-Positive (True Positive)	Same values (Predicted value= Actual value)
T-Negative (True Negative)	Both are false (Prediction & Actual)
F-Positive (False Positive)	Opposite values (Actual is false and prediction is true)
F-Negative (False Negative)	Opposite values (Actual is true but prediction is false)

In the confusion matrix the predicting about the languages can be seen if the language is correctly classified that prediction and actual values are true, its mean that actual language and prediction language is same. Table 4.3 illustrates an example of the confusion matrix.

Table 4.3: Confusion matrix of language identification

	Urdu	Punjabi	Balochi	Pashto	Sindhi
Urdu	21	1	1	1	1
Punjabi	0	23	0	1	1
Balochi	3	1	19	2	0
Pashto	1	2	1	21	0
Sindhi	0	1	0	0	24

True prediction values are shown as diagonal values, therefore 21+23+19+21+24 = 108. Total prediction of the classifiers is 125. Therefore, to compute the accuracy rate is equal to:

Total Positive / Total prediction = 108/125= 86%

The accuracy rate of the above confusion matrix is 86%.

4.9.2 Equal Error Rate

Error rate defines the error or defaults in system. Mostly error rate is checked frequently to avoid the errors in system. The EER is the error rate when false acceptance rate and rejection rate are matched with each other. Lower the equal error rate value, higher the accuracy rate of the system.

4.10 Summary

This section discusses the information of dataset use in this thesis. It also explains the process to collect the data, formatting in data and storing of data. Division of the data for training and testing is also explained in this chapter. A specific language model and general model is the part of this chapter. Classifier is used to train the machine in this section. Performance criteria are also the part of the chapter which includes accuracy rate and equal error rate.

CHAPTER 5

EXPERIMENT SETUP AND RESULTS

5.1 Overview

The chapter presents the experiment results. Experimental results are based on the corpuses. Three main corpuses are discussed. Further the experimental setup is described which consists of Speech corpuses, speech features and classifiers used. Then experimental results are presented using the proposed NNFM method. The results are divided into two parts (i) Comparison of speech features and classifiers for LID (ii) Comparison of state of art are with the proposed NNFM method.

5.2 Speech corpuses

Two corpuses are discussed that are Corpus of Regional Language (CRL) that is generated for this thesis and Call-friend corpus that is already generated and used in many applications. This second one is modified for comparison with our corpus (CRL), in which some of its languages are taken and set according to our requirements. These corpuses are further explained in this section.

5.3 Corpus of Regional Languages

This corpus is referred to as Corpus of Regional Languages (CRL) and consists of five different language of Pakistan which is Balochi, Pashto, Punjabi, Sindhi and Urdu. Each language of this corpus consists of 150 samples. Each sample is of 15 secs. Each sample is in .wav format and sampled at 16000 Hz. The samples of each language category is randomly divided into two disjoint sets, the first set for training (100 samples) and other one for testing (remaining 50 samples). Table 5.1 summarizes the setup of CRL for experimentation.

Table 5.1: Number of samples and duration of samples

Language	Training Samples	Testing Samples	Duration
Punjabi	100	50	15secs
Sindhi	100	50	15secs
Pashto	100	50	15secs
Balochi	100	50	15secs
Urdu	100	50	15secs

5.4 Call-friend Corpus

This corpus is referred to as Call-Friend Corpus (CFC) [7]. The corpus consists of six different languages. The languages are English, Korean, Chinese, Taiwan, Japanese, German, French, Spanish, Arabic, Farsi, Tamil and Mandarin. The samples in each language category contain a telephone conversation between two persons in the same language. Each sample is of different durations in minute's i.e, 5-30 minutes. From these samples, the speech samples of 15 secs are randomly selected. So that there are 300 speech samples per language category. The corpus is text independent and speaker independent. In each telephonic conversation, both the caller and callee are native speakers of the same language. All calls are recorded inside United States and Canada. Call-Friend corpus is taken in an open environment. It is recorded from different calls obtained from different speakers. It is in .wav format with 16 KHz as sampling rate and has been widely used for LID. Table 5.2 summarizes the Call-Friend corpus used in this thesis for experiments. The samples of each language category consist of 300 samples. Each sample is of 15 secs. The sample of each category is randomly divided into two sets. The first set consists of 200 samples for training purpose and the other one consist of 100 samples for testing and validation.

Table 5.2: Modified Call-friend Corpus

Language	**Training Samples**	**Testing Samples**	**Duration**
English	200	100	15secs
Japanese	200	100	15secs
German	200	100	15secs
French	200	100	15secs
Spanish	200	100	15secs
Mandarin	200	100	15secs

In Table 5.3 a description of parameters of samples of Call-friend corpus is explained. The format of record the recording samples from the speaker and format of saving those samples is also explained. Sampling rate is given as well. Sources to get these recordings are also the part of the table. Invention region of this corpus is also given in this section. Duration of the samples is present in the table.

Table 5.3: Attributes of Call-friend corpus

S. No	**Parameters**	**Value**
1	File format (input)	.wav
2	Sampling rate	8kHz
3	Bit rate	8

4	Type of channel	Mono
5	Data source	Telephone calls
6	Application	Language identification
7	Region	United states of America, Canada
8	Target	MFCC
10	Time duration	5-30 minutes

5.5 Speech Features

Different speech features are extracted from the speech samples of CRL and CFC corpuses. These features are MFCC, LPCC and SDC. In the experiments if MFCC issued for training that MFCC is also applied on the test samples for classification, similarly for other features.

5.6 Classifiers

GMM-UBM is used as a main classification algorithm in this thesis. In-fact NNFM is proposed to improve the performance of GMM-UBM method. Other classifiers used are SVM, KNN and I-vector

5.7 Comparison of Speech features and Classifiers for LID

This section presents a comparison of MFCC, LPCC and SDC using GMM-UBM and I-vector methods. For GMM-UBM different Gaussian Mixture component are extracted. These components are 32, 64, 128, 256, 512, 1024, 2048 and 4096. Accuracy and EER rate are used as metrics for comparison. Table 5.4 shows an EER based comparison between MFCC, LPCC and SDC on CRL. GMM-UBM is used a classifier. The CRL has 5 (languages) x 50 = 250 test samples. The compression shows that MFCC obtains 34% EER with 32 Gaussian components, whereas SDC and LPCC achieve 45.3% and 51.3% EER respectively. So MFCC outperforms SDC and LPCC with 32 Gaussian components. The Table 5.4 shows that with increase in the number of mixture components the EER decreases. For example, in case of 4096 components MFCC, LPCC and SDC achieve EER of 27.3 %, 36.7% and 48.2%. The comparison shows that MFCC outperforms SDC and LPCC if GMM-UBM is used with different mixture components.

Table 5.4: EER (%) based comparison between features using GMM-UBM on CRL

Gaussian Mixture Components	32	64	128	256	512	1024	2048	4096
MFCC	34.0	35.7	33.5	33.2	31.2	31.3	29.0	27.3
SDC	45.3	44.7	42.7	42.6	42.5	40.0	39.0	36.7
LPCC	51.3	51.2	50.7	50.3	49.9	49.3	49.0	48.2

Table 5.5 shows accuracy based comparison between MFCC, LPCC and SDC on CRL. GMM-UBM is used a classifier. The accuracy is computed on 250 test samples of CRL. The comparison shows that MFCC obtains 39.3% EER with 32 Gaussian components, whereas SDC and LPCC achieve 28.7% and 20% accuracy, respectively. So MFCC outperforms SDC and LPCC with 32 Gaussian components. The table shows that with increase in the number of mixture components the accuracy increases. For example, with 4096 components MFCC, LPCC and SDC demonstrates accuracy of 53.3%, 40.7% and 24.7%. The comparison shows that MFCC outperforms SDC and LPCC using GMM-UBM with different mixture components.

Table 5.5: Accuracy (%) based comparison between features using GMM-UBM on CRL

Gaussian Mixture Components	32	64	128	256	512	1024	2048	4096
MFCC	39.3	40.0	42.7	46.7	48.7	50.7	50.0	53.3
SDC	28.7	32.3	33.5	33.9	34.7	35.3	36.0	40.7
LPCC	20.0	20.0	20.0	20.0	20.7	20.7	22.7	24.7

Table 5.6 shows an EER and Accuracy based comparison between MFCC, LPCC and SDC. In this comparison different duration test samples are used for example 3, 5, 10 and 15 seconds. The aim of this comparison is to find how well The GMM-UBM call identify the language in short utterances. All the values listed in this table are obtained with 4096 Gaussian components. The selection of 4096 components is based on the above results which show that all the speech features achieves the lowest EER and high accuracy.

In Table 5.6 the comparison shows that as the duration of test speech utterance increase the EER decrease at the same time the accuracy increases. MFCC achieves the best EER for each evaluated test sample duration. It achieves EER of 32.7%, 29.0%, 27.5% and 27.3% for 3, 5, 10 and 15 seconds duration respectively and out performs LPCC and SDC. The comparison shows that accuracy achieved by MFCC for different test sample duration is also better than LPCC and SDC. It achieves the overall the best accuracy of 53.3% for 15 seconds duration.

Table 5.6: Comparison between different speech features on CRL using GMM-UBM and different test sample durations

	EER (%)				Accuracy (%)			
Duration of test samples	3 secs	5 secs	10 secs	15 secs	3 secs	5 secs	10 secs	15 secs
MFCC	32.7	29.0	27.5	27.3	48.0	50.0	52.0	53.3
SDC	40.7	39.2	37.0	36.7	38.0	38.7	39.2	40.7
LPCC	51.2	50.3	49.5	48.2	24.0	24.3	24.5	24.7

Now i-vector classifiers are used for a comparison between the features. In the i-vector based comparison the objective is to channel how the MFCC perform with respect to other features, as the previous results shows that MFCC outperform other if GMM-UBM is used.

In Table 5.7, EER based comparison is shown between MFCC, LPCC and SDC. I-vector classifier is used. I-vector is trained with different Gaussian mixture components which are 32, 64,128, 256, 512, 1024, 2048, and 4096 components. The purpose of assuming different *t* components is to analyze the performance of i-vector how the components affects its performance. The comparison shows that MFCC obtain 36% EER and performs others on 32 mixture components similarly MFCC performs other feature on the Gaussian mixture components. So MFCC shows super performance. We compare the MFCC with GMM-UBM it can be seen that with same mixture components of 4096, it gives 27.3% EER whereas with same component and i-vector it achieves 28%. So the combining MFCC with GMM-UBM better performance is obtained compared to comprising wit with i-vector. In the comparison achieves the least performance.

Table 5.7: EER (%) based comparison between features using i-vector classifier on CRL

Gaussian Mixture Components	32	64	128	256	512	1024	2048	4096
MFCC	36.0	34.3	32.3	31.0	30.7	30.7	28.7	28.0
SDC	44.7	44.5	43.3	43.5	43.2	41.3	39.3	37.0
LPCC	52.3	52.0	52.0	52.9	51.5	50.2	50.0	48.0

Table 5.8 shows comparison better feature on CRL using the i-vector classifier. With the Gaussian components of 4096, all the features achieve the best the accuracy rates. It can be seen that MFCC demonstrate the best accuracy with i-vector as well. The accuracy, it achieves with 4096 component and i-vector is 50% whereas with sample components and GMM-UBM it gives an accuracy of 53.4 %.

Figure 5.1 shows the comparison between the speech features using i-vector on CRL. Different duration test utterances are used. It can be seen that with increase in the duration of utterance the performance of i-vector based feature performance increases. With 15 secs EER obtained by features are lower that other duration similarly with 15 secs the accuracy is also better.

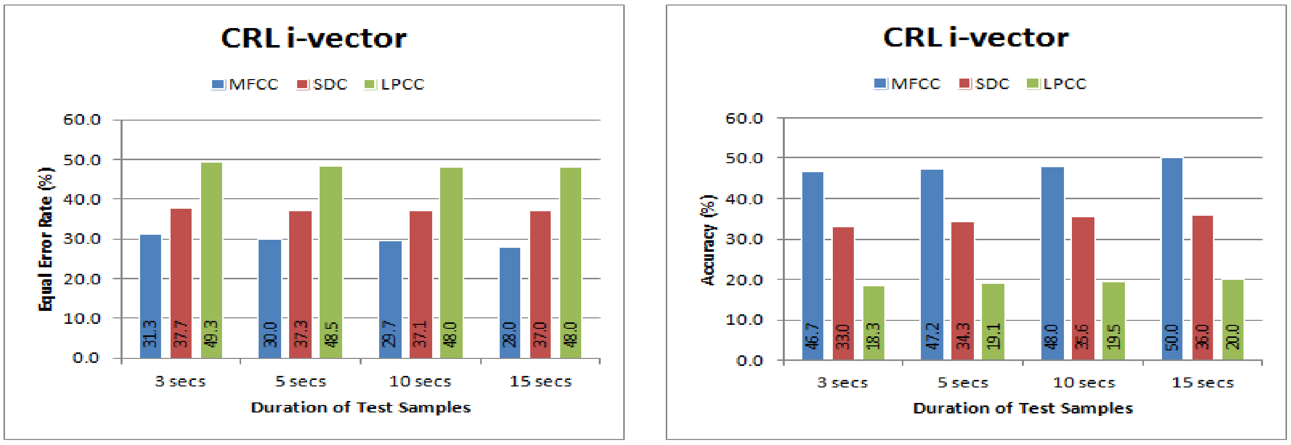
Table 5.8: Accuracy (%) based comparison between features using i-vector classifier on CRL

Gaussian Mixture Components	32	64	128	256	512	1024	2048	4096
MFCC	38.0	40.0	45.3	46.7	47.3	48.0	49.3	50.0
SDC	30.0	35.3	35.3	35.6	35.7	35.7	36.0	36.0
LPCC	15.3	16.7	16.7	16.7	18.7	19.3	19.3	20.0

Figure 5.2 shows evaluation speech features and classifiers on the Call Friend corpus. The results are obtained with GMM-UBM and i-vector. For both GMM-UBM and i-vector, 4096 Gaussian mixture components are computed. The test utterances are of different duration. In case of GMM-UBM it can be seen that MFCC with 15 sec duration obtain the least ER of 37.1 % and outperforms other features. Similarly, GMM-UBM with MFCC obtains the 36.6% accuracy for 15 secs duration and out performs other. Similarly, the i-vector based comparison also show the at MFCC is best feature compared to LPCC and SDC.

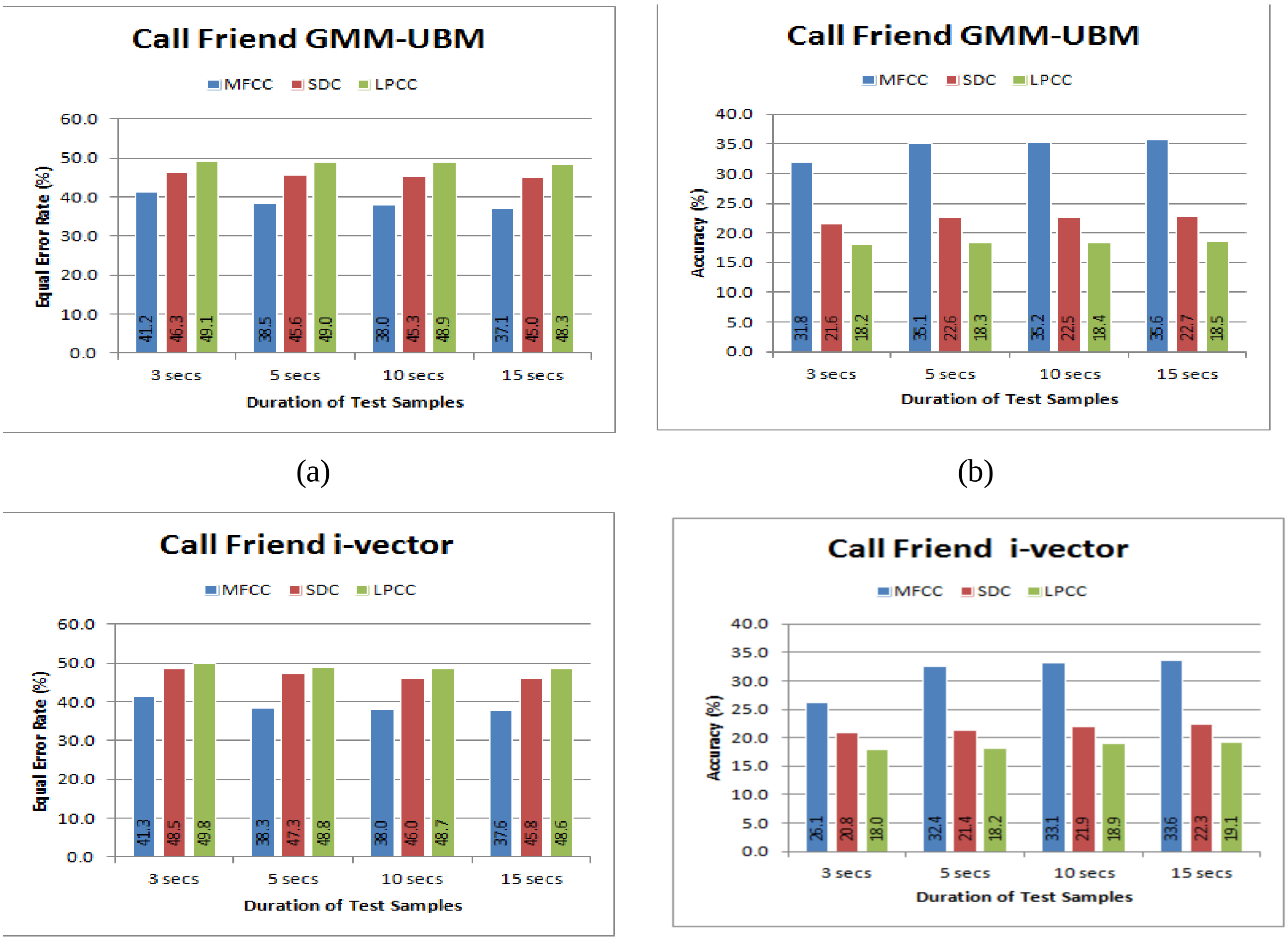
5.8 Comparison using Proposed Method

In this section, the proposed NNFM method is used. It is showing that how the performance of feature and classifier vary with the performed method. In Table 5.9, the GMM-UBM is used on CRL. The features are evaluated on different duration utterances. It can be seen that all the features demonstrate good performance on 15 secs samples.



(a) (b)

Figure 5.1: Comparison of speech features using i-vector method on CRL. Different duration of test utterances are used



(a) (b)

(c) (d)

Figure 5.2: Comparison between feature and classifier on Call-friend corpus (a)-(b) Comparison between speech features in the GMM-UBM (c)-(d) Comparison between speech features in i-vector method

A notation MFCC+NNFM in the comparison denotes that MFCC features are used with the proposed NNFM methods where as MFCC is with the NNFM. For instance, MFCC achieves 32.7% with 3 secs utterances then MFCC+NNFM achieves 26.7%. So improvement in the EER is almost 6%. Similarly, it can be seen that NNFM also improve the EER of other features. NNFM also improves the accuracy of LID.

Table 5.9: GMM-UBM based comparison between features with and without using the proposed NNFM method on CRL. With using the NNFM method is denoted with '+' signs

	EER (%)				**Accuracy (%)**			
Duration	**3 secs**	**5 secs**	**10 secs**	**15 secs**	**3 secs**	**5 secs**	**10 secs**	**15 secs**
MFCC	32.7	29.0	27.5	27.3	48.0	50.0	52.0	53.3
SDC	40.7	39.2	37.0	36.7	38.0	38.7	39.2	40.7
LPCC	51.2	50.3	49.5	48.2	24.0	24.3	24.5	24.7
MFCC+ NNFM	26.7	24.7	22.0	21.5	56.7	60.7	63.3	65.3
SDC+ NNFM	31.7	33.3	31.7	30.7	44.0	45.3	46.0	47.3
LPCC+ NNFM	44.8	44.0	43.2	41.3	27.3	29.2	30.7	32.0

For instance, with 15 secs MFCC, SDC and LPCC demonstrates accuracy of 53.3%, 40.7% and 24.7% without using NNFM respectively. With NNFM they demonstrate better accuracy rates i.e, 65.3%, 47.3% and 32.0%, respectively. So, improvement in accuracy is 12%, 6.6% and 7.3% respectively. It can be seen with NNFM is more compatible with MFCC then LPCC and SDC.

Table 5.10 shows comparison based on i-vector method on CRL with and without using the proposed NNFM methods. With using the proposed method is denoted with '+' signs. It can be seen that with using NNFM all speech features achieve better results than without using the proposed method. Compared to i-vector, the GMM-UBM based results shown in Table 5.10 are better. So all the results show that GMM-UBM is the best classifier and MFCC is the best features for LID on CRL. With NNFM the performance of GMM-UBM and MFCC can be further boosted.

Table 5.10:i-vector based comparison between features with and without using the proposed NNFM method on CRL. With using the NNFM method is denoted with '+' signs

	EER (%)				Accuracy (%)			
Duration	**3 secs**	**5 secs**	**10 secs**	**15 secs**	**3 secs**	**5 secs**	**10 secs**	**15 secs**
MFCC	31.3	30.0	29.7	28.0	46.7	47.2	48.0	50.0
SDC	37.7	37.3	37.1	37.0	33.0	34.3	35.6	36.0
LPCC	49.3	48.5	48.0	48.0	18.3	19.1	19.5	20.0
MFCC+ NNFM	30.0	29.2	26.7	26.0	52.1	53.9	55.2	58.7
SDC+ NNFM	33.3	33.3	32.2	32.2	40.7	42.7	46.0	46.0
LPCC+ NNFM	46.5	46.0	45.3	45.0	27.3	28.0	29.3	30.7

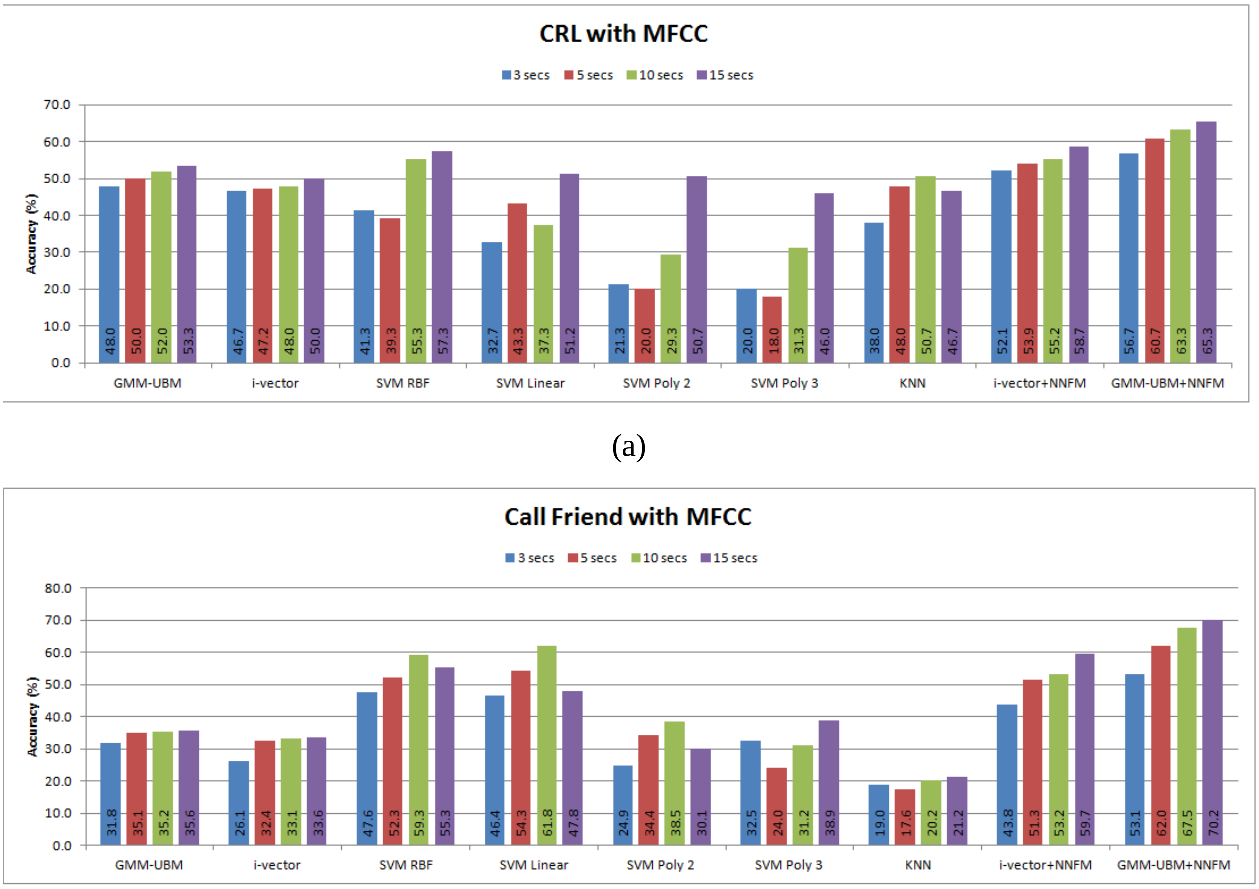
Table 5.11: GMM-UBM based comparison between features with and without using the proposed NNFM method on Call Friend Corpus. With using the NNFM method is denoted with '+' signs

	EER (%)				Accuracy (%)			
Duration	**3 secs**	**5 secs**	**10 secs**	**15 secs**	**3 secs**	**5 secs**	**10 secs**	**15 secs**
MFCC	41.2	38.5	38.0	37.1	31.8	35.1	35.2	35.6
SDC	46.3	45.6	45.3	45.0	21.6	22.6	22.5	22.7
LPCC	49.1	49.0	48.9	48.3	18.2	18.3	18.4	18.5
MFCC+ NNFM	25.3	22.3	18.3	17.8	53.1	62.0	67.5	70.2
SDC+ NNFM	44.3	43.6	42.4	42.2	22.5	24.8	25.4	26.9
LPCC+ NNFM	47.3	47.3	46.9	46.0	19.6	20.3	20.5	20.8

Table 5.12: i-vector based comparison between features with and without using the proposed NNFM method on Call friend Corpus. With using the NNFM method is denoted with '+' signs.

	EER (%)				Accuracy (%)			
Duration	**3 secs**	**5 secs**	**10 secs**	**15 secs**	**3 secs**	**5 secs**	**10 secs**	**15 secs**
MFCC	41.3	38.3	38.0	37.6	26.1	32.4	33.1	33.6
SDC	48.5	47.3	46.0	45.8	20.8	21.4	21.9	22.3
LPCC	49.8	48.8	48.7	48.6	18.0	18.2	18.9	19.1
MFCC+ NNFM	31.2	26.5	25.2	22.3	43.8	51.3	53.2	59.7
SDC+ NNFM	45.5	45.1	43.8	43.7	22.4	24.3	25.0	26.1
LPCC+ NNFM	47.7	47.3	47.2	46.9	19.0	19.5	19.8	20.3

Table 5.11 and Table 5.12 show comparison on Call-friend corpus with and without using the NNFM method. Comparison in both tables shows that NNFM efficient improves the performance of all the feature and classifier. On call friend corpus as well, this suggest the propose NNFM works CRL as well on call friend corpus and it is not corpus depended. The results on call friend shows that GMM-UBM is the best classifier and MFCC is the best feature for LID and their performances can be boosted efficiently with NNFM. In Figure 5.3, a comparison of different classifiers with MFCC features is presented on CRL and Call friend corpuses. The classifier used are GMM-UBM, SVM Linear, SVM (Polynomial kernel of degree 2 and 3), KNN (K=1), i-vector + NNFM and GMM-UBM+FMNN. It can be seen that GMM-UBM combination with NNFM outperformed all the classifiers on bite corpuses. It can be seen that the performance of KNN is not impressive compare to NNFM. The proposed NNFM also outperformed SVM classifier. It can be seen that tall classifier gives better results on 15 secs duration utterances. Decrease in the duration decrease the accuracy rate. On CRL and Call-friend the best accuracy achieved is 65.3% and 70.2 % using GMM-UBM+NNFM respectively.



(a)

(b)

Figure 5.3: Comparison of different classification methods using MFCC features (a) CRL (b) Call Friend Corpus

5.9 Summary

Proposed system and methodology is discussed in this chapter. Classifiers and feature extractors are used in this chapter. Training of the machine is the initial step. Process of extraction of the features is the base of machine training stage. From all the recording samples, some samples are used to train the machine with the help of classifiers and remaining are used for testing phase.

CHAPTER 6

CONCLUSION AND FUTURE WORK

6.1 Overview

This section summarizes the conclusion part of the thesis. Main contributions of the thesis experiments are evaluated on its basis and their results are discussed. Furthermore, this chapter presents limitations and future work that can be done in this field.

6.2 Conclusion

A new method, which is referred to as Nearest Neighbour Feature Matching (NNFM) for automatic identification of regional languages of Pakistan, was proposed. The proposed method is tested on Urdu, Punjabi, Sindhi, Balochi and Pashto spoken utterances which provides accurate results on the test samples. The proposed method is also tested on Call Friend corpus consisting of six international languages. The experimental results show superior performance of the proposed method.

Three different types of feature extraction schemes are evaluated on the regional languages, which are MFCC, LPCC and SDC. The experimental results show that MFCC compared to LPCC and SDC gives better results.

Experimental results achieved with different classifiers such as GMM-UBM, i-vector, SVM and KNN show that the GMM-UBM classifier gives accurate results compared to others on both regional languages and the languages of Call Friend corpus. However, using the proposed NNFM with GMM-UBM even better results are obtained. The experimental results show that proposed NNFM improves the performance of GMM-UBM by almost 8.7% to 12%.

Experimental on different duration test utterances are performed. The utterances of duration three, five, ten and fifteen seconds are used. The experimental results show that GMM-UBM with NNFM gives accuracies of 56.7%, 60.7%, 63.3% and 65.3% on three, five, ten and fifteen seconds utterances. This shows that with increase in the duration of short utterances, the accuracy of LID increases.

6.3 Limitations and Future work

This thesis is limited to few regional languages. Only Urdu and four provincial languages (Punjabi, Pashto, Sindhi and Balochi) of Pakistan are observed. For future work more languages can be explored. Sub regional languages can also be considered such as Hindko and Saraiki.

Only there feature extraction techniques like MFCC, LPCC and SDC are evaluated. In future work more feature extraction techniques can also be observed. In a same way some only GMM, I-vector, SVM and KNN are used for training on speech features. Other schemes such as deep learning can be used. In this thesis best system performance and efficiency is obtained with the combination of MFCC features and GMM classifier. Other combinations of features and classifiers can also be used for comparison.

Data is collected through internet sources; system is trained on text and speaker independent corpus. In future text and speaker can also be dependent. And data can be collected through other sources such as live recording, talk shows, interviews and telephonic conversations.

REFERENCES

[1] D. A. Reynolds, N. Dehak, P. A. Torres-carrasquillo, D. Reynolds, and R. Dehak, “Language Recognition via Ivectors and Dimensionality Reduction,” no. May, 2014.

[2] H. Li, B. Ma, and K. A. Lee, “Spoken Language Recognition: From Fundamentals to Practice,” *Proc. IEEE*, vol. 101, no. 5, pp. 1136–1159, May 2013.

[3] P. Kumar, A. Biswas, a . N. Mishra, and M. Chandra, “Spoken Language Identification Using Hybrid Feature Extraction Methods,” *J. Telecommun.*, vol. 1, no. 2, pp. 11–15, Mar. 2010.

[4] U. Shrawankar and V. M. Thakare, “Techniques for Feature Extraction In Speech Recognition System: A Comparative Study,” *Int. J. Comput. Appl. Eng. Technol. Sci.*, pp. 412–418, May 2010.

[5] E. Timoshenko, “Rhythm Information for Automated Spoken Language Identification,” 2012.

[6] S. Manchala, V. Kamakshi Prasad, and V. Janaki, “GMM based language identification system using robust features,” *Int. J. Speech Technol.*, vol. 17, no. 2, pp. 99–105, Jun. 2014.

[7] P. Matejka, P. Schwarz, J. H. Cernock\`y, and P. Chytil, “Phonotactic language identification using high quality phoneme recognition,” in *International Conference on Speech Communication and Technology*, 2005, pp. 2237–2240.

[8] I. Lopez-Moreno, J. Gonzalez-Dominguez, O. Plchot, D. Martinez, J. Gonzalez-Rodriguez, and P. Moreno, “Automatic language identification using deep neural networks,” in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2014, pp. 5337–5341.

[9] H. Gupta and D. Gupta, “LPC AND LPCC METHOD OF FEATURE EXTRACTION IN SPEECH RECOGNITION,” pp. 498–502, 2016.

[10] C. Vimala and V. Radha, “A Review on Speech Recognition Challenges and Approaches,” *World Comput. Sci. Inf. Technol. J.*, vol. 2, no. 1, pp. 2221–741, 2012.

[11] K. Kumar, R. Aggarwal, and A. Jain, “A Hindi speech recognition system for connected words using HTK,” *Int. J. Comput. Syst. Eng.*, vol. 1, no. 1, p. 25, 2012.

[12] M. A. Zissman, "Language identification using phoneme recognition and phonotactic language modeling," in *nternational Conference on Acoustics, Speech, and Signal Processing*, 1995, vol. 5, pp. 3503–3506.

[13] E. Barnard, "Random walk theory applied to language identification." .

[14] J. L. Hieronymus and S. Kadambe, "Robust spoken language identification using large vocabulary speech recognition," in *IEEE International Conference on Acoustics, Speech, and Signal Processing*, 1997, vol. 2, pp. 1111–1114.

[15] H. Zen, K. Tokuda, T. Masuko, T. Kobayashi, and T. Kitamura, "ISCA Archive Hidden Semi-Markov Model Based Speech Synthesis," no. 5, pp. 1–4, 2004.

[16] T. V Duong, H. H. Bui, D. Q. Phung, S. Venkatesh, R. Ave, and M. Park, "Activity Recognition and Abnormality Detection with the Switching Hidden Semi-Markov Model," 2005.

[17] J. Ma, B. Zhang, S. Matsoukas, S. H. Mallidi, F. Li, and H. Hermansky, "Improvements in Language Identification on the RATS Noisy Speech Corpus Raytheon BBN Technologies , USA 2 . The RATS LID Data Corpus," *Interspeech*, no. August, pp. 69–73, 2013.

[18] Kevin Micheal Squire, "HMM-BASED SEMANTIC LEARNING FOR A MOBILE ROBOT," 2004.

[19] S. Dupont and J. Luettin, "Audio-visual speech modeling for continuous speech recognition," *IEEE Trans. Multimed.*, vol. 2, no. 3, pp. 141–151, 2000.

[20] W. Ghai and N. Singh, "Literature Review on Automatic Speech Recognition," *Int. J. Comput. Appl.*, vol. 41, no. 8, pp. 42–50, Mar. 2012.

[21] M. Li and S. Narayanan, "Simplified supervised i-vector modeling with application to robust and efficient language identification and speaker verification," *Comput. Speech Lang.*, vol. 28, no. 4, pp. 940–958, Jul. 2014.

[22] P. Y. Santosh K. Gaikwad, Bharti W. Gawali, "A review on Speech Recognition Technique," *Int. J. Comput. Appl.*, vol. 10–No 3, no. 3, pp. 16–24, 2010.

[23] V. Ramu Reddy, S. Maity, and K. Sreenivasa Rao, "Identification of Indian languages using multi-level spectral and prosodic features," *Int. J. Speech Technol.*, vol. 16, no. 4, pp. 489–511, Dec. 2013.

[24] S. M. Siniscalchi, J. Reed, T. T. Svendsen, and C.-H. H. Lee, "Universal attribute characterization of spoken languages for automatic spoken language recognition," *Comput. Speech Lang.*, vol. 27, no. 1, pp. 209–227, Jan. 2013.

[25] P. S. Section, R. S. Processing, and R. Environments, "Automatic Language Identification on SVM with Discriminative Language Characterization," no. 3, 2008.

[26] E. Ambikairajah, H. Li, L. Wang, B. Yin, and V. Sethu, "Language Identification: A Tutorial," *IEEE Circuits Syst. Mag.*, vol. 11, no. 2, pp. 82–108, 2011.

[27] S. Therese and C. Lingam, "Review of Feature Extraction Techniques in Automatic Speech Recognition," *Int. J. Sci. Eng. Technol.*, vol. 484, no. 2, pp. 479–484, 2013.

[28] M. E. U. I. Dar, S. Ahmad, T. Habib, H. Shaheen, and M. A. Hussain, "An Overview of Text-Independent Speaker Recognition: from Features to Supervectors," *J. Food, Agric. Environ.*, vol. 12, no. 2, pp. 922–925, 2014.

[29] V. Tiwari, "MFCC and its applications in speaker recognition," *Int. J. Emerg. Technol.*, vol. 1, no. 1, pp. 19–22, 2010.

[30] S. Nakagawa, L. Wang, and S. Ohtsuka, "Speaker identification and verification by combining MFCC and phase information," *IEEE Trans. Audio, Speech Lang. Process.*, vol. 20, no. 4, pp. 1085–1095, 2012.

[31] Y. Song, X. Hong, B. Jiang, R. Cui, I. McLoughlin, and L. Dai, "Deep bottleneck network based i-vector representation for language identification," *Annu. Conf. Int. Speech Commun. Assoc.*, vol. 2015–Janua, pp. 398–402, 2015.

[32] J. Gonzalez-Dominguez, I. Lopez-Moreno, P. J. Moreno, and J. Gonzalez-Rodriguez, "Frame-by-frame language identification in short utterances using deep neural networks," *Neural Networks*, vol. 64, pp. 49–58, Apr. 2015.

[33] A. W. Abbas, N. Ahmad, and H. Ali, "Pashto spoken digits database for the automatic speech recognition research," *18th Int. Conf. Autom. Comput. Integr. Des. Eng.*, no. September, pp. 348–351, 2012.

[34] M. Dua, R. K. Aggarwal, V. Kadyan, and S. Dua, "Punjabi Automatic Speech Recognition Using HTK," *Int. J. Comput. Sci. Issues*, vol. 9, no. 4, pp. 359–364, 2012.

[35] I. Ahmed, N. Ahmad, H. Ali, and G. Ahmad, "The Development of Isolated Words Pashto

Automatic Speech Recognition System," in *18th International International Conference on Automation & Computing*, 2012, no. September, pp. 348–351.

[36] M. Itrat, S. A. Ali, R. Asif, K. Khanzada, and M. K. Rathi, "Automatic Language Identification for Languages of Pakistan," vol. 17, no. 2, pp. 161–169, 2017.

[37] Y. K. Muthusamy, E. Barnard, and R. A. Cole, "Automatic Language Identification: A Review / Tutorial 1 Introduction 2 Sources of Information Useful for Language ID," pp. 1–16.

[38] M. A. M. Anusuya and S. K. Katti, "Speech Recognition by Machine, A Review," *Int. J. Comput. Sci. Inf. Secur.*, vol. 6, no. 3, pp. 181–205, Jan. 2009.

[39] Bing-Hwang Juang and S. Furui, "Automatic recognition and understanding of spoken language - a first step toward natural human-machine communication," *Proc. IEEE*, vol. 88, no. 8, pp. 1142–1165, Aug. 2000.

[40] F. J. Goodman, A. F. Martin, and R. E. Wohlford, "Improved automatic language identification in noisy speech," in *Acoustics, Speech, and Signal Processing, 1989. ICASSP-89., 1989 international conference on*, 1989, pp. 528–531.

[41] R. M. Stern and N. Morgan, "Hearing is believing - biologically-inspired feature extraction for robust automatic speech recognition," *Signal Process. Mag. IEEE*, vol. 29, no. 6, pp. 34–43, 2012.

[42] M. A. Zissman, "Comparison of four approaches to automatic language identification of telephone speech," *IEEE Trans. Speech Audio Process.*, vol. 4, no. 1, p. 31, Jan. 1996.

[43] M. Sugiyama, "Automatic language recognition using acoustic features," in *IEEE International Conference on Acoustics, Speech, and Signal Processing*, 1991, pp. 813–816.

[44] M. Savic, E. Acosta, and S. Gupta, "An automatic language identification system," in *IEEE International Conference on Acoustics, Speech, and Signal Processing*, 1991, pp. 817–820.

[45] M. a. Zissman, "Automatic language identification using Gaussian mixture and hidden Markov models," in *IEEE International Conference on Acoustics Speech and Signal Processing*, 1993, vol. 2, pp. 399–402 vol.2.

[46] L. F. Lamel and J. L. Gauvain, "Language identification using phone-based acoustic likelihoods," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 1977, vol. i, p. I/293-I/296.

[47] Y. K. Muthusamy, R. A. Cole, and B. T. Oshika, "The OGI multi-language telephone speech corpus," in *Second International Conference on Spoken Language Processing*, 1992.

[48] R. A. C. Yeshwant K. Muthusamy, "Automatic Segmentation and Identification of Ten Languages Using Telephone Speech," in *International Conference on Spoken Language Processing*, 1992, pp. 321–324.

[49] N. K. Hiroshi Irii, Kenzo Ito, "Multilingual speech data base for evaluating quality of digitized speech," in *First International Conference on Spoken Language Processing*, 1990, pp. 1025–1028.

[50] L. Riek, W. Mistretta, and D. Morgan, "Experiments in language identification," *Lockheed Sanders, Inc., Nashua, NH, Tech. Rep. SPCOT-91-002*, 1991.

[51] P. A. Torres-Carrasquillo, E. Singer, M. A. Kohler, R. J. Greene, D. A. Reynolds, and J. R. Deller, "Approaches to language identification using Gaussian mixture models and shifted delta cepstral features," in *7th International Conference on Spoken Language Processing*, 2002, vol. 2, pp. 89–92.

[52] Kung-Pu Li, "Automatic language identification using syllabic spectral features," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 1994, vol. i, p. I/297-I/300.

[53] F. Allen, E. Ambikairajah, and J. Epps, "Language Identification using Warping and the Shifted Delta Cepstrum," in *IEEE 7th Workshop on Multimedia Signal Processing*, 2005, no. December, pp. 1–4.

[54] J. Foil, "Language identification using noisy speech," *IEEE Int. Conf. Acoust. Speech, Signal Process.*, vol. 11, pp. 861–864, 1986.

[55] K. Li and T. Edwards, "Statistical models for automatic language identification," in *IEEE International Conference on Acoustics, Speech, and Signal Processing*, 1980, vol. 5, pp. 884–887.

[56] Y. K. Muthusamy, R. A. Cole, and M. Gopalakrishnan, “A segment-based approach to automatic language identification,” in *IEEE International Conference on Acoustics, Speech, and Signal Processing*, 1991, pp. 353–356.

[57] B. Ma, H. Li, and C. Lee, “An Acoustic Segment Modeling Approach to Automatic Language Identification,” in *Ninth European Conference on Speech Communication and Technology*, 2005, pp. 2829–2832.

[58] V. Chandrasekhar, M. Emre Sargin, and D. A. Ross, “Automatic Language Identification in music videos with low level audio and visual features,” in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2011, pp. 5724–5727.

[59] A. Dustor and P. Szwarc, “Spoken language identification based on GMM models,” *Int. Conf. signals Electron. Syst.*, pp. 105–108, 2010.

[60] E. Singer, P. Torres-Carrasquillo, T. P. Gleason, W. M. Campbell, and D. A. Reynolds, “Acoustic, phonetic, and discriminative approaches to automatic language identification.,” in *Eurospeech*, 2003, pp. 1345–1348.

[61] S. Maity, A. Kumar Vuppala, K. S. Rao, and D. Nandi, “IITKGP-MLILSC speech database for language identification,” in *National Conference on Communications (NCC)*, 2012, pp. 1–5.

[62] W. M. Campbell, E. Singer, P. a. Torres-Carrasquillo, and D. a. Reynolds, “Language recognition with support vector machines,” in *The Speaker and Language Recognition Workshop*, 2004, no. 1, pp. 1–4.

[63] N. F. Chen, B. Ma, and H. Li, “Minimal-resource phonetic language models to summarize untranscribed speech,” in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2013, no. May 2015, pp. 8357–8361.

[64] D. B. Paul and J. M. Baker, “The design for the wall street journal-based CSR corpus,” in *Workshop on Speech and Natural Language*, 1992, p. 357.

[65] P. Schwarz, P. Matejka, L. Burget, and O. Glembek, “Phoneme recognizer based on long temporal context,” *Speech Process. Group, Fac. Inf. Technol. Brno Univ. Technol. Available http//speech. fit. vutbr. cz/en/software*, 2006.

[66] M. Penagarikano, A. Varona, M. Zamalloa, L. J. Rodriguez, G. B. G, and P. Uribe,

"University of the Basque Country + Ikerlan System for NIST 2009 Language Recognition Evaluation," *NIST Lang. Recognit. Eval. Work.*, no. January, 2009.

[67] J. Gonzalez-Dominguez, I. Lopez-Moreno, J. Gonzalez-Rodriguez, and P. J. Moreno, "Automatic Language Identification using Long Short-Term Memory Recurrent Neural Networks," in *15th Annual Conference of the International Speech Communication Association*, 2014, no. September, pp. 2155–2159.

[68] P. K. Polasi and K. S. R. Krishna, "Performance of Speaker Independent Language Identification System Under Various Noise Environments," pp. 315–320.

[69] Y. K. Muthusamy, K. M. Berkling, T. Arai, R. A. Cole, and E. Barnard, "A comparison of approaches to automatic language identification using telephone speech," in *Third European Conference on Speech Communication and Technology*, 1993, vol. 1, no. January, pp. 1307–1310.

[70] J. Manikandan, B. Venkataramani, V. Amudha, A. M. Arafat, and H. Sahu, "A Novel Technique for Support Vector Machine based Multi-class Classifier," pp. 3–8.

[71] I. I. Journal, D. S. Processing, and I. Llc, "The International Congress for global Science and Technology," no. 9, 2009.

[72] S. P.-L. Vandendorpe, "The M2VTS Multimodal Face Database," in *First International Conference on Audio-and Video-based Biometric Person Authentication*, 1997, pp. 403–409.

[73] R. C. Rose and D. A. Reynolds, "TEXT INDEPENDENT SPEAKER IDENTIFICATION USING AUTOMATIC ACOUSTIC SEGMENTATION," pp. 293–296, 1990.

[74] D. A. Reynolds, "COMPARISON OF BACKGROUND NORMALIZATIONMETHODS FOR TEXT-INDEPENDENT SPEAKER VERIFICATION," no. 4, 1998.

[75] L. Muda, M. Begam, and I. Elamvazuthi, "Voice Recognition Algorithms using Mel Frequency Cepstral Coefficient (MFCC) and Dynamic Time Warping (DTW) Techniques," vol. 2, no. 3, pp. 138–143, 2010.

[76] C. Ittichaichareon, S. Suksri, and T. Yingthawornsuk, "Speech Recognition using MFCC," *Int. Conf. Comput. Graph. Simul. Model.*, pp. 135–138, 2012.

[77] S. Dhingra, G. Nijhawan, and P. Pandit, "Isolated speech recognition using MFCC and

DTW," *Int. J. Adv. ...*, vol. 2, no. 8, pp. 4085–4092, 2013.

[78] M. A. Hossan, S. Memon, and M. A. Gregory, "A novel approach for MFCC feature extraction," *4th Int. Conf. Signal Process. Commun. Syst. ICSPCS'2010 - Proc.*, no. January, 2010.

[79] O. Chia Ai, M. Hariharan, S. Yaacob, and L. Sin Chee, "Classification of speech dysfluencies with MFCC and LPCC features," *Expert Syst. Appl.*, vol. 39, no. 2, pp. 2157–2165, 2012.

[80] T. T. Le, J. Watton, and D. T. Pham, "Fault classification of fluid power systems using a dynamics feature extraction technique and neural networks," *Proc. Inst. Mech. Eng. Part I J. Syst. Control Eng.*, vol. 212, no. 2, pp. 87–97, 1998.

[81] A. K. Paul, D. Das, and M. M. Kamal, "Bangla Speech Recognition System Using LPC and ANN," in *Seventh International Conference on Advances in Pattern Recognition*, 2009, pp. 171–174.

[82] H. Wang, C.-C. Leung, T. Lee, B. Ma, and H. Li, "Shifted-Delta MLP Features for Spoken Language Recognition," *IEEE Signal Process. Lett.*, vol. 20, no. 1, pp. 15–18, Jan. 2013.

[83] B. Jiang, Y. Song, S. Wei, M. G. Wang, I. McLoughlin, and L. R. Dai, "Performance evaluation of deep bottleneck features for spoken language identification," *Proc. 9th Int. Symp. Chinese Spok. Lang. Process. ISCSLP 2014*, vol. 9, no. 7, pp. 143–147, 2014.

[84] W.-Q. Zhang, L. He, Y. Deng, J. Liu, and M. T. Johnson, "Time Frequency Cepstral Features and Heteroscedastic Linear Discriminant Analysis for Language Recognition," *IEEE Trans. Audio. Speech. Lang. Processing*, vol. 19, no. 2, pp. 266–276, 2011.

[85] W. Yang, R. Yu, W. Jiang, and H. Shu, "Efficient Implementation of Gaussian Mixture Models Using Vote Count Circuit," vol. 2, no. 2, 2014.

[86] J.-L. Gauvain and Chin-Hui Lee, "Maximum a posteriori estimation for multivariate Gaussian mixture observations of Markov chains," *IEEE Trans. Speech Audio Process.*, vol. 2, no. 2, pp. 291–298, Apr. 1994.

.

.