JIMMA UNIVERSITY

JIMMA INSTITUTE OF TECHNOLOGY

SCHOOL OF POST GRADUATE STUDIES

FACULTY OF ELECTRICAL AND COMPUTER ENGINEERING

Voice Biometric Based Forensic Speaker Identification Using Machine Learning

BY SILESHI AWEKE GEBEYEHU
(sileshi120@gmail.com)

A Thesis Submitted to School of Graduate Studies of Jimma University in Partial Fulfillment of the Requirements for the Degree of Master of Science in Computer Engineering

November 2021
Jimma, Ethiopia

JIMMA UNIVERSITY

JIMMA INSTITUTE OF TECHNOLOGY

SCHOOL OF POST GRADUATE STUDIES

FACULTY OF ELECTRICAL AND COMPUTER ENGINEERING

Voice Biometric Based Forensic Speaker Identification Using Machine Learning

BY SILESHI AWEKE GEBEYEHU

Advisor: Dr. Kinde Anlay(Assistant Professor)

Co-Advisor: Mr. Fetulhak Abdurahman(MSc.)

November 2021

Jimma, Ethiopia

Declaration

I, the main researcher, declare that this thesis work entitled, Voice Biometric Based Forensic Speaker Identification Using Machine Learning is my original work and has not been done and presented for any degree in any university, and all sources and references used for this thesis work have been cited and acknowledged.

Researcher: Sileshi Aweke Signature:__________ Date:____________

This thesis has been submitted for examination with our approval as the university advisers.

Main Advisor:Dr.Kinde Anlay(Assistant Professor) Signature:__________ Date:____________

Co-advisor: Mr. Fetulhak Abdurhman(MSc.) Signature:__________ Date:____________

The thesis had been examined by:

External Examiner: Dr. Million Meshesha Signature: Million Date: Nov. 03, 2021

Internal Examiner: Dr. Getachew Mamo Signature: __________ Date:____________

Chair-person: Mr. Kokeb Dese (MSc.) Signature: __________ Date:____________

Acknowledgment

First and foremost, I would like to thank the **Almighty God** with his mother **Virgin Mary** for offering us a potential to carried out this thesis work collaboratively with my advisers.

Next, my deepest gratitude goes to my main adviser **Dr. Kinde Anlay** for his supervision, guidance, valuable comments, constructive suggestions, encouragement and continuous follow-up throughout the entire duration of this thesis work.

Next, my cordial gratitude goes to my co-adviser **Mr. Fetulhak Abdurahman (MSc.)** for his ideas, assistance, constructive suggestions and innumerable supports.

Next, I gratefully acknowledge the **20 volunteers** for sacrificing their precious time for the achievement of the corpora recording process. They had a lion share for the accomplishment of this thesis work.

Next, I really would like to praise all the **former researchers** those who had been conducted their study on our research area and offering us a versatile clue on how to proceed to every step of our study and able to accomplish this task.

Also, I would like to thank Jimma University and Faculty of Electrical and Computer Engineering for supporting my study, both professionally and financially. Thanks to all my instructors and colleagues for your decisive assists in many ways.

Finally, I would like to express my special honor and deepest cordial gratitude to **my family** for their versatile involvement, support, affection, encouragement, and motivation in every single pace of my life.

Contents

List of Figures

List of Tables

List of Abbreviations

ll

ANN	Artifitial Neural Network
BFCC	Bark Frequency Cepstral Cefficent
BPNN	Back Propagation Neural Network
DCT	Discrete Cosine Transform
DET	Detection Error Trade-off
DNN	Deep Neural Network
BPNN	Discrete Time Fourier Transform
DTW	Dynamic Time Warpping
EER	Equal Error Rate
EM	Expectation Maximization
FFT	Fast Fourier Transform
FIR	Finite Impulse Response
FSR	Forensic Speaker Recognition
GB	Gigabyte
GFCC	Gammaton Frequency Cepstral Coeffiecent
GMM	Gaussian Mixture Model
HMM	Hidden Markov Model
IDR	Identification Rate
LPCC	Linear Predictive CepstralCoeffiecent
MATLAB	Matrix Labratory
MFCC	Mel Frequency Cepstral Coeffiecent
ML	Machine Learning
MLL	Maximum Log-Likelihood
PDF	Probability Density Function
RNN	Recurrent Neural Network
SR	Speaker Recognition
STFT	Short-Term Fourier Transform
SVM	Support Vector Machine
VAD	Voice Activity Detection
VQ	Vector Quantization

Abstract

Following the progress of general-purpose speaker recognition technology, specific application oriented systems are emerging based on voice bio-metric. Forensic speaker recognition is one core application area of speaker recognition. The chief application of forensic speaker recognition is identifying the actual criminal among handed suspects relying upon traced voice evidence. This thesis work aims to adopt a text-independent speaker identification for forensic speaker recognition, and examine the impact of training and testing speech corpora levels of utterance on proofing of identity of the actual criminal among handed suspects.

The proposed system is designed relying upon two indispensable and consecutive approaches, so-called front-end feature extraction and back-end feature classification. The front-end approach was employed for speaker-specific feature extraction purposes, and it had been done using a digital signal processing background, specifically using a Mel-frequency Cepstral Coefficient (MFCC). The back-end approach is employed for feature classification (suspected criminal speaker modeling and actual criminal identification) tasks. A Machine Learning (ML) based Gaussian Mixture Model (GMM) state- of-the-art with Expectation-Maximization (EM) algorithm used to build a reference model for each suspected criminal speaker and the Maximum Log-Likelihood (MLL) score technique was employed for actual criminal identification. Also, to enhance the quality of the speech corpora, and minimize computational complexity from the feature extraction and feature classification stages, a preprocessing techniques (spectral noise gate based background noise removal and short-time energy based VAD silence truncation) has been used before the feature extraction stage.

To evaluate the performance of the proposed system, we have carried out a simulation-based implementation using Python programming on the PyCharm environment. A self-collected and prepared Amharic language speech corpora used for implementation. The experimental evaluation of the proposed system is conducted on 20 speakers (who performed on the behave of suspects) recorded from ongoing mobile phone conversation at the callee side using a smartphone, and an interview room using a recorder microphone in the form of a rehearsal reading speech. The system trained and tested using the speech corpora at three levels of utterance (word, sentence and paragraph). The system achieved 84.29%, 95.00% and 97.50% respective IDRs for WLU, SLU, and PLU of mobile phone recorded speech corpora and also 85.00%, 96.25%, and 97.50% for microphone recorded speech corpora. From this study observation, apart from selecting fitting feature extraction and modeling approaches, the level of utterance of a corpora also a significant role in determining the recognition performance, and a corpora with longer level of utterances is more convenient to attain a better performance. However,the proposed system poorly performed for crossed levels of utterances and multi-modal recording training-testing scenarios yet. Hence, improving these poor performances can be the next research direction of this study.

Key Words: EM, Feature Extraction, FSR, GMM, Level of Utterance, MFCC, MLL score, Speech Corpora, Suspects.

CHAPTER 1

INTRODUCTION

1.1 Background of the study

To have an integrated and cooperative interaction in humans' day-to-day life, body language, textual language, pictorial language and speech are the most common communication mechanisms. However, because of its content richness, speech is regarded as the most powerful mechanism to express thoughts and information. The primary role of a speech is delivering a message. Besides to that, during conversation, it can offer a number of speaker specific information such as gender, age category, attitude, emotion, health condition and identity of a given speaker. Having such plentiful information contents inspired the researchers to decode speech signal related information for practical applications; such as language, speech and speaker recognition. And this led for emerging systems that able to procure and process the assembled information of a speech signal[1].

Biometrics is the measurement and statistical analysis of people's unique physical and behavioral characteristics. Fingerprint, face, iris, and voice are some well-known sorts of biometrics. Biometrics Recognition technologies refer to the use of technology to recognize an individual based on some aspect of his or her biological attribute. Voice biometrics is the technology of using a person's voice as a unique biological characteristic to recognize the speaker identity[a]. The human voice is as unique as a fingerprint, which is the reason why voice bio-metrics technology is used for speaker identification purpose in various industries, ranging from banks' call center to crime investigation agencies to identify speakers reliably based on their voice.

Speaker recognition (also known as voiceprint recognition) is the task of recognizing an identity of a person relying on his or her voice traits[2]. It used to answer the question "Who is speaking?"[b], and in the past few years, this speaker recognition technology has gained considerable acceptance in a wide variety of practical applications. For instance, remote telephone banking, computer login, cellular telephone fraud prevention, automatic speaker labeling of recorded meetings for speaker-dependent indexing, intelligent answering machines with personalized caller greeting and forensics analysis (the concern of our study) are some of the potential applications of it.

As general, speaker recognition is the task of recognizing an individual relying on his or her voice traits. Specifically, the task of determining if a suspected is the source of a questioned voice trace evidence, it is known as forensic speaker recognition (FSR)[3]. FSR

[a] Voice Biometrics, accessed on August, 30, 2021

[b] Speaker recognition, accessed on March, 23, 2020

is an established term used when a speaker recognition techniques are adopted for forensic applications. It is the comparison of recordings of an unknown voice trace evidence with a known reference voice set of the suspects[4].

Usually in actual forensic incidents, there might be a mismatch between the training and testing scenarios. As described in [5], for automatic, semi-automatic and human-based techniques, recognizing a speaker using forensic-quality samples is typically a challenging task. The speech samples being compared might be recorded in different situations. For instance, the questionable trace evidence most probably obtained from an intercepted or recorded phone call of the unknown criminal while the training corpora are usually recorded from the suspects in the police department, one sample might be yelling over the telephone whereas the other might be a whisper in an interview room, a speaker could be disguising his or her voice, ill or being under the influence of drugs, alcohol, stress, etc. Also, the speech samples may be noisy, may be very short, and may not contain enough relevant speech signal information for comparative purposes. And any of these mismatch scenarios can be a challenging hinder in designing a robust FSR system.

Machine learning describes a set of techniques that are commonly used to solve a variety of real-world problems with the help of computer systems which can learn to solve a problem instead of being explicitly programmed[6]. Vector Quantization (VQ), Gaussian Mixture Models (GMMs), Hidden Markov Model (HMM), Artificial Neural Network (ANN) and Deep Neural Network (DNN) are some commonly used classification approach of speaker recognition. The first four approaches considered as classical machine learning techniques while the last two considered as advanced machine learning techniques. As former related works proved that advanced machine learning techniques are cost intensive compared to the classical techniques (the detail presented under section 2.3).

1.2 Motivation of the study

Two things were motivated us to come up with this thesis idea. The first thing was that the style of these days' crimes. Nowadays, committing voice-oriented crimes remotely through telecommunication and voice-oriented social media becoming a suitable tactic for criminals. On the other hand, from the law enforcement and justice entities perspective, remotely committing crimes are too tough to gather evidences and identify the actual criminal among handed suspects via the usual fingerprint or/and eyewitness approaches. The second thing was that the status of forensic speaker recognition (FSR) research works for low resourced languages. To the best of my (the principal researcher) survey, hitherto a number of FSR studies have been conducted for high resourced languages, such as English, Japanese, Chinese, Arabic,etc. But, it is too far behind for low resourced African languages. For instance, there is no such a work has been done for Amharic language yet.

1.3 Statement of the Problem

Imagine a scenario where there is a suspect being charged in court, and he denies saying something. The prosecutions brings a recording, saying they have his confession on tape. As the accused vigorously denies the voice being his, an expert shows just why the voice could be no one else's[c].

Following the drastic expansion of the electronics world, the communication technology is getting more convenient from time to time to offer versatile services for the users in every aspect of the day-to-day life. And the accessibility of such a handy technology in various forms of communication ways to the individual user, enabling to ease the human life at a significant level and numerous perspectives.

However, besides to their multipurpose utilities, the communication technologies (telephone and other various forms of social media) are exposing for misuse purposes. Illegal users are making them as proficient tools in committing criminal offenses such as obscene calls, anonymous calls, harassment calls, ransom calls, terrorist calls, calling for corruption deals, calling for gender-based violence, etc[7]. To support this with a recent incident which was happened in Ethiopia, on December 20, 2019, unidentified bandits kidnapped eight children nearby Gondor, Amhara regional state and requested that their families to pay a ransom of 120, 000 Ethiopian Birr(around $3,700) each. And the families failing to afford the ransom money, the bandits executed the children a week later[d]. Also recording and disseminating hate speeches via social media is becoming another common incident of the illegal users.

To commit these crimes in such a way, criminals relying on remotely voice usage believing that they will remain incognito and no one would recognize them. Which means that, the criminals committing such crimes by taking the gaps of the usual fingerprint and/or eyewitness criminal identification mechanisms as an advantage in case of voice oriented remotely committing crimes.

To confront the problem, enhancing the criminal identification mechanism supporting through an up-to-date and appropriate technology believed to be one feasible solution. So that, a number of FSR research works have been done for high-resourced languages such as English, Arabic, Japanese, etc. But this hasn't been done that much for low resourced languages. For instance, there is no such a work has been done for Amharic language speakers yet. In fact, few speaker recognition works have been done for Amharic speakers, but they were from general purpose perspective. In FSR, speech evidence used to recognize its speaker. But this evidence might be unpredictably distorted compared to the speech sample utilized for general purpose speaker recognition. As of [5], noise, channel mismatch, multiple speakers, voice disguise, voice forgery and speech duration

[c] Why Are Human Voices Different?, accessed on July, 30, 2020

[d] Ethiopia: 6 kidnapped children killed after families fail to pay ransoms, accessed on June 17,2020

of the evidence are the major factors of FSR. This study carried out to design FSR, and examine the impact of level of utterance over recognition performance.

The research will answer the following question:

i. Does the level of an utterance of a speech corpora have an impact on FSR performance? if the replay is yes, which level of utterance corpora is convenient to accomplish a better recognition performance?

1.4 Objective

1.4.1 General Objective

The general objective of this study is to design a voice bio-metric based forensic speaker recognition system using Machine Learning.

1.4.2 Specific Objectives

To achieve the general objective of this thesis work, the specific objectives are:

i. To deeply understand the speaker recognition technology and related speech analysis techniques through proper review

ii. To collect and prepare Amharic speech corpora for training-testing implementation

iii. To investigate and select fitting approaches for front-end feature extraction and back-end feature classification operations

iv. To design a FSR system, and bridge the gaps of fingerprint and eyewitness criminal identification techniques in case of crimes with voice trace evidence

v. To examine the impact of level of utterances of a speech corpora on FSR performance

1.5 Methodology of the study

Research methodology is the specific procedures used to identify, select, process, and analyze information about a topic[e]. In a research paper, the methodology section allows the reader to critically evaluate a study's overall validity and reliability.

1.5.1 Research Design

Experimental research is a scientific approach to research, which is centrally concerned with constructing research that is high in causal validity[f]. Causal validity concerns the accuracy of statements regarding cause and effect relationships. This kind of study will produce conclusions which can be verified through experiment or observation. This thesis work follows an experimental research design with Data preparation, preprocessing,

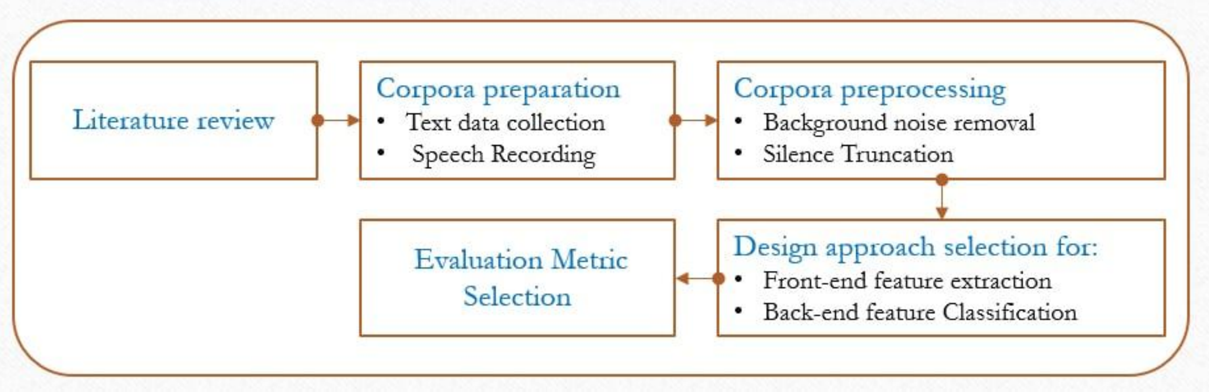


Figure 1.1: Methodology of this study

selection of feature extraction and classification techniques, and evaluation metrics are basic procedures. Figure 1.1 reveals the methodology of this study.

Literature Review: To have a deep understanding related to this study subject matter, various literature reviews were conducted on the areas of speech signal processing, front-end feature extraction and back-end feature classification techniques that are relevant for our work. During the review, available books, journals, case studies and previous research works were reviewed.

Corpora Collection and Preparation: This study conducted to design a FSR, and implement for Amharic speakers. Unfortunately, Amharic is a low resourced language[??]; it hasn't a publicly available speech corpora resource for such studies. For that matter, for this thesis work implementation, we have prepared our own corpora. The corpora preparation started from collecting randomly selected word, sentence, and paragraph-level Amharic text files from social media (Facebook and Telegram). Then, the recording is done as a reading speech from an ongoing mobile phone calling at the receiver side using smartphone, and using a microphone at the caller side. The detail of this step presented under section 4.1.

Pre-processing: Identifying a voice using forensic-quality samples is generally a challenging task for automatic, semiautomatic, and human based methods[5]. The speech samples being compared may be recorded in different situations. That means, the speech evidence from criminal incidents highly expose-able to a noise. The pre-processing step is needed to enhance the corpora quality; so that to minimize computational complexity from the subsequent front-end feature extraction and back-end feature classification steps. For this study, background noise removal and silence truncation techniques employed as pre-proceesing step.The detail of this step presented under section 4.2.

Design approach selection: Front-end feature extraction and back-end feature classification are the two indispensable and subsequent design sections in speech and speaker

[e] Research Methodology, accessed on October, 05, 2021

[f] What is Experimental Research?, accessed on October, 05, 2021

recognition related systems. In speaker recognition systems, the front-end feature extraction aims to extract a speaker-specific acoustic feature vectors that can represent the given speaker uniquely, and serve as an input for the subsequent back-end feature classification. The back-end feature classification section used to build models for classes representation during the enrollment phase, and identification during the testing phase. We employed a digital speech signal processing (in terms of cepstrum) based front-end feature extraction, and Machine learning based back-end feature classification approaches.The detail of this step presented under sections 4.3 and 4.4.

Evaluation Metric Selection: Once the proposed system is designed, during testing the effectiveness of that system measured with performance metric(s). There are many measurement metrics to evaluate performance of the proposed speaker recognition system. These metrics are slightly different for different types of speaker recognition systems. For instance Detection Cost Function(DCF), Detection Error Trade-off curve (DET) and Equal Error Rate (EER) are common evaluation metrics for speaker verification and open-set speaker identification systems. While Identification Rate(IDR) is the most widely used evaluation metric for closed-set speaker identification systems[8].The detail of this step presented under section 5.6.

1.6 Scope of the study

The scope of the proposed system is constructing a reference model for handed suspects from a pre-recorded training speech corpora, and identifying the unknown criminal speaker under closed-set mode scenario from a single speaker voice trace evidence relying upon his or her maximum log-likelihood score (for this thesis work implementation, we have collected a text file from Facebook and Telegram social media, and made a record from AN ongoing mobile phone conversation and an interview using microphone recorder. The collection and recording process has been done during this research time). The system has not been designed to replace the usual fingerprint and/or eyewitness based criminal identification mechanisms; rather, it has been proposed to bridge their gaps in case of voice-oriented crimes. Even though, a forensic problem inclined to the open set scenarios, due to lack of a defined threshold value related with intra-variation of an individual's voice, the proposed system took into consideration only for training and testing in a closed set mode scenario of criminal identification from the given suspects. Also, the issue of twines, and the act of impersonation has not been deliberated.

1.7 Limitation of the study

- Eve if there are many local languages, due time and many constraint to collect and and prepare corpus, the system implemented using single language (the training-testing language dependency has not been checked).

- Because of time constraint, the training-testing corpus collected with short time interval (voice intra-variation due to time mismatch has not been examined)

- The suspects trained and tested in their normal condition (health mismatches have not been examined)

1.8 Significance of the study

As described earlier, crimes that are committing via telephone and voice-oriented social media are remote incidents by nature; so that, they are inconvenient to have evidence through the usual ways. Thus, in such cases, technically the proposed system used to bridge the gaps of fingerprint and eyewitness oriented criminal identification mechanisms through offering an assistance for law enforcement entities (polices, detectives (forensic experts), prosecutors and Judges) and institutions (police stations, courts, etc.) in case of voice-oriented crimes. From the research state-of-art perspective, the prepared corpora set can be used as an open source for junior researchers who are interested on speaker-independence speech recognition, speaker diarization, speaker verification and language recognition study areas. Besides, the result section used to reveal the level of utterance preference of a corpora in developing such related system with a better recognition performance.

1.9 Document Organization of the study

The rest section of this thesis work outlined in the following manner: In chapter two under literature review overviews of speaker recognition, forensic speaker recognition and related works are presented. In the subsequent chapter, under speech and speech signal processing, a brief insight on the speech signal, human speech production mechanism and its uniqueness, and the theory of speech signal processing that are the foundation of extracting speech features, such as frequency analysis, short-term processing, and cepstrum are provided. Chapter four presented the design methodology of this thesis work. The design methodology major steps: speech corpora preparation, the speech signal preprocessing, front-end feature extraction, back-end feature classification are presented in detail with their respective subsections. The implementation section of the study is presented in chapter five. Chapter six presented the thesis experimental results with their corresponding discussions. Chapter seven is the final section of our thesis work and it comprises of the conclusion and recommendation.

CHAPTER 2

LITERATURE REVIEW

To offer an overall insight regarding the basics of speaker recognition and related issues, in this chapter three sections with their respective subsections are presented: overview of speaker recognition, forensic speaker recognition and related works.

2.1 Overview of Speaker Recognition

2.1.1 Speaker Recognition

The most natural way we humans interact with each other is through spoken language. Rich information, including language information (linguistic content, accent, etc.), speaker information (identity, emotion, physiological characteristics, etc.) and environmental information (background, channel, etc.) are conveyed in spoken language. Although this information encoded in a complex form, we humans can effortlessly decode them through our auditory system. And this decoding capability of the human auditory system has inspired researchers to automatically extract and process the information in spoken languages[8].

Hitherto, several studies have been carried out from different perspectives of the spoken languages; such as accent, language, expression, feeling, gender and speaker recognition. Among these, speaker recognition used a set of observable features of a human voice that can symbolize and recognize a person uniquely. As mentioned earlier, this speaker recognition refers to the task of recognizing (verifying or identifying) a person based on his or her voice traits. The studies conducted on speaker recognition showed that a person's voice-print has the Properties of individuality and continuity, which can remain relatively stable and are not easy to alter[8]. In addition, researchers still consent that owing to physical (anatomical, vocal tract dissimilarities) and behavioral (learned) features, no two human sounds are alike.

2.1.2 Categories of Speaker Recognition

Relying upon different viewpoints, speaker recognition can be categorized into the following categories[8]:

2.1.2.1 Identification Vs Verification

Essentially, the task of speaker recognition can be categorized into two as verification Vs identification[2]. Speaker verification or authentication is the technique of determining whether the speaker's identity is who the person claims to be. It performs a one-to-one matching (it is also called binary decision) between the features of an input voice and those of the claimed voice that is registered in the system. Figure 2.1 reveals the fundamental structure of the speaker verification system. Front-end processing, speaker modeling and pattern matching are the three main components as seen in the structure. To get the feature vectors of the input voice, front-end processing will be performed, and then depending on the match scores would be determined for templates used in pattern matching. If the match is above a certain threshold, the identity claim would be verified, else it would be rejected. Using a high threshold, the system gets high safety and prevents impostors to be accepted, but in the meanwhile, it also takes the risk of rejecting the genuine person, and vice versa [2].

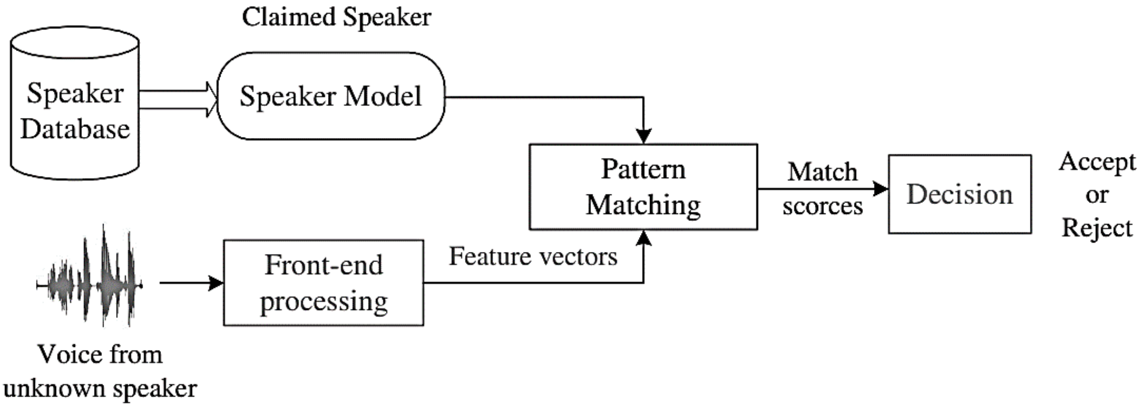


Figure 2.1: Basic structure of Speaker Verification[2]

On the other hand, speaker identification is the task of finding the identity of an unknown speaker by comparing his or her voice with the voices of registered speakers in the database. Unlike that of speaker verification, it is a one-to- many comparison [3].

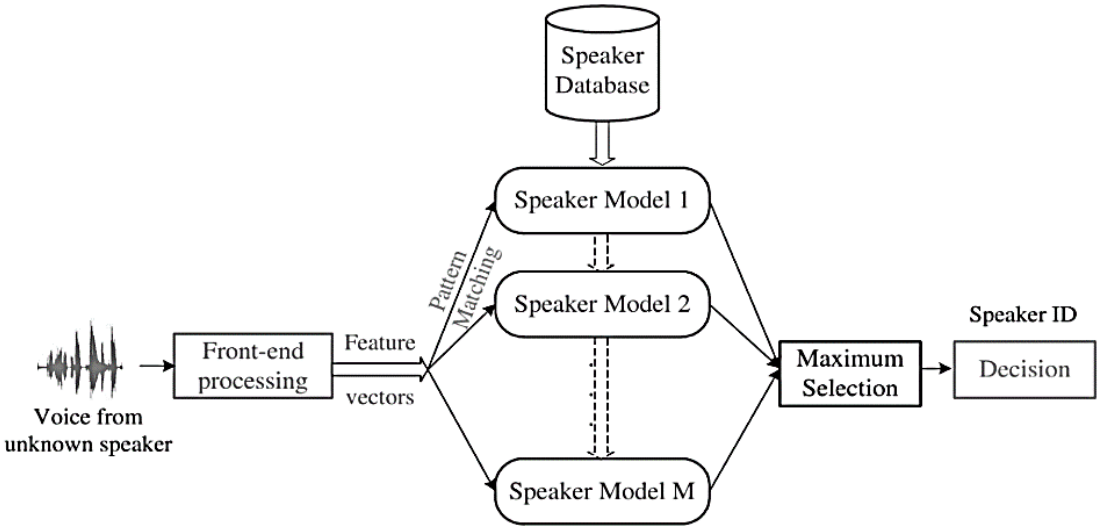


Figure 2.2: Basic structure of Speaker Identification[2]

Figure 2.2 reveals the basic structure of the speaker identification system, and as we can notice from the structure, the core components in speaker identification are the same as in speaker verification system. In speaker identification, S number of speakers' models are scored in parallel and the most-likely one is reported, and consequently, the decision will be one of the speaker's ID in the database for closed-set speaker identification system, or will be "none of the above" for open-set speaker identification system. Usually, for speaker identification, there are two operating modes[8]: open-set and close-set. The system assumes in the close-set mode that the unknown speech samples must come from one of the enrolled speaker sets. And for each enrolled speaker, at the time of assessment, a matching score is calculated and the speaker corresponding to the model with the highest matching score is chosen, as an identified speaker. The speaker can be within or outside the set of enrolled speakers in the open-set mode, and anyone who is not enrolled should be rejected.

Both speaker verification and identification have their own practical applications. For instance telephone banking, computer login, cellular telephone fraud prevention, etc. are potential applications of speaker verification. While automatic speaker labeling of recorded meetings for speaker- dependent audio indexing, intelligent answering machines with personalized caller greetings, forensic analysis, etc. are potential applications of speaker identification.

2.1.2.2 Text-Dependent Vs Text-Independent

Based on the text modality, the speaker recognition systems also can be categorized as text-dependent Vs text-independent. In the text-dependent systems, the speaker is expected to say or utter the same text file for both training and testing phases. Due to the prior knowledge (lexical content) of the spoken phrase, these text-dependent systems are more robust and can achieve good performance. However, there are cases when such constraints can be cumbersome or impossible to enforce. In the text-independent systems, there are no constraints on the text. Thus, the enrollment and testing utterances may have completely different texts. In the text-independent case, the system must model the general underlying properties of the speaker's vocal spectrum. Unfortunately, since the content information is not used, there exists a distribution mismatch between enrollment and test due to the text variations, which leads to performance degradation.

Text-dependent speaker recognition systems are used mostly in services such as access control and telephone-based services, where the speakers are considered to be cooperative. Text-independent speaker recognition systems are the most flexible, and widely used in events where speakers can be considered non-cooperative users, as they do not specifically wish to be recognized such as forensic analysis and surveillance cases[9].

2.1.2.3 Close-Set Vs Open-Set

Furthermore, it is possible to categorize speaker recognition systems as Close-set Vs Open-set operating modes[8]. In case of close-set mode, the system assumes that the unknown speech samples must come from one of the enrolled speaker set. And for each enrolled speaker, a matching score is calculated during training, and at the time of testing, the speaker corresponding to the model with the highest matching score is chosen as a recognized speaker. On the other hand, in case of open set mode, the source speaker of the unknown utterance might be within or outside the set of enrolled speakers, and anyone who is not enrolled should be rejected. In the testing process, the feature vectors extracted from the unknown speaker are compared against the reference models; and the decision to accept or reject a speaker depends on the threshold value.

For this thesis work, we adopted a text-independent speaker identification under a closed-set mode for forensic speaker recognition application.

2.1.3 Phases of Speaker Recognition

Speaker recognition is a typical pattern recognition task; which is composed of two consecutive phases, namely the training phase and the recognition phase. Figure 2.3 reveals the fundamental framework of the speaker recognition system[8].

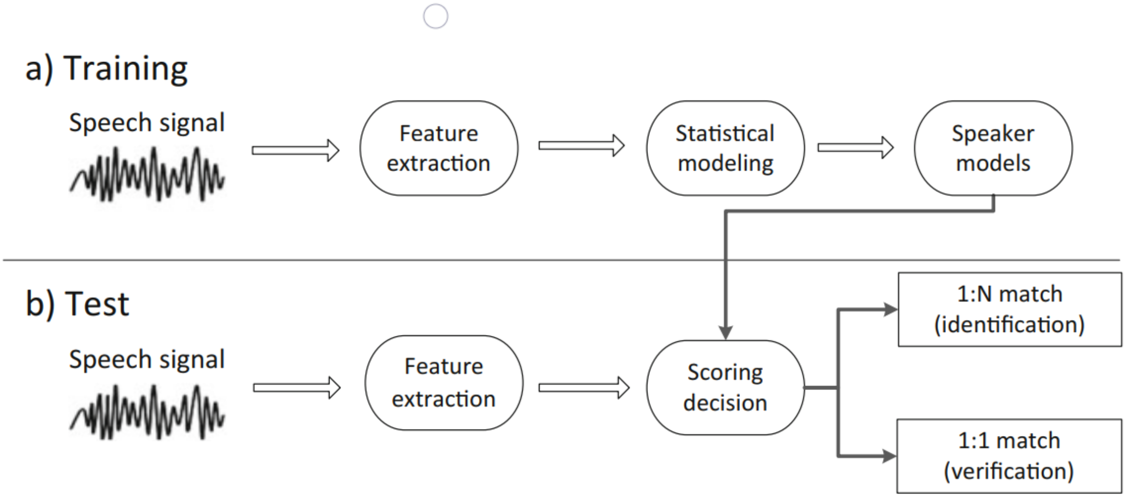


Figure 2.3: System framework of speaker recognition[8]

The training phase, reveals in Figure 2.3a, also referred to as registration or enrollment phase, in which a speaker enrolls by offering voice samples to the system. It consists of two steps basic design elements. The first element is to extract features from the speech signal while the second element is to obtain a statistical model from the extracted features, referred to as speakers' models. The testing phase, as shown in Figure 2.3b, is also called classification or recognition(verification or identification) phase, in which a test voice sample is used by the system to measure the similarity between the user's voice and each of the previously enrolled speaker models to make a decision. In a speaker identification task, the system measures the similarity between the test sample and every stored speaker's model, while in a speaker verification task, the similarity is measured only with the model of the claimed identity.

2.1.4 Progresses of Speaker Recognition Technology

As of the existing works indication, The idea of speaker recognition technology has been studying for the last sixty to seventy years, starting early 1960s. In this section, we offered a brief overview of its overall progress.

The problem of recognizing an individual by his or her voice is an age-old issue. Genesis records Isaac's dilemma in verifying a speaker when Jacob acts as an impostor of his brother Esau. Isaac's confusion was with two contradictory bio-metrics. "The voice is Jacob's voice, but the hands are the hands of Esau." Jacob trusted tactility over auditory "and he discerned him not." (Gen. 27:22- 23). The speaker recognition problem appears in a judicial case as early as 1660. A triple of centuries later, academic research would begin investigating voice bio-metric[10].

The Charles and Anne Lindbergh's baby boy was kidnapped in 1932 and later executed. In a cemetery where a Lindbergh agent met with an unidentified man pretending to be the kidnapper, the investigation led to a covert reward. In a nearby car, Charles and Lindbergh sat down. The anonymous man was overheard by Lindbergh saying "Hey Doctor, over here, over here". Two and a half years later, at the trial of Bruno Hauptmann, the suspected kidnapper, Lindbergh believed that he could recognize the voice of Hauptmann as the same voice heard in the cemetery[10]. Frances McGehee was inspired by the Lindbergh case to conduct the first documented study on earwitness reliability[11]. And the later development of the autonomous speaker recognition system has its roots in the work of McGehee.

Well organized attempts towards speaker recognition were began in the 1960s. In 1962, the Bell Lab Physicist L. Kersta published an article entitled "Voiceprint Identification" in Nature Magazine[12]. Simultaneously, S.Pruzansky from Bell Labs[13],investigated systems for speaker recognition using spectral pattern matching by. The achievement of this study drew broad attention from scientists in the field of signal processing and the extension of speaker recognition science was put in motion.These early successful trials were all text-dependent. At the time, investigations into text-independent techniques had no such promising results.

Text-independent researches made a significant advance in 1969 when James Luck proposed that the cepstrum be applied to recognize speakers[14]. The cepstral analysis would become the predominant method for obtaining measurable traits in a person's voice. However, it took some time before Luck's concept of cepstrum based speaker recognition became widely used. The results of a study published by Atal[15] demonstrated an improvement in the identification accuracy of the cepstral approach over other approaches.

In the 1980s, studies on speaker recognition were concentrated on acoustic feature extraction. The use of cepstral coefficients and their orthogonal polynomial coefficients was presented by Furui on in a frame-based method[16]. The system was tested extensively and successfully. The success of his work sparked a renewed research effort in the use of the cepstrum approach. In order to separate the vocal tract envelope from the glottal excitation component of a speech, this technique uses the deconvolution capability of the cepstrum. This ability to interpret the deconvoluted voice signal makes Cepstral analysis a valuable instrument,and dominant approach. Subsequently, number of cepstral based acoustic feature extraction approaches, such as Linear Predictive Cepstral Coefficient(LPCC), Bark Frequency Cepstral Coefficient (BFCC), Gamaton Frequency Cepstral Coefficient(GFCC), Mel-Frequency Cepstral Coefficient (MFCC)[17],etc. had been employed for speaker recognition studies.

Besides, several modeling and decision-making Machine Learning algorithm approaches, such as Vector Quantization (VQ)[18], Hidden Markov Model (HMM)[19], and Artificial Neural Network (ANN)[20] had employed as back-end classification along with the cepsral front-end feature extraction approaches, and made significant improvements in the speaker recognition studies.

Since the 1990s, especially after the detailed introduction of the Gaussian Mixture Model (GMM) by Reynolds[21], GMM had rapidly become a mainstream model for text-independent speaker recognition, due to its advantage of flexibility, high efficiency, and good robustness. In 2000, Reynolds[21] brought up the Gaussian Mixture Model-Universal background model (GMM-UBM), which had made a great contribution to making the speaker recognition technology from lab experiments to practical use. P. Kenny, N. Dehak, and other researchers proposed the Joint Factor Analysis (JFA)[22] and the i-vector[23],[24] models.

Starting in 2010, inspired by the success of deep neural networks (DNNs) in speech recognition[25], DNN and its recurrent variant (recurrent neural networks, RNN) had been applied in the speaker recognition and achieved promising results. Based on these models, many exciting structures were designed for deep feature learning or deep speaker embedding. For instance, in[26], DNNs trained for automatic speech recognition were used to replace the UBM model to derive the acoustic statistics for i-vector models, and this phonetic-aware method could provide more accurate frame posteriors for statistical computation.

Recently, Tang and Li[27] presented a collaborative learning structure based on long short-term memory models. The main idea was to merge task-specific neural models with inter-task recurrent connections into a unified model. This model fits well with the joint training of speech and speaker recognition. In this scenario, the speech content and speaker identity were produced at each frame step by the speech and speaker components, respectively. By exchanging these bits of information, performances of both speech and speaker recognition were sought to be improved.

As we observed from the given evolution overview, the speaker recognition systems have been progressing continuously for shifting from lab scale to real-time practical applications, and nowadays specific application-oriented studies are becoming the common concerns of the researchers.

2.1.5 Application areas of Speaker Recognition

As mentioned earlier in the background section, several real time applications had been considered for speaker recognition. Such as secure access control by voice, customizing services to an individual by voice, indexing speakers in recorded conversations, surveillance and forensic investigations, etc. The following brief discussion regarding application areas of speaker recognition is according to[28], which was specifically studied on the application of speaker recognition, and discussed in a better way.

Speaker Recognition for Authentication: Different features such as signature, fingerprint, voiceprint, facial, etc. may distinguish an individual. And this kind of authentication is known as biometric authentication. In such method, the chance of misuse of these types of identity problems is lesser as compared to the key or credit card, can be stolen or lost, followed by personal identification number or password can be easily misused or forgotten. Each individual has a unique anatomy, physiology and learned habits which families use to identify the person in everyday life. This can be much more convenient than conventional authentication methods that require a key to be carried or a password to remember.

Speaker Recognition for Surveillance: Intelligence agencies have different information collection processes. One of these is the electronic eavesdropping of telephone and radio conversations. As these results might be obtained from vast volumes of data, filtering mechanisms must be applied to find the most necessary data. And one of these filters may be the recognition of the targeted speakers that are of interest to the service.

Speaker Recognition for Security: It is the most obvious application of any biometric authentication techniques. In credit card purchases, speaker recognition may be used as an authenticating process in conjunction with others such as face recognition. Speaker recognition technologies can provide the facility or control of computer access, monitoring, long-distance voice authentication, banking access, etc.

Speaker Recognition for Forensic Analysis: If there is evidence of a voice trace (speech sample) that was captured during a certain crime by an unknown criminal, the voice of the suspect may be compared to this to suggest the two voices in particular. Proving a captured voice can help to convict a suspect or release an innocent person. Although automatic speaker recognition systems cannot perform this role entirely, they may facilitate the actual crime identification of suspected offenders of forensic experts. And as already mentioned, our thesis work is concerned with this particular application field.

2.2 Forensic Speaker Recognition

Judges, prosecutors, detectives and law enforcement bodies have been able for several years to make use of forensic voice authentication to prosecute a suspect or to validate the conviction of innocence[5].

Speaker recognition is the general term used to discerning people based on their voices. In particular, forensic speaker recognition (FSR) is the task of determining if a suspected speaker is the source of a questioned recording (voice trace evidence). This process involves the comparison of recordings of an unknown voice (questioned recording) with a known voice set of the suspects. In a technical expression, during FSR, the statistical models of acoustic features of the suspected criminal speakers' voices and the acoustic features of the questioned recordings are compared[9].

Biometrics is the science of establishing an identity of an individual based on his or her biological and behavioral characteristics[29]. On the other hand, forensic analysis refers to the deployment of scientific principles and technical methods to the investigation of criminal activities; i.e to demonstrate the existence of a crime and to determine the identity of its doer(s)[30]. Forensic speaker recognition uses science and technology in the investigation and establishment of facts or evidence in the court of law. Figure 2.4 depicts processing chain for calculating bio-metric speech evidence for forensic speaker recognition[4].

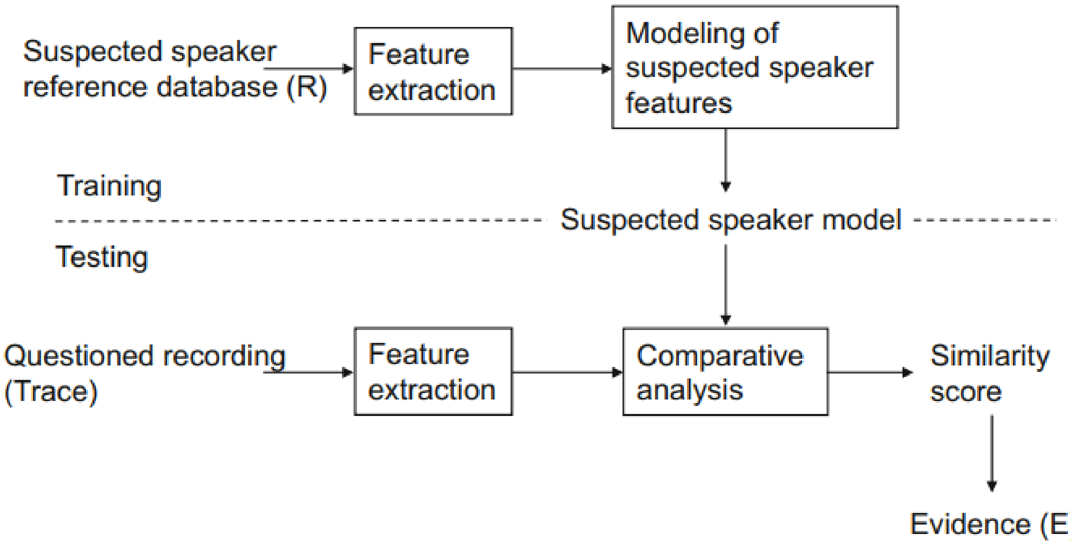


Figure 2.4: Processing chain for calculating bio-metric speech evidence[4]

Identifying a voice using forensic-quality samples is generally a challenging task for automatic, semiautomatic, and human based methods[5]. The speech samples being compared may be recorded in different situations; e.g., one sample could be a yelling over the telephone, whereas the other might be a whisper in an interview room. A speaker could be disguising his or her voice, ill, or under the influence of drugs, alcohol, or stress in one or more of the samples. That means, the speech samples will most likely contain noise, may be very short, and may not contain enough relevant speech material for comparative purposes. Each of these variables, in addition to the known variability of speech in general, makes reliable discrimination of speakers a complicated and daunting task.

As stated in section 2.1.2.1, speaker recognition can be classified into two major application tasks as speaker verification versus speaker identification. Previously, the usage of speaker verification for forensic speaker recognition has been common. But, recently an investigation concerning the inference of identity in forensic speaker recognition has shown the inadequacy of the speaker verification, and speaker identification techniques for forensic application[4][31].

2.3 Machine Learning Classification Approaches

Once the sequence of feature vectors are extracted from the speech signal of a given speaker, the role of a speaker recognition system is to check whether that feature vector belongs to one of the registered speakers. To do so, speaker modeling is responsible in generating reference model(s) for each registered speaker during the enrolment phase depending on the features extracted from the speech signal. Then, during the recognition phase, the test utterance from an unknown speaker is compared with a reference model of the claimed speaker, in case of speaker verification or with all speaker models in case of speaker identification to get the matching score, which indicates the degree of matching.

In general, these approaches can be classified into generative and discriminative models. Generative models try to capture the whole underlying distribution, i.e., the class mean and variation around that mean, of training data. In addition, this model is trained to represent the entire distribution space of the training data generated from a particular class. The trained class model considers only matching data, discarding the distribution of the other classes [32]. The generative models can also be classified into the template models and stochastic models. Discriminative models, on the other hand, do not need to model the entire distribution, but only the most discriminative areas of distributions. The aim of training a discriminative model is to minimize the classification errors for a set of training samples. Consequently, not only samples from the matching class but also those from all the rival classes are taken into account when training the discriminative model for each class. These models include Support Vector Machine (SVM) and Artificial Neural Network (ANN).

The models generated in a discriminative model require the training sample from both the target speaker and all the imposter speakers, and modeling only the boundary may cause discarding of some information from the client which may contain the boundary information between the target speaker and other imposters. As a result, the discriminative model may work poorly for these imposters. However, in Generative models, all the target information is retained, which makes the generative models more robust against these imposters. Furthermore, in discriminative models, if the reference models in the speaker recognition system are updated with some new speaker, all the reference models have to be retrained. The generative models, on the other hand, do not require re-training because each target model is trained independently [32]. This makes the generative model more appropriate for speaker recognition compared with discriminative models.

i. Vector Quantization (VQ)

VQ is an earliest template classification model for speaker recognition ([33],[34]. This technique, also known as a centroid model, includes separating the features vectors into a set of non-overlapping clusters which individually represent different acoustic classes. Each of these clusters is represented by a code vector which is the centroid of that cluster (mean vector). In this approach, a collection of centroid vectors represents a speaker reference model which is referred to as a codebook [35]. VQ approaches offer an efficient way of decreasing data storage requirements while preserving the fundamental aspect of the original distribution [33]. During the classification phase, the distance of each of the extracted features vectors of the recognition utterance to its closest codebook vector is accumulated to produce an utterance scores[35].

ii. Support Vector Machine (SVM)

SVM is one of the discriminative binary classifiers adopted in speaker verification. It includes modeling the linear boundary between two classes as a separating hyper plane [36],[37]. In speaker verification, the first class consists of target speaker enrollment vectors (labeled +1) and the second class consists of training vectors from a huge number of background features. (Labeled -1) ([9]. In addition, SVM can be used to learn non-linear boundary regions between samples by mapping the input samples into higher dimensional space. This can be done by using a kernel function [35]. Depending on these labeled training vectors, SVM is responsible for finding a splitting hyper plane that maximizes the margin of separation between two classes. In the recognition phase, a classification score is then obtained by evaluating the distance of the recognition sample in relation to the hyper plane.

iii. Hidden Markov model (HMM)

HMM is a stochastic model commonly used in speaker recognition, especially for text dependent and text-prompted speaker verification, where a whole phrase is matched. It has the ability to model the temporal variations between the various acoustic classes [35]. HMM models are first-order discrete time series with some hidden information known as states[37]. In the field of speaker recognition, each state may be referred to phones or larger units of speech. Through discrete time, the state of the HMM system is changed according to a set of probabilities related to it. The output from each current state is emitted after each transition. Although these outputs can be observed, the connected states are hidden and can only be deduced from outputs. This inf[35]. Matsui and Furui (1994) made a comparison between a vector quantization based texts–independent speaker identification system and a discrete/continuous ergodic HMMs based one. The experiments show that the continuous ergodic HMMs based system has the same robustness as the system that used VQ for variation of utterances. The authors also mentioned that the robustness of continuous ergodic HMMs based systems are restricted with the availability of sufficient data while VQ based systems show greater robustness when the amount of data is limited.

iv. Gaussian Mixture Model (GMM)

GMM represents an extension of VQ, and become the most dominant modeling approach in speaker recognition[9]. It is a stochastic approach that expresses the probability density function of a random variable in terms of a weighted calculation of the sum of Gaussian components. These components are the mean, covariance and weight associated with each of them which together represent each model of a speaker in GMM [38]. These models' parameters are generally computed by applying an iterative Expectation Maximization (EM) algorithm [39]. Reynolds and Rose (1995)[40] proposed a speaker recognition system using GMM model, and they demonstrated the strength of using this model in text-independent speaker identification compared with VQ. In the training phase, the collect utterances from the speakers pooled to train the GMM model using EM algorithms as speaker-independent models. Then, during the recognition phase, the test data are matched with register speakers' models using maximum likelihood rule.

Over the last two decades, GMM has become the most popular modeling approach in text independent speaker recognition [32], [35], [41]. It represents one of the better modeling approaches when the speech data used in enrolled and recognition phases was limited [42]. GMM has two important features: First, it has shown an efficient performance with limited dataset than other machine learning modeling approaches. Second, the speech signal is based on stochastic process, and to create an efficient model for the speakers the modeling used should contain statistical analysis. That means the Gaussian built in by GMM become more efficient for the speech signal. To take the cited advantages of the GMM model. We have chosen it for this study to design a FSR system. The mathematical issues and more details of GMM modeling presented in the design chapter, under section 4.4.1.

v. Artificial Neural Network (ANN)

ANN is a discriminative model that is widely used in speaker recognition [43], [44]. One of the main advantages of ANN is that the feature extraction and speaker modeling can be applied to a single network, enabling joint optimization of the feature extractor and the speaker model [45], [46]. This approach uses enormous parallel networks of many densely interconnected computational units known as neurons which are analogous to the neurons that exist in the human central nervous system [37]. Each neuron is responsible for sums a number of weighted inputs and passes the output through a non-linear activation function. Although there are many different kinds of ANN, the Multi-layer Perceptron (MLP) has been a commonly used architecture for speaker recognition. An MLP is made up of a network of simple neurons which are known as perceptron. The main idea of the MLP is based on a two stage process: first, compute a linear weighted sum of its input connections. And second, a non-linear activation function is applied in order to calculate the output of the neuron. In speaker verification, an MLP has only one output neuron, since the goal, in this case, is to obtain a score over all the frames of the given recognition utterance [35]. Wahab et al. (2005) on [43] used an MLP neural network and Generic Self-organizing Fuzzy Neural Network (GenSoFNN) with extracted hidden features as an input for this network for a speaker verification system. The experiment was conducted on 10 speakers consisting of 6 males and 4 females recorded in a quiet room in a Digital system lab by using a digital tape recorder. Their experimental results showed the ability of both systems to verify speakers with high accuracy. Furthermore, the authors mentioned that the MLP can achieve high accuracy of verification with shorter training and testing time if it is applied to online speaker verification purposes.

vi. Deep Neural Network (DNN)

DNN is one newest approach in speaker recognition fields [47]; particularly after the impressive results obtained from using DNNs for automatic speech recognition. DNN is essentially a multi-layer perceptron with more than two hidden layers that typically uses random initialization and stochastic gradient decent to initialize and optimize the weights [47]. Lei et al. (2014) on [48] proposed a new framework for speaker recognition in which extraction of sufficient statistics for i-vector model is derived by DNN instead of standard GMM-UBM. The proposed framework shows that the DNN approach significantly improved the i-vector speaker recognition system when compared with GMM. The approach is efficient, when there is a large data set.

2.4 Related Works

From our observation at the time of this thesis work, researchers have been striving a lot, and achieving significant improvements in the area of speaker recognition and as well as Forensic speaker recognition. The following are recent works, some of them were conducted on general-purpose speaker recognition and the rest were on forensic speaker recognition for various languages.

2.4.1 Foreign Related Works

In 2017,a group of researchers, have been carried out a study presented on[49]. It was aimed to design a robust forensic speaker recognition (verification) system by mitigating the impact of a speech corpora noise. To design the proposed system, Gammatone Frequency Cepstral Coefficients (GFCC) feature extraction technique employed with Gaussian Mixture Model-Universal Background Model (GMM-UBM) modeling approach. To mitigate the noise impact and made the system robust, voice activity detection (VAD) algorithm used as data quality enhancement technique at the preprocessing stage. The system implemented using the NOIZEUS database, which contains 30 IEEE sentences obtained from six different speakers. The proposed system trained with 4 sentence level utterances and tested with 1 sentence level of utterance for each speaker. During the experiment, the effects of training-testing duration, background noise, session variability and channel mismatch were studied. The experimental results were evaluated with equal error rate (EER) metric, and shown using detection error trade-off (DET) curves. Finally the system achieved its maximum result of 16.67% ERR. The researchers have tried a lot and achieved their best; but since speaker verification is a binary classification, speaker identification is preferable for such a multi-class classification problem.

In 2017, A. Mauray, Kumar and R.K Agrawal have been done a study on[50]. The study aimed to design both the text-dependent and text-independent speaker recognition for Hindi speech. To design the system, Mel-frequency cepstral coefficient (MFCC) was used as a feature extraction technique with vector quantization (VQ) and Gaussian mixture model (GMM) modeling approaches. To evaluate performance of their system, the researchers experimented using a 150 word level utterance speech corpora from 15 speakers. From the obtained results indication, the MFCC feature extraction technique with the GMM speaker modeling approach depicted a better recognition performance than with the VQ speaker modeling approach in both text-dependent and text-independent recognition cases. For the text-dependent case, MFCC with the VQ and GMM speaker modeling approaches achieved 85.49% and 94.12% recognition accuracy respectively. while, for the text-independent case, the MFCC feature extraction technique with the VQ and GMM speaker modeling approaches, the proposed system achieved 77.64% and 86.27% recognition accuracy respectively. The researchers tried a lot and achieved their best. The system trained and implemented using a short length (world level utterance) speech corpora; but if it had been implemented with a longer utterance corpora, the system might be achieved more.

In 2018, Arathy P. Anu G. and Leena M. studied paper[51]. The authors aimed to propose a forensic automatic speaker recognition system for Malayalam language, spoken in the Indian state. In this study, the feature extraction was conducted through the i-vector extraction technique with Probabilistic Linear Discriminant Analysis (PLDA) feature classification. To evaluate performance of the system, the implementation had been done using a 1232 paragraph level utterance speech corpora collected from 18 speakers. From the conducted experimental result indication, although the i-vector based approach offers better results in speaker recognition using normal data, the recognition accuracy decreased significantly in the case of the variability such as channel mismatch, VoIP and voice disguise.

In 2019, a group of researchers presented paper[52]. The researchers aimed to design a forensic speaker recognition system for Mexican Spanish speakers. To design the system, Mel-frequency Cepstral Coefficient(MFCC) feature extraction technique employed with Gaussian Mixture Model(GMM) feature classification approach. To implement the proposed system, the researchers used a 1200 paragraph level utterance speech corpora collected from 120 speakers of phone calls. During the experiment, the researchers provided more attention in observing the impacts of the MFCC's feature vector dimension and GMM's modeling order size on a recognition performance.

2.4.2 Local Related Works

To the best of our survey during this work, we have found the following four works on general-purpose speaker recognition, and nothing has been done on speaker recognition for specific applications of Amharic language speakers.

In 2015 Aykefam Azene[53] has been done a thesis, and the study aimed to design a text-independent offline speaker identification system for the Amharic language. As the author described in his thesis (under the section of comparison with others), this is the first speaker recognition work for the Amharic language. The researcher used MFCC feature extraction approach with VQ and GMM modeling approaches. For implementation purposes, the researcher collected a speech corpora from 50 speakers (a total of 500 speech utterances, a fixed duration (10 seconds) 10 sentence-level utterances per speaker). The study was implemented on the MATLAB platform. As the experimental results indication, the MFCC feature extraction technique with the GMM modeling approach outperformed than the VQ modeling approach, and achieved 74.20% and 84.30% IDR for VQ and GMM approaches respectively. The author had been striven a lot and achieved all his best and as well paved the way for his juniors on the area of the Amharic language. But, even if the length of the sentences is the same across all the speakers used for this study implementation, using a corresponding equal time interval utterance across all speakers questionable, because all speakers could not have equal reading or uttering speed at the time of data collection.

In January 2017, Abraham Debasu and Dagnachew Melesew published an article[54]. This article aimed to do performance analysis on a text-independent speaker identification system for the Amharic language in a noisy environments. To design the system, the researchers applied a hybrid of MFCC and GFCC feature extraction techniques with VQ, GMM and BPNN classification techniques. To handle the effect of noise, the researchers used a VAD algorithm. For implementation, a 270 paragraph level utterance speech corpora collected from 90 different speakers, and implemented on the MATLAB platform. From the carried out experimental results indication, the hybrid feature extraction technique achieved 59.2%, 70.9%, and 87.4% accuracy for VQ, GMM and BPNN modeling approaches respectively. In overall, the researchers have been done their best. But as gap, the researchers used too limited size of corpora per speaker (on average, 270/90 = 3 speech utterances per speaker), which is insufficient to train and test efficiently. So, if this work had been trained and tested with more number corpora size per speaker, the system might be deserved more than what it had been achieved.

In March 2017, Abraham Debasu published an article[55], the researcher repeated the work on [54] with a certain modification on the feature classification approaches. On article[36], a hybrid feature extraction of MFCC and GFCC techniques, while on the modified version[55], the researcher applied a hybrid feature extraction technique of MFCC, GFCC, and LPCC. Additionally, in this article, a hybrid of GMM and BPNN back-end feature classification approach is used. An experiment was performed using 300 paragraph level utterance of speech corpora, collected from 100 speakers with a fixed duration of 10 seconds. And from the obtained result indication, the proposed system depicted a significant improvement compared to the former work, i.e. it achieved 93.7% IDR. Although the researcher used an admirable design approaches and achieved a significant performance improvement, still there is a training and testing corpora size limitation per speaker, (on average,300/100 = 3 speech utterances per speaker) like the previous case on[53], and this condition exposes the obtained result to doubt. Also, as the case of[53], there is a usage of fixed time interval utterance across all speakers is another issue. In this paper, at the recommendation section, the author recommended as future work to do the speaker recognition for Amharic speaker from telephone conversation speech corpora, and on our thesis work, we accounted on this recommendation entitled, Voice Biometric based Forensic Speaker Recognition Using Machine Learning.

In 2019 Gizachew Belayneh[56] performed a thesis research. On this thesis work the researcher aimed to design Artificial Neural Network (ANN) based speaker identification for Amharic language speakers. The researcher utilized Mel Frequency Cepstral Coefficient (MFCC) as a feature extraction technique and Artificial Neural Network (ANN) as a modeling approach. To implement his study, the researcher used a total of 200 sentence level speech samples collected from 10 different public figure speakers. The experiment carried out using MATLAB platform. This research achieved an improved result (97.30% IDR) compared to former works which were performed on Amharic language speaker recognition. To the best of our observation, the researcher carried out an admirable work and also shown improved speech corpora size usage per speaker for training and testing (on average 200/10 = 20 speech utterances per speaker). But as a gap, at the time of data collection and preparation, the researcher organized his speech corpora with a 44,100Hz sample rate; while at the time of feature extraction, he extracted 13-dimensional Mel Frequency Cepstral Coefficients per frame, and from this two conditions, there is no logical relationship between the sampling frequency of the speech corpora and size of the extracted feature vectors.

As we noticed from the aforesaid overview, through a subsequent and collaborative efforts, significant progress and achievements were obtained. Especially on general-purpose (ideal cases, means the are conducted using clearly clearly recorded data, the researchers do not bother that much for the preprocessing aspects), a speaker recognition technology has shown a well-improved recognition performance. However, from the given related works, we can conclude that a speaker recognition for practical applications is a challenging problem yet. Moreover, forensic speaker recognition for the Amharic language has not been studied and applied. Additionally, if we notice the studies which are incorporated as related works of this thesis, their experiment section has been conducted using a uniform level of utterance of speech corpora to train and test the system. But, as mention in the background section, utterance level (length) of the speech corpora used for training and testing can affect the recognition performance, like other factors (corpora size, population, recording channel mismatch, session mismatch, noise, etc.).

This thesis work aims to adopt a text-independent speaker identification technique for Forensic Speaker Recognition, and examine the impact of level of utterance of a speech corpora on proofing identity a criminal among the given suspects. To do that, a speech corpora at word, sentence and paragraph levels of utterance will be used to train and test the proposed system.

CHAPTER 3

SPEECH AND SPEECH SIGNAL PROCESSING

In this chapter, the speech and speech signal processing aspects that are relevant to speaker recognition are discussed. Also, a brief overview about the Amharic language offered at last.

3.1 Speech

We humans express our thoughts, feelings and ideas orally to one another through a series of complex movements that alter and mold the basic tone created by voice into specific, decodable sounds. Speech is an air-pressure continuous signal produced by jointed and precisely coordinated vocal tract organs.

3.1.1 Production of human speech and its uniqueness

The act of speaking involves movements of hundreds of muscles in split-second coordination. Larynx (voice box) is the speech production system. It is made up of cartilage, and its inside lining has two folds of tissue stretching on each side and leaving a gap between them. These are known as vocal cords. When the person is quiet and breathing, the gap between the cords remains wide open and the cords are slack. But at the time of talking, singing, shouting, etc. the cords become tightening. At the tightening moment, the exhaled air vibrates the cords, and that being a cause for sound production. Our vocal cords can be in any of the different positions. If the vocal cords are slack, they vibrate approximately 80 times per second and deep tones produced. But, if they are tense, they vibrate quickly, maybe 1000 times per second and produces short waves of sound or high tones[a].

Since kids having short vocal cords, they produce short airwaves and a high voice. As the children grow, their vocal cords become longer; and that causes the voice to become deeper. Thus, the voices of adults are heavier and deeper than children's voices. Likewise, the voice of most adult men is deeper than those of women. This is because the larynx of a men is greater than the larynx of a women. Also, men have longer vocal cords.

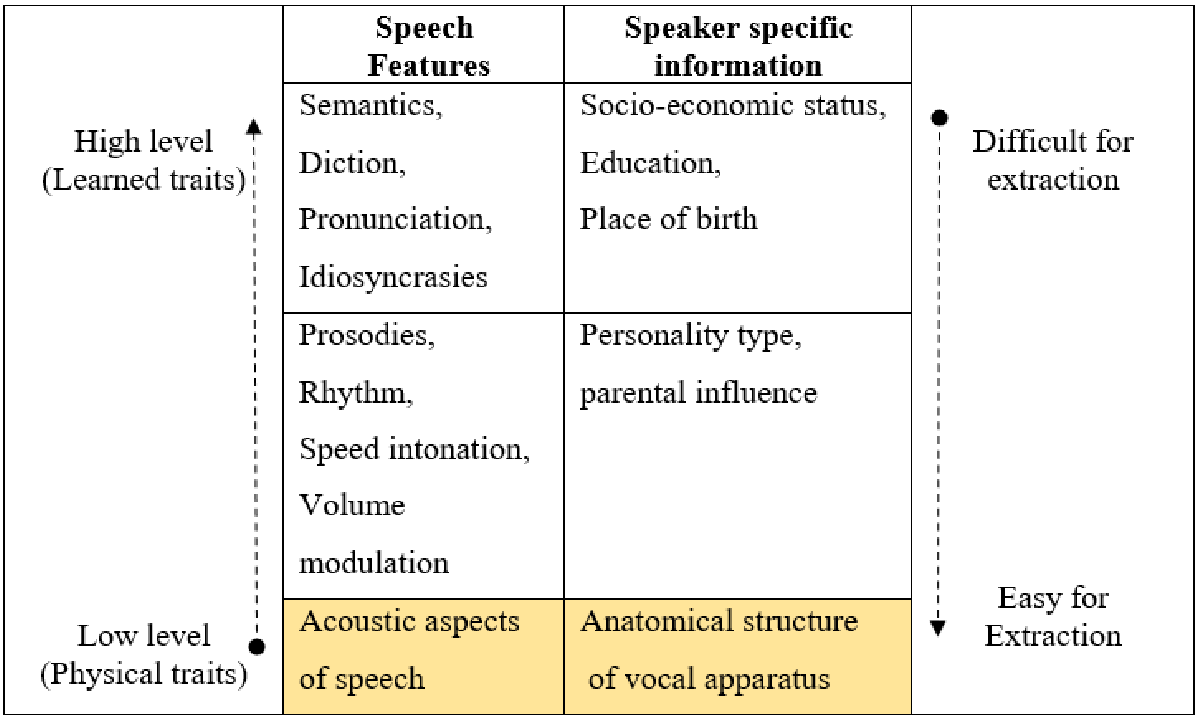
The voice pitch depends on voice cord length. There are a number of frequencies for each voice. And this range determines a person's type of voice. Many other things like a resonant space, plumes, cavities in the nose, etc. are also determinant factors for human voice variation. Movement of the tongue against the palate (also known as the roof of the mouth, forms a division between the nasal and oral cavities), shape of the lips and arrangement of the teeth are additional factors for variation of voice. Since the structures

[a] Why are human voices different?, accessed on January, 30, 2020.

and movements of all these organs are different in different persons, the voices of no two persons in the world can be identical. This dissimilarity or uniqueness of human voice is the fundamental assumption behind the idea of speaker recognition technology. Our voice is as unique as our fingerprint[b]. It helps to characterize our mood, health and personality.

Relying upon the assumption of human voice uniqueness, from the vantage point of speaker recognition technology, a person's voice is different from another due to the acoustic properties of a speech signal. It is unique to an individual due to differences which occur as structure dissimilarities in the vocal tract and/or the speaking behaviors of that that individual[1].

Behavioral (learned) and physical (anatomical) traits are the two-essential source for the uniqueness of human voice[53]. Behavioral traits are belonging to high-level cues for speaker recognition, and refer to rate of speaking, usage of phrase, pitch patterns, timing patterns, accent, dialect, etc. These high-level cues are more robust and are not much affected by noise and channel mismatch. However, only human beings can analyze and recognize them, i.e. they are too tough to extract using feature extraction techniques. On the other hand, physical traits are belonging to the low-level cues. And refer to contents like formant frequency, fundamental frequency (F0), intensity, pitch, tone, rhythm, spectral magnitude and bandwidths of an individual's voice. They are relatively easy to be extracted using feature extraction techniques, and convenient for speaker recognition purposes.



	Speech Features	Speaker specific information	
High level (Learned traits)	Semantics, Diction, Pronunciation, Idiosyncrasies	Socio-economic status, Education, Place of birth	Difficult for extraction
	Prosodies, Rhythm, Speed intonation, Volume modulation	Personality type, parental influence	
Low level (Physical traits)	Acoustic aspects of speech	Anatomical structure of vocal apparatus	Easy for Extraction

Figure 3.1: Traits of human voice

Figure 3.1 reveals the information which can be extracted from a speech signal. The information on the left side refers to the feature while the right side is related to the speaker-specific information which can be extracted from the corresponding feature. As mentioned

[b] What is Voice? What is Speech? What is Language?, accessed on August, 05, 2020

in the preceding paragraph, the higher-level features do not rely on the speaker's physical attributes. Behavioral traits are quite harder to extract than the physical ones and have drawbacks. Also, since they do not rely on the physical attributes of the speaker's vocal apparatus, they can be imitated by an impostor [53]. On the other hand, they are very robust against noise. For speaker recognition related tasks, the most important information is presented in the anatomical structure of the vocal apparatus, and they cannot be imitated. Also, unless related to aging, it is not easy for a speaker to alter his or her physical voice traits intentionally.

3.2 Speech Signal Processing

The techniques of speech processing are based on signal processing. Hence, we have to glance at the most important concepts related to signal; such signal, signal representation and frequency domain analysis.

3.2.1 Signal

Signal is an observed measurement of certain phenomenon[57]. It is modeled as a function of some independent variable. Usually, this variable is time, and we can represent the given signal as $f(t)$.

If the range and the domain of a given signal are continuous, that signal is known as analog signal. Analog signals have the advantage of being analyzed through calculus methods. However, they are too tough to be stored on computer machines. Hence, analog signals need to be converted into digital signals, in which the range and the domain are discrete. The digitized form enables to measure the signal's value at specific points of interest. Digitization is performed through sampling and quantization. For instance, let $s_n(t)$ be an analog signal as a function of time t, and if we sampled it with a sampling period T, the digitized output, $s[n]$ is given by:

$$s[n] = s_i(nT) \tag{3.1}$$

where the sampling period, T is defined as an inverse of the sampling frequency (f_s), $T = 1/f_s$. After sampling, the obtained values of the signal must be converted into some discrete set of values, and this process is known as quantization. In audio signals, the quantization level is normally given as the number of bits needed to represent range of the signal.

3.2.2 Time and Frequency Domains Signal Representation

In digital signal processing, usually, signals are studied either in time or frequency domain. Time-domain used to describe the domain for analysis of signals with respect to time. When an audio signal is examined in the time domain, the x-axis is time, so the value of the y-axis (the amplitude) depends on the changing of the signal with respect to time. Meanwhile, frequency domain used to describe the domain for analysis of signals with respect to frequency. When an audio signal is examined in the frequency domain, the x-axis is frequency, so the value of the y-axis (the magnitude) depends on the changing of the signal with respect to frequency.

As mentioned beforehand, the human speech is generated by the vibration of vocal cords. Sound pressure, which is changes in the air pressure induced by a sound wave, is the output of this operation. The sound pressure measurements are known as amplitude. A speech waveform is a time domain representation of sound. And this indicates variations over time in amplitude. This speech waveform shape tells us intuitively about the periodicity of the speech signal, i.e., its repetition over a period of time, and its representation shows us the loudness (amplitude) of the sound wave changing with time. From the definition of the sound wave, this amplitude reveals the amplitude of air particles that are oscillating because of the pressure change in the atmosphere in producing sound. However, since amplitudes only tell us about the loudness of the recorded speech, they are not well informative to acquire detailed features of the speech.

On the other hand, the frequency domain representation of a speech signal can tell us what different frequencies are presented in a give signal. Hence, for better understand and analysis of the speech signal, it is necessary to transform the signal into the frequency domain. Fourier Transform is a mathematical operation that converts the domain of a continuous signal from time to frequency. In the speech or speaker recognition systems feature extraction step, Fourier Transform is used to transform each speech frame from the time domain into the frequency domain[58].

3.2.2.1 Fourier Transform

Frequency domain analysis of a speech signal can be seen as decomposing it as sums of sinusoidal, and the analysis relying on Fourier Transform.

An audio signal is a complex signal composed of multiple ‘single-frequency sound waves’ which travel together as a disturbance in the medium. When sound is recorded, we only capture the resultant amplitudes of those multiple waves. And Fourier Transform used to decompose this signal into its constituent frequencies. It does not just give the frequencies present in the signal, it also provides the magnitude of each frequency present in the signal.

Discrete Time Fourier Transform (DFT) is a mathematical algorithm used to compute the Discrete Time Fourier Transform (DTFT) from a given time domain discrete signal sequence. The the lone variation between FT and DFT is that FT considers a continuous time signal while DFT takes a discrete time signal as an input. I.e, DFT transforms discrete time signal into its frequency constituents just like FT does for a continuous time signal. Fast Fourier Transform (FFT) is the fastest implementation algorithm of DFT. For our case, we have a sequence of amplitudes that were sampled from a continuous speech signal, so that, during the front-end feature extraction step, we will employ FFT algorithm to transform this time domain discrete signal into a frequency domain.

By nature, speech signals are quasi-stationary; their statistical parameters (intensity, variance, etc.) change over time [59]. They may be periodic in a small segments, but no longer have that characteristic when longer segments are considered. Hence, it is difficult to analyze them using Fourier transformation since it requires the knowledge of signals for an infinite time. This problem led to a set of techniques called short-time analysis. The idea is splitting a signal into short segments, known as frames, assuming that the signal is stationary and periodic in one segment and analyzing each frame separately. While employing short-time analysis technique, the initial step in extracting features is dividing the digitized speech signal into smaller frames and perform a Fourier transform on each frame to determine the containing frequencies, and this process is known as Short-Time Fourier Transform (STFT), and it allows the frames to be analyzed separately. As of [60], for speaker recognition systems, the signal of an audio does not alter much for the intervals 20 to 40 milliseconds, when the vocal tract components are assumed to be stationary.

3.3 Amharic Language

Amharic **(አማርኛ)** is an Ethiopian spoken Semitic language, which is originated from an ancient language, known as Ge'ez. When Ge'ez ceased to be spoken popularly sometime between 900 and 1200 A.D, Amharic has began to be the language of court for the population in the Highlands of Ethiopia. Currently, it is one of the five (Amharic, Oromo, Somali, Afar and Tigrinya) official languages of Ethiopia[c]. As of the 2007 census, Amharic is the second largest language in Ethiopia (next to Oromo) and possibly among the five largest languages in African.

Amharic is one of the rare languages in Africa with its own writing system. Its alphabet is known as Fidel **(ፊደል)**, which grew out from the Ethiopic script **(ግዕዝ ፊደል)**. In Amharic writing system, each syllable pattern comes in seven different forms (usually known as orders), reflecting the seven vowel sounds. The first order represented the basic form, while the rest six are derived from the basic form through modification (by attaching strokes at the middle or end of the basic form, elongating one of the leg of the basic character, etc.). The alphabet is written from left to right, in contrast to some other Semitic languages, and consists of 33 core consonants, giving 7*33=231 syllable Fidels. As Table 3.1 reveals, each Fidel represents a consonant together with its vowel (2nd column reveals the basic forms while the rests are modifications of the basic)

In addition to the 231 Fidels, Amharic language has symbols that are used to represent labialization, numerals, and punctuation marks. And as shown from Table 3.2, the Amharic language consists of 310 Fidels. The vowels are fused to the consonant form in the form of diacritic markings. The diacritic markings are strokes attached to the base characters to change their order[61].

[c] Amharic, the official language of Ethiopia, accessed on April 5, 2020.

Table 3.1: Sample of Amharic alphabets (የአማርኛ ፊደላት ናሙና)[60]

	1st	2nd	3rd	4th	5th	6th	7th
	ä	u	ï	a	é	i	o
h	ሀ	ሁ	ሂ	ሃ	ሄ	ህ	ሆ
l	ለ	ሉ	ሊ	ላ	ሌ	ል	ሎ
h	ሐ	ሑ	ሒ	ሓ	ሔ	ሕ	ሖ
m	መ	ሙ	ሚ	ማ	ሜ	ም	ሞ
.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.
f	ፈ	ፉ	ፊ	ፋ	ፌ	ፍ	ፎ
p	ፐ	ፑ	ፒ	ፓ	ፔ	ፕ	ፖ

Table 3.2: Total number of Amharic Fidels)[60].

No.	Fidels	Number of Fidels
i	Core	231
ii	Labialized	51
iii	Numerals	20
iv	Punctuation marks	8
	Total	310

When we see the Amharic sound system, it has seven vowel phonemes(see the first row of Table 3.1, from the 1^{st} to the 7^{th} order/column), i.e., sounds that distinguish words meaning. Also Amharic has a rich consonant system. And a distinguishing features of consonants is the process of emphatic sounds. As described earlier, and presented on Table 3.1, each Amharic Fidel has seven variations that has been grouped into seven orders; and each order has a particular vowel sound[d].

Currently, Amharic is one of the most widely studied languages in Ethiopia. It offers as a subject in most primary and secondary education levels. Also, it is a field of study at the Bachelor of Art (B.A.) and Master of Art (M.A.) levels. Despite it has a large population speaker and significant role in our socioeconomic affairs, Amharic is a language with low computational linguistic resources have been developed, and almost nothing has been done in terms of making the language use in the area of speaker recognition.

[d] Amharic Language, accessed on October 27,2021

CHAPTER 4

DESIGN METHODOLOGY

This chapter presented the design methodology of the proposed forensic speaker recognition system. As we can see from Figure 4.1, the proposed architecture incorporated four vital system elements: Speech corpora collection and preparation, Corpora Preprocessing, Front-end Feature Extraction and Back- end Feature classification (Suspected Criminals Modeling and Actual Criminal Identification (Feature Matching)).

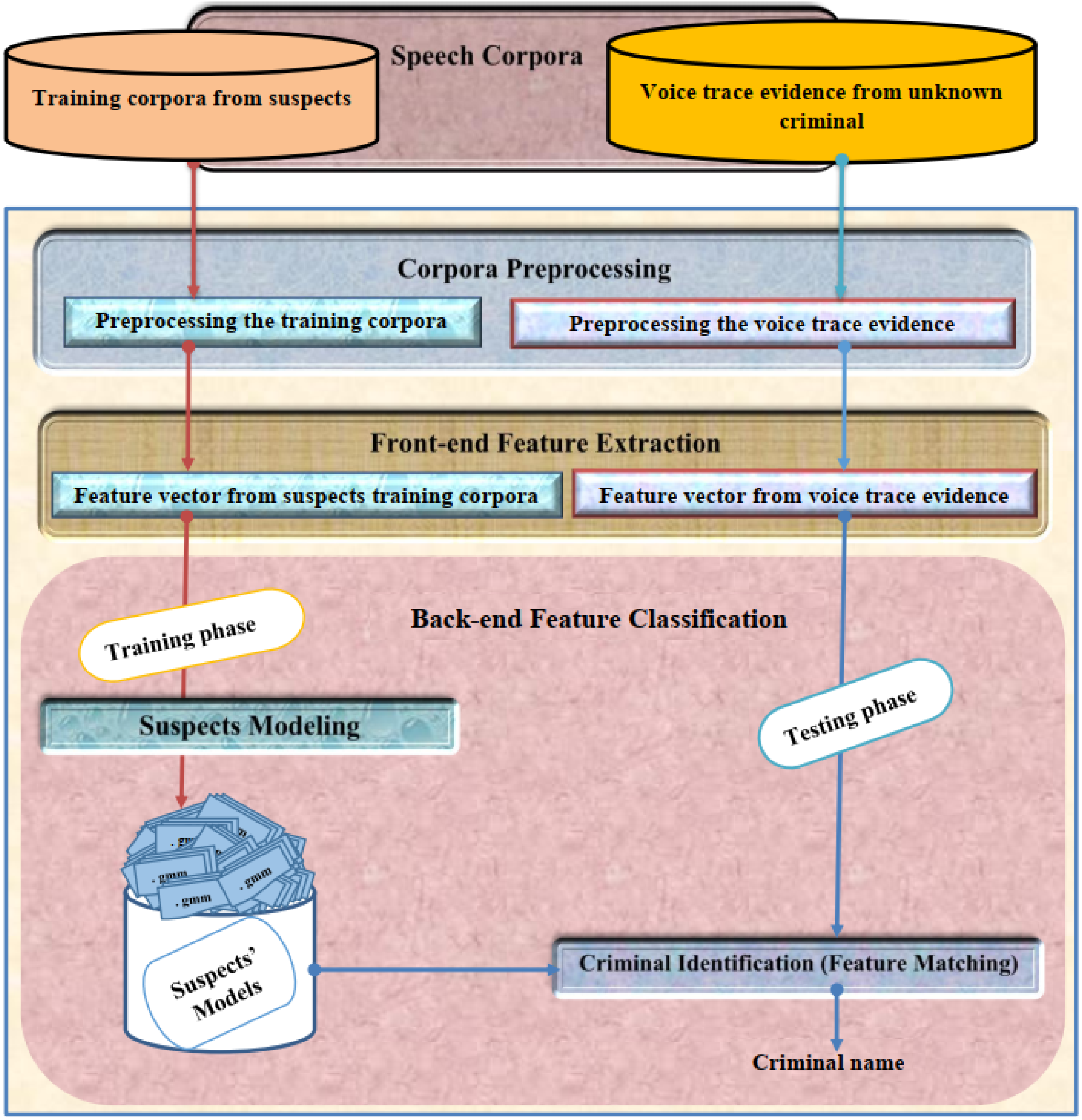

Speech Corpora
Training corpora from suspects
Voice trace evidence from unknown criminal
Corpora Preprocessing
Preprocessing the training corpora
Preprocessing the voice trace evidence
Front-end Feature Extraction
Feature vector from suspects training corpora
Feature vector from voice trace evidence
Back-end Feature Classification
Training phase
Suspects Modeling
.gmm
Suspects' Models
Testing phase
Criminal Identification (Feature Matching)
Criminal name

Figure 4.1: Overall architecture of the proposed system

4.1 Speech Corpora

Machine Learning based modeling algorithms learn from data. So, for the given problem to be solved, it is necessary that we need to feed them the right data. Also, we need to make sure that it is in a useful scale and format. In speech and speaker recognition related studies, speech corpora (audio data set) are the most indispensable resources.

There are certain publicly accessible speech corpora resources for speech and speaker recognition studies. For instance, VoxForge, VoxCeleb, LibriSpeech, TIMIT, Freesound, Common Voice Arabic Speech Corpus, etc. are among the most well know speech corpora resources[a]. However, all the world's languages are not on a balanced level in having the desired ready-made publicly available speech corpora resources to support study works in the area of speech and speaker recognition. As an evidence, we couldn't found publicly available speech corpora for Amharic language. Under-resourced languages such as Amharic, create a significant challenge to conduct speech-oriented studies, such as speech recognition, speaker recognition, speaker diarization, etc. Hence, the lack of publicly available data has been enforced us to prepare our speech corpora for this thesis work implementation. Figure 4.2 reveals the methods we used to collect and prepare our corpora.

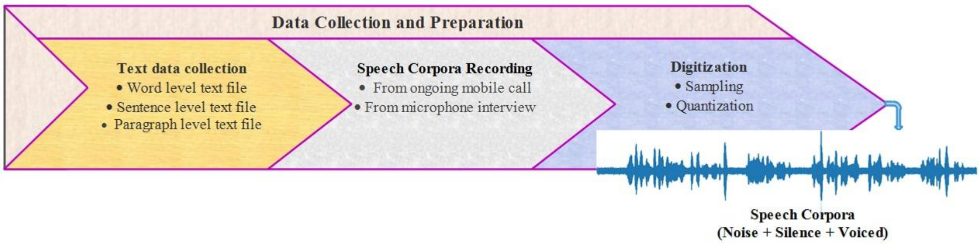

Data Collection and Preparation
Text data collection
• Word level text file
• Sentence level text file
• Paragraph level text file
Speech Corpora Recording
• From ongoing mobile call
• From microphone interview
Digitization
• Sampling
• Quantization
Speech Corpora
(Noise + Silence + Voiced)

Figure 4.2: Data collection and preparation methods

4.1.1 Text Data Collection

To prepare speech corpora for this thesis work implementation, we have started from collecting text corpora. We have collected Amharic text corpora at word, sentence, and paragraph levels. The word and paragraph level text files are collected from Facebook and Telegram social media while the sentence level text files are constructed by the principal researcher of this thesis work. During collection, intentionally the text files are selected randomly to take into consideration the real-time situations, which means that, in case of crime committing moment criminals in their harassment, ransom, extortion, corruption, etc. calls has spoken out randomly from word, sentence and paragraph selection or orthogonality (consonant-vowel) structure perspective. Hence, to account for the real scenario in having the speech corpora to train and test the proposed system with the corpora which looks like similar to the reality, we preferred to use a randomly selected word, paragraph, and randomly constructed sentence files to prepare a reading speech type speech corpus. In fact, in selecting the text files, we have tried to encompass most

[a] 15 Best Audio and Music Datasets for Machine Learning Projects, accessed on February 9, 2020.

of the Amharic Fidels' (Amharic script characters) family sound, including the digits' sound.

4.1.2 Speech Recording

Once we have done the text data collection, we took 20 randomly selected cooperative volunteers and distributed the text corpora to the volunteers for rehearsal a week before the recording time. Lastly, on the recording day, the recording process is done simultaneously from mobile phone one-side conversation at the recipient side using the receiving smartphone, and at the caller, side using an interview microphone recorder in an open-air environment through instructing to utter 40-word, 10-sentence and 10-paragraph level Amharic text files for each volunteer speaker (each volunteer speaker acts on the behalf of impostor suspected). As revealed on Figure 4.2, each suspected speaker has a total of 60 speech utterances, 40-world level utterances (WLU), 10-sentence level utterances (SLU), and 10 -paragraph level utterances (PLU).

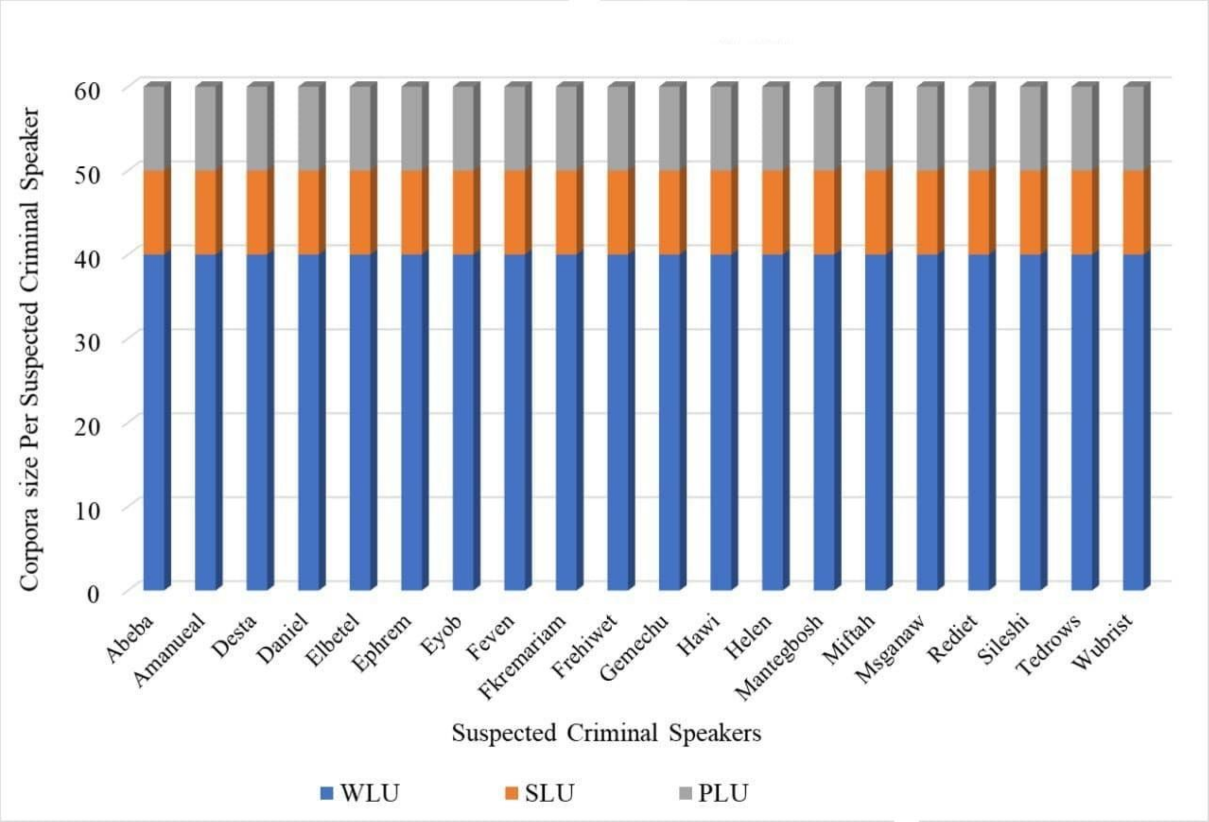


Figure 4.3: The number of utterances per pretend suspect

Table 4.1 reveals the size of speech data at word level utterance (WLU), sentence level utterance (SLU) and paragraph-level utterance (PLU) for both microphone and mobile phone recordings. While recording, we have done continuously for the entire speech of each suspected speaker. Thus, for the training and testing suitability, we have fragmented and named the speech files as presented in Table 4.2. The speech file naming involves the suspected criminal's ID (class ID column), name of the suspected criminals, name of the recording device (mobile phone or microphone), utterance level (p for paragraph level

Table 4.1: The prepared corpora size

Level of Utterance	From microphone record (No Utterance * No Speaker)	From mobile conversation record (No Utterance * No Speaker)	Sum
WLU	40*20 = 800	40*20 = 800	1600
SLU	10*20 = 200	10*20 = 200	400
PLU	10*20 = 200	10*20 = 200	400
Total	**1200**	**1200**	**2400**

of utterances, s for sentence-level of utterances and w for word-level of utterances) and utterances' ID (digit next to the utterance level).

Table 4.2: Corpus file naming and file format

Class ID	Speaker Name	Corpus File Name
Suspect 0	Abeba	Abeba_0_mobile/micraphone_p0.wav to _p9.wav Abeba_0_mobile/micraphone_s10.wav to _s19.wav Abeba_0_mobile/micraphone_w20.wav to_w59.wav
Suspect 1	Amanueal	Amanueal_1_mobile/micraphone_p0.wav to _p0.wav Amanueal_1_mobile/micraphone_s10.wav to _s19.wav Amanueal_1_mobile/micraphone_w20.wav to _w59.wav
. . .	. . .	. . .
Suspect 19	Wubrist	Wubrist_19_mobile/micraphone_p0.wav to _p9.wav Wubrist_19_mobile/micraphone_s10.wav to _s19.wav Wubrist_19_mobile/micraphone_w20.wav to _w59.wav

WAV and MP3 are the two widely used audio file formats. They have a vital role in the digital audio processing. WAV is a lossless audio file format that does not compress the original analog audio recording from which it is derived. WAV files offer a very high sample rate and bit depth, which permits them to include all the frequencies heard by the human ear. A WAV file encoder uses a technique called pulse code modulation (PCM). While MP3 is a lossy format that an audio file has compressed to a manageable size for storage, streaming, and download purposes[b].

As it can span the entire spectrum of frequencies that are audible to the human ear, WAV format is considered more useful. A 44,100 Hz 16-bit WAV has a complete frequency response up to 22 KHz. On the other hand, MP3 does not hold all the data which is audible to the human ear, where 18 KHz mark its cut off[c]. Due to its losslessness format, feature extraction from these WAV files is extremely crucial and preferable by most researchers for speech and speaker-related works.

[b] WAV Vs MP3 Files: A Guide to Audio File Formats, accessed on February 9, 2020

[c] 20 Reasons Why WAV Is Better Than MP3, accessed on March 5, 2020.

4.1.3 Digitizing the Speech Signal

As stated earlier, speech signal is a continuous air-pressure signal that can be captured by the recording device, and the recording device transforms this air-pressure signal into a continuous electrical signal. Due to this continuous nature, the speech signal captured by the recording device is analog by nature. However, in digital signal processing, it is too difficult to retain this continuity. Analog to digital converter is used to convert this continuous representation into the discreet domain so that it can be processed in the digital domain. As mentioned in the preceding chapter, sampling and quantization are the vital operations to digitize the speech signal.

4.1.3.1 Sampling

Sampling is the reduction of a continuous-time signal into a discrete. The transformation of a sound wave into a sequence of samples is a typical example of sampling. Speech recording devices convert changes in air pressure to a continuous electric signal $s(t)$, which is then sampled at an interval T, known as the period, producing a time-discrete signal $s[n] = s(nT)$, where T is the reciprocal of the sampling rate(frequency), f_s.

Sampling frequency and bit rate determine how much information from its initial analog form will be recorded and stored as a digital audio file. To avoid the aliasing effect, the analog signal should be filtered and its frequency band should be limited to the range from 0 Hz to $\frac{f_s}{2}$ Hz[62]. Thus, sampling frequency limits the maximum existing frequency in a digital file according to the Nyquist theorem. 8000, 11025,16000, 22050, 32000, 44100 Hz, etc. are the most widely used sampling frequencies of an audio file format. It means that the frequency range of the original signal will be limited up to 4000, 5512, 8000, 11025, 16000, 22050 Hz, etc.

44100Hz is the most common sampling frequency, and it is the standard for WAV audio files. It is not an arbitrary number, we humans can hear frequencies between 20 Hz and 20 kHz. Although, most of us lose our ability to hear upper frequencies due to aging, and can only hear frequencies up to 15 kHz–18 kHz; the “20-to-20” rule is still accepted as the standard range for everything we could hear. This sampling frequency allows for audio files at frequencies up to 22.05 kHz to be recorded[d].

4.1.3.2 Quantization

The sampling process produces a train of distinct real values $s[n]$; each representing the magnitude of the signal at time, nT. The domain of these real values is continuous and cannot be represented by a digital machine, an approximation of each sample value to one of the distinct levels is made; and the value is then stored as level number. I.e, the time- discrete samples are further quantized to discrete amplitude values. And end up with a signal $s[n]$ that is discrete in time and amplitude of certain bits. The bit rate limits the dynamic range of the recorded audio signal. 8-bit, 16-bit, 24-bit, 32-bit, etc.

[d] Sampling (signal processing), accessed on March 5, 2020

are well known quantizing bit rates[62]. As described in section 4.1.2, a WAV file with 44,100 Hz sampling frequency and 16-bit rate quantization has a full frequency response up to 22KHz. So, we have prepared our speech corpora with this sample frequency and 16-bit rate encoding format.

Finally, the prepared speech corpora consisted of 2,400 speech utterances from 20 speakers (10 males and 10 females) who were acted an impostor suspects. Table 4.2 reveals the overall description of the prepared speech corpora.

Table 4.3: Overall description of the prepared speech corpora

Parameter	Characteristics
Language	Amharic
Speech length	Word, Sentence and Paragraph levels of utterance
Audio type (channel)	Mono
Recorder	Smartphone and Microphone
File extension	.wav
Recording way	From Mobile call conversation and Microphone interview
Sampling frequency	44100Hz
Encoding format/Sampling format	16-bit Pulse Code Modulation (PCM)
Recording environment	Open-air
Corpora size	2.569GB

4.2 Corpora Preprocessing

To maximize effectiveness of a given study, most publicly available speech corpora are usually recorded in a constrained sound-proof environments. However, this might not be the case in practical applications, such as forensic trace evidence, which could be obtained from criminal incidents. i.e. real-life scenarios would have a chance to expose to various quality reducing factors. And these factors could be causes to degrade effectiveness of the given system. besides, the production of a speech consists of a silence and voiced regions. And as earlier research works assured, the silence regions of a speech signal retained low energy, and having insignificant contributions to offer either speech or speaker specific attributes. Figure 4.4 depicts our speech corpora preprocessing techniques.

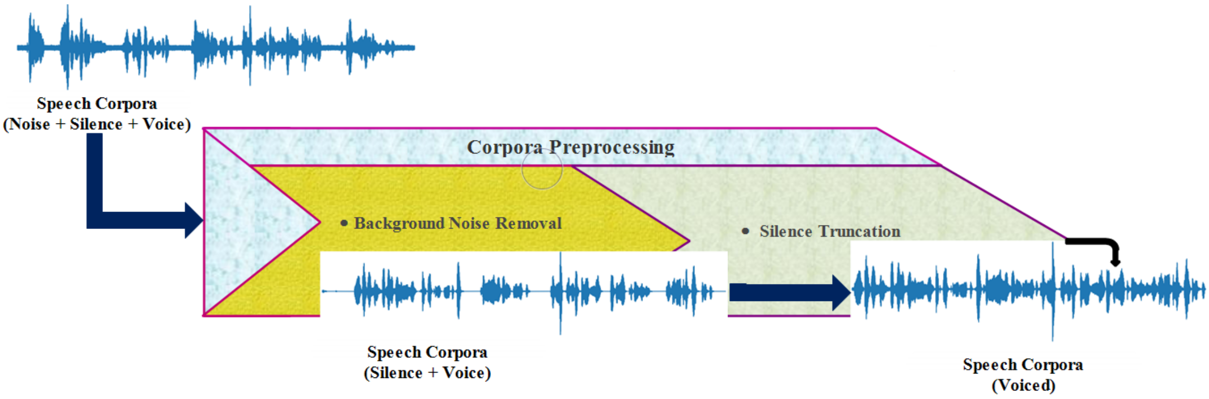


Figure 4.4: Corpora preprocessing techniques

Once having the speech corpora as an input, beforehand the feature extraction step, the speech corpora that is used either for training or testing needed to be preprocessed. The preprocessing step is used to avoid computational complexity by enhancing quality of the speech corpora; as a result, there will be a convenient condition for the subsequent front-end feature extraction and back-end feature classification steps. For our case we used background noise removal and silence truncation operations as preprocessing step.

4.2.1 Background Noise Removal

As reveals on Figure 4.5, the speech wave recorded in real environments often comprehends background noise from the recording environment. And this noise has an adverse impact on the feature extraction and feature classification process, and so degrades the performance of the speaker recognition system. Hence, removing this noise from the speech is helps to obtain better feature vector engineering.

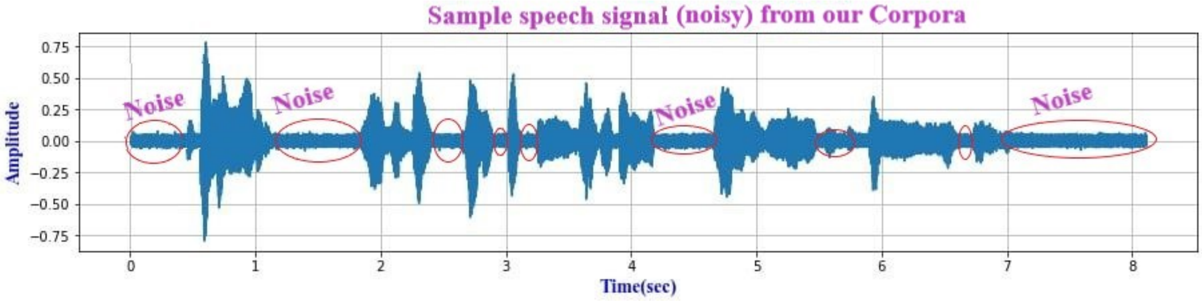


Figure 4.5: A sample noisy speech signal

For our case, to reduce an background noise from our corpora, we used a spectral gate based technique with the following steps we have used:

- Frame the speech signal into short time intervals
- Compute an FFT over the noise signal
- Compute statistics over an FFT of the noise
- Compute a threshold based upon the statistics of noise
- Compute an FFT over the voiced signal
- Determine a mask by comparing the voiced signal FFT to the threshold
- Smoothed the mask with a filter over frequency and time
- Apply the mask to the FFT of the voiced signal
- Reduce the noised and save it back the voiced signal

4.2.2 Silence Truncation

As of [63], for practical applications, such as speech recognition, speaker recognition, speech coding and speech synthesis the most significant speech or speaker specific attributes are presented in the voiced regions of the speech signal. In contrast, the silence regions do not have a significant feature content. Hence, truncating the silence regions, and extracting the desired features from the voiced regions helps to reduce computational complexity, and achieve high system performance. To do so, like depicted on Figure 4.6(the speech signal given on Figure 4.5, after the background noise removed), detecting the silence regions from the voiced is needed.

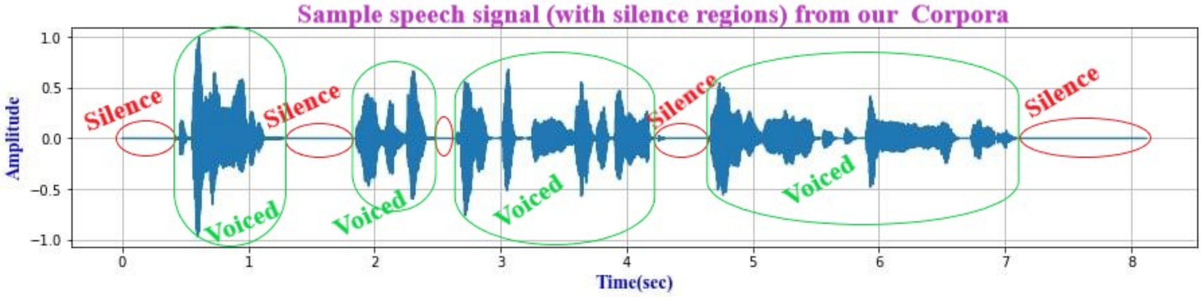


Figure 4.6: A sample speech signal with voiced and silence regions

Voice activity detection (VAD) is a speech signal processing technique in which the presence or absence of human speech is detected. It is a useful technique to enhance quality of the corpora in audio related works (such speech recognition, speech coding, speech enhancement and speaker recognition). It is a well known technique in speaker recognition framework to truncate the silence regions[64].

Short time energy based VAD is one typical implementation means of VAD in making a distinction between the silence and voiced regions of the speech signal. the speech's signal being stationary in short time interval and the voced regions having more frame energy than the silence one are the two fundamental consideration behind the short time energy based VAD algorithm. Thus, while implementing the short time energy based VAD, framing the speech signal, computing the energy per frame, defining a threshold value and discarding the region below the threshold value are vital activities to enhance quality of audio corpora. The normalized short time energy E_f of the speech signal $s_i[n]$, for each i^{th} frame having samples n, with frame length N_f given by equation (4.1)[63]:

$$E_f[n] = (\frac{1}{N_f}) \sum_{n=1}^{N_f} |s[n]|^2 \tag{4.1}$$

where E_0 is an energy value characterizing speech to silence ratio, it is feasible to express the computed energy, $E_f[n]$ in decibel as $E_{f_{dB}}$:

$$E_{f_{dB}}[n] = 10 \log_{10} \left(\frac{E_f[n]}{E_0} \right) \tag{4.2}$$

Once the frames' energy obtained, the silence regions detected by finding frames with maximum (peak-to-peak) energy less than the predefined threshold value, E_t. Once the short time energy computed by above equation, the VAD array constructed by:

$$vad[n] = \begin{cases} 0, & E_f \le E_t, \ Silence \ region \\ 1, & E_f > E_t, \ Voiced \ region \end{cases} \tag{4.3}$$

Then, the voiced regions (speech frames) $s_v[n]$ can be computed by:

$$\tilde{s}[n] = s_i[n] * vad[n] \tag{4.4}$$

Finally, the frames retained higher energy are classified as speech and while the rest of the frames are discarded as being non- speech. This method is parametric, so that it needs a threshold parameter to be set manually.

For our case, we used the following steps presented on Algorithm 1 by summering the ideas stated from equation (4.1) to equation (4.4):

Algorithm 1: Short time energy based VAD

```
Step 1: Framing the speech signal.
        $s \leftarrow \sum_{n=1}^{N_f} s[n] \implies \sum_{i=1}^{N_f} f[i]$
Step 2: Computing the normalized short time energy per frame.
        $E_f[n] \leftarrow (\frac{1}{N_f}) \sum_{n=1}^{N_f} |s[n]|^2$
Step 3: Express the computed short time energy in decibel.
        $E_{f_{dB}}[n] = 10 \log_{10} \left( \frac{E_f[n]}{E_0} \right)$
Step 4: Constructing a VAD array.
        $vad[n] = \begin{cases} 0, & E_f[n] \le E_t & \texttt{/* Silence region */} \\ 1, & E_f[n] > E_t & \texttt{/* Voiced region */} \end{cases}$
Step 4: Filtering the voiced region.
        $\tilde{s}[n] \leftarrow s_i[n] * vad[n]$
Step 5: Discarding the silence region and saving back the viced region
        to the original format.
```

4.3 Front-end Feature Extraction

Despite our speech corpora went through the preparation and pre-processing steps, hitherto, it is a raw and complex input from the machine understanding perspective. i.e. feeding this raw speech corpora directly to the classifier model is not suitable. Hence, before the back-end feature classification step, the requirement of front-end feature extraction is arising.

In speaker recognition systems, feature extraction is a front-end signal processing technique, which is used to convert the preprocessed digital speech signal into sets of numerical descriptors, known as feature vectors. These feature vectors carrying the essential speaker-specific attributes of the speech signal that enable the machines to recognize identity of the speaker using his or her voice. Also, the feature extraction technique is used to reduce dimension of the input feature vector despite maintaining the perceptive power of the signal. The feature vectors are thought of as a more compact, less redundant, and more statistical modeling friendly way of representing the raw speech signal[60]. They capture the speaker's vocal tract structure which is a substantial part how the voices differ and used to train or test the model in the back-end feature classification step.

Following the works of James E Luck,1969 [14], Bishnu S Atal, 1974 [15] and Sadaoki Furui, 1981 [16], Cepstrum based feature extraction became the dominant technique in speaker recognition, specifically for text-independent speaker recognition tasks. The reason is that, as stated under section 2.1.4, cepstrum based feature extraction obtained widely acceptance due to their deconvolution capacities to separate the vocal tract envelope and glottal excitation components of the speech signal. In text-independent speaker recognition the focus is modeling the given speaker uniquely via his or her vocal tract envelope(vocal tract structure).

Linear Prediction Cepstral Coefficients (LPCC) [16], Perceptual Linear Prediction Cepstral Coefficients (PLPC) [17], Bark Frequency Cepstral Coefficients (BFCC) [65], and Mel-Frequency Cepstral Coefficients (MFCC) [18] are few of the well-known and most commonly used cepstral feature extraction mechanisms both in speech and speaker recognition areas. In designing of the speaker recognition systems, to accomplish a better recognition performance, selecting an appropriate and most effective feature extracting technique is among the vital issues.

In 2014, a study on [66] carried out for performance comparison between LPCC, BFCC, and MFCC feature extraction techniques with ANN classifier. And findings of the experiments have shown as MFCC outperformed compared to LPCC and BFCC. In 2017 and 2018, another two studies, [67] and [68] had been done by the same group of researchers. On [67], the performance of MFCC and BFCC for speaker identification has been analyzed with VQ classifier on the bases of identification accuracy for population size, gender and computational time. On[68] the same scenario of [67] repeated just by replacing the VQ classifier with GMM. Finally, it is found that the MFCC feature extraction technique outperformed as compared with BFCC in both cases.

MFCC is a cepstral feature extraction technique which operates based on the known variation of the human ear's critical bandwidths. Psychophysical studies have shown that the human perception of frequency contents of sounds for speech signals does not follow a linear scale[69]. Meanwhile, the speech generated by humans is filtered through the shape of the vocal tract, and representing that shape accurately is the principal role of a given feature extraction technique. The advantage of MFCC is its ability to represent that shape in a more appropriate and accurate way better than the other techniques. To simulate the human auditory system through capturing phonetically important characteristics of a speech from a given speaker, MFCC employs a mel-scale spaced collection of filters. For this thesis work, we have selected MFCC as a front-end end feature extraction approach. The detailed procedures of MFCC with its block diagram presented below in section 4.3.1.

4.3.1 Mel Frequency Cepstral Coefficients (MFCC)

Initially, MFCC was acquainted by Davis and Mermelstein in 1980[18]. For speech and speaker recognition, it is an audio feature extraction technique, and in speaker recognition, it is used to extract speaker-specific parameters (features that enable to represent a given speaker uniquely) from a given speech using mel-scale spaced filter bank.

MFCC uses the principle of human auditory system simulation. It tries to mimic the way of our ears work, the ears analyze speech waves linearly at low frequencies and logarithmically at high frequencies. When the frequency bands are placed logarithmically in MFCC, it estimates the human system response more carefully than any other technique. MFCC plays on the following five facts to mimic the human hearing perception[58]:

- The human hearing perception does not follow a linear scale
- Each voice tone has an actual frequency measured by hertz
- Each voice tone has a subjective frequency measured by Mel scale
- Subjective frequency helps to capture important characteristic of phonetics
- Mel-frequency scale is linear below 1kHz and logarithmic above 1kHz

Figure 4.7[58] reveals the procedures that we followed to extract MFCC feature vectors from our preprocessed digitized speech corpora, $s[n]$.

4.3.1.1 Pre-Emphasis

As stated under section 3.1.1, vocal tract is a system that is responsible for speech production. Due to the structure of this system, damping occurs in high-frequency regions (high frequencies of the speech signal formed in the vocal tract are attenuated as the sound passes through the lips). As a result, in the process of computing the speech spectrum, the computation of the high frequencies is more difficult than that of the low frequencies[70]. For that matter, the speech signals of the voiced regions needed to be enhanced through pre-emphasis which amplifies high-frequency regions. Usually, a finite impulse response (FIR) high- pass filter is used for this purpose. The pre-emphasis operation for the given speech signal $s[n]$, in the time domain can be expressed as:

$$\tilde{s}[n] = s(n) - \alpha s(n-1), \qquad 0.9 \leq \alpha \leq 1 \tag{4.5}$$

where α is a pre-emphasis filter constant.Thus, during the pre-emphasis by dampening some of the low-frequency information in the resultant speech signal, a more balanced between high and low frequency information would achieve for the spectrum feature computations.

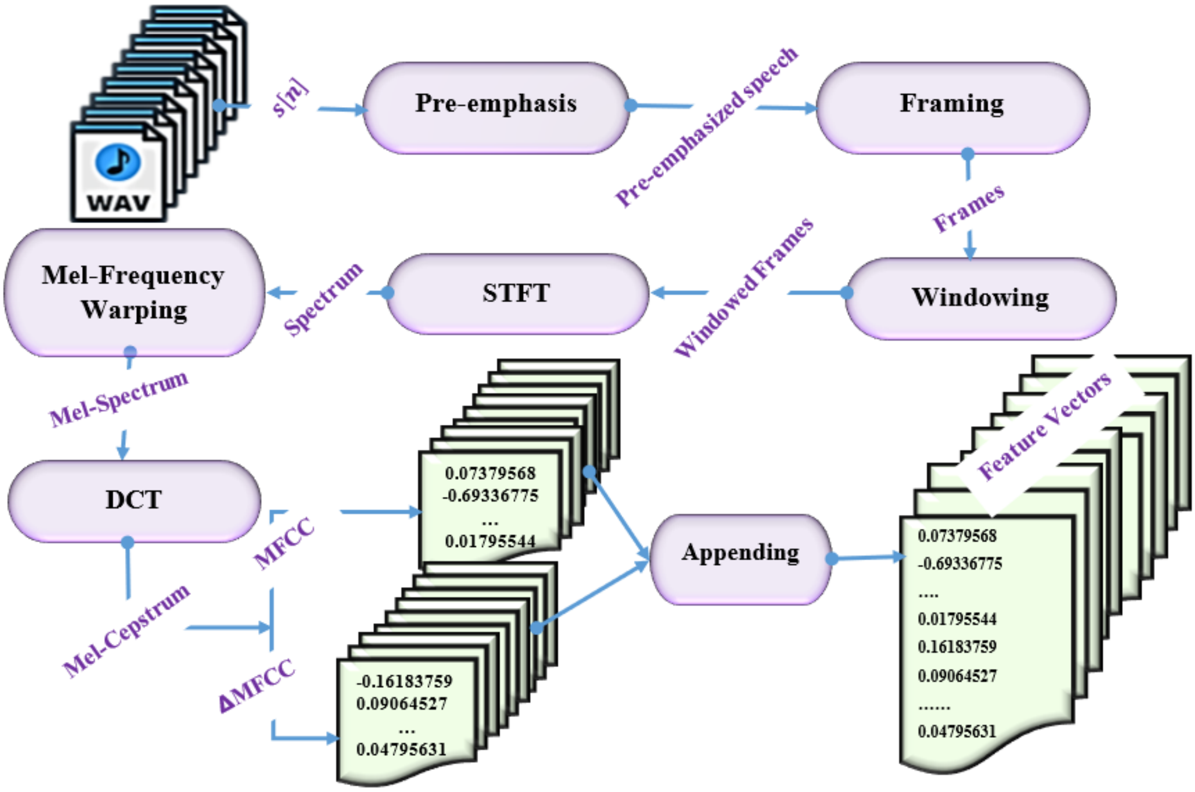


Figure 4.7: Procedures for MFCC based feature extraction

4.3.1.2 Framing

As described earlier(under section 3.2.4), speech is a quasi-stationary (time-varying) signal. But, when it analyzed over a short time interval, its properties are fairly stationary; so that,the idea of framing is required. Framing is an action by which the given length speech signal is segmented into a number of frames with N number of data points (samples) per frame [58]. Usually, the length of frames ranges from 20 to 40ms, where the vocal tract is assumed to be in a stationary moment. In speech and speaker recognition operations, framing is a means for short-time analysis of a speech signal.

When the speech signal is framed, edges of the frames at the beginning and end become sharp and lose their true and harmonic nature. This condition could be a factor to lose continuity between adjacent frames, and which could lead for misrepresentation the given speech frequencies by the systems. To tackle this problem, the concept of overlapping between consecutive frames is used while framing the speech. Frequently, overlapping between the consecutive adjacent frames ranges 25 to 75% of the frame length. For example, if a 30ms frame length is chosen, and 50% overlapping with its adjacent, the first frame would contain information about the frequencies between 0 – 30ms, the second frame would contain information about 15 – 45ms, the third would contain between 30 – 60ms, the fourth frame 45-75ms and so on till the entire length of the given speech. Thus, as shown on Figure 4.8, each frame would have a chance to retain information about ending and beginning halves of the consecutive frames[60].

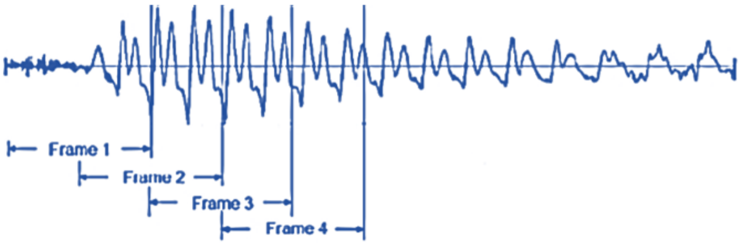


Figure 4.8: Instance of speech signal framing and overlapping

4.3.1.3 Windowing

Once the given speech signal framed and overlapped, each frame would be windowed using a window function, and this process is known as windowing. This windowing used to taper the speech signal to zero at the beginning and end of each frame, and this enables the end of each frame connected smoothly with the beginning of the next frame. Figure 4.9 reveals the task of windowing. In overall, windowing helps to minimize signal discontinuities at the beginning and the end of each frame by taking the block of the next frame into consider and integrates all closest frequency lines, so that it can diminish spectral distortion which happened during framing and overlapping of the speech signal. For instance, if $\tilde{s}_i[n]$ is the i^{th} frame of the given speech signal and the window function

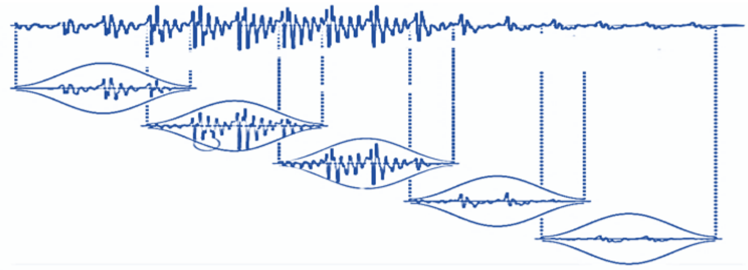


Figure 4.9: Sample for Framing, Overlapping and Windowing

is defined as $w(n)$, 0$\leq$ n $\leq$$N$-1, where N is the number of samples per frame, then the output $\tilde{y}_i[n]$ of the windowed speech signal would be defined as:

$$\tilde{y}_i[n] = \tilde{s}_i[n]w(n) \tag{4.6}$$

Triangular, Rectangular, Hanning and Hamming are the well-known window functions. Previous studies indicated that the Hamming window with MFCC feature extraction is more efficient for speaker recognition systems. For instance, on [71], a group of researchers conducted a study on three windowing functions to determine which window function has the best combination with MFCC. From their experimental result, the combination of the Hamming window with MFCC feature extraction technique has been outperformed. Mathematically, the Hamming window defined as [18]:

$$w(n) = \begin{cases} 0.54 - 0.46\cos(\frac{2\pi n}{N-1}), & 0 \leq n \leq N-1; \\ 0, & Otherwise; \end{cases} \tag{4.7}$$

4.3.1.4 Short Time Fourier Transform (STFT)

As we have been discussed under section 3.2.2, most of the time, in digital signal processing, signals are analyzed either in time or frequency domain. In time domain, the speech signal representation depicts the amplitude of sound wave changing with time. But, this amplitude is not very informative, as it only tells about the loudness of the speech. On the other hand, the frequency domain representation of a speech signal can tell what different frequencies are present in a given speech signal. Fourier Transform is a technique to transforming a signal in time domain into its spectrum in a frequency domain.

Once the windowing is done, the next step is transforming each windowed frame into a frequency domain. DFT is an algorithm that transforms the time domain signals to frequency domain components. I.e, discrete time domain data sets are transformed into discrete frequency domain representation. For instance, if $\tilde{y_i}[k]$ represents the DFT for the framed signal i, and k denotes the DFT length, the spectrum in frequency domain can be defined as[58]:

$$\tilde{y_i}[k] = \sum_{k=0}^{N-1} \tilde{y_i}[n] e^{\frac{-2\pi jkn}{N}}, 0 \leq n \leq N-1 \tag{4.8}$$

where N is the number of sample data points used to compute the FFT and j is an imaginary unit. The spectrum and magnitude spectrum can be computed as:

$$\tilde{y_k} = (\frac{1}{N})|\tilde{y_i}[k]|^2 \tag{4.9}$$

FFT is the fastest implementation of DFT. For M size of operation, DFT can be performed as $O(M^2)$ in time complexity, whereas FFT reduces the time complexity in the order of $O(Mlog_2M)$. In equation (4.7), for $0 \leq k \leq N-1$, to facilitate FFT, the frame length has to be as power of 2. To do so, it is required to pad zeros with the frame to make frame length a nearest power of 2. If N is not a power of 2, otherwise zero padding is not required [72].

4.3.1.5 Mel-Frequency Warping

Mel is a unit of pitch proposed by Stevens,Newman and Volkmann in 1937 [72]. The mel-scale helps to relate the perceived subjective frequency, m_f to its actual measured frequency, f_h of a voice tone. We humans are good at discerning small changes in pitch at low frequencies than they are at high frequencies. The mel scale performs based on the way humans distinguish between frequencies.

Mel-Frequency warping is the task of mapping the actual frequency into mel-scale. As shown on Figure 4.10, Mel-frequency filterbank is a collection of triangular overlapping band pass filters where the placement is based on the mel-frequency scale. As described earlier, this frequency scale is designed to mimic the human hearing perception. The relationship between the actual frequency, f_h in Hertz and the subjective frequency, f_m in mel-scale can be defined as:

$$f_m = \log_{10} 2595(1 + \frac{f_h}{700}) \tag{4.10}$$

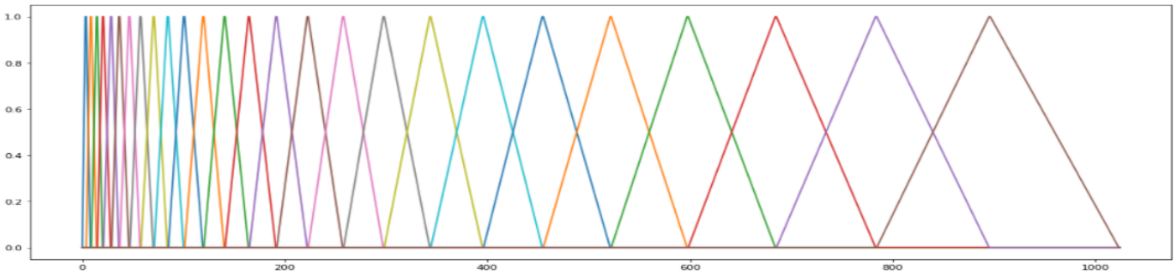


Figure 4.10: Sample Triangular Mel scale filterbank with 26 filters

If the number of filters in the filterbank is L, these L number of overlapping triangular filters are spaced between mel_{min} and mel_{max} to form a filterbank in a mel-scale. As shown on Figure 4.6, each filter overlaps at the center of the mel-scale frequency. A filterbank has bandwidth that can determine from f_{min} and f_{max} difference, $bandwidth =$ f_{max} - f_{min}. As described earlier, our corpora prepared with 44,100Hz sampling rate. Hence, we chose $f_{min} = 0Hz$ and $f_{max} = 22,050Hz$. Indeed, mel_{min} and mel_{max} are the the mel scale representation of f_{min} and f_{max} respectively.

Each filter in the filterbank is characterized by start(s), center(c) and end(e) frequencies, i.e., f_{ms}, f_{mc} and f_{me} respectively. Using an inverse operation of equation (4.10) we can compute f_{hs}, f_{hc} and f_{he} by $f_h = 700(10^{\frac{f_m}{2595}} - 1)$ Hz. Next, we map the frequencies f_{hs}, f_{hc} and f_{he} to the corresponding nearest FFT indexing numbers given by f_{bin}^{hs}, f_{bin}^{hc} and f_{bin}^{he} respectively, known as FFT bins, and defined as[55]:

$$f_{bin}^{h\theta} = \left\lfloor \frac{(nfft+1)f_m}{f_s} \right\rfloor, \qquad \theta \in \{s, c, e\} \tag{4.11}$$

where f_s is sampling rate of the speech signal, $nfft$ is the number of FFT and $\lfloor . \rfloor$ denotes the floor value. As we can see from Figure 4.10, maximum weight of the filters is at center bin, f_{bin}^{hc}, that is 1 and 0 weight at start and end bins, f_{bin}^{hs} and f_{bin}^{he}. The weights, $H_m(k)$ are computed as[72]:

$$H_m(k) = \begin{cases} 0, & if k < f_{bin}^{hs}; \\ \frac{k - f_{bin}^{s}}{f_{bin}^{c} - f_{bin}^{hs}}, & if f_{bin}^{hs} \leq k \leq f_{bin}^{hc}; \\ \frac{f_{bin}^{he} - k}{f_{bin}^{he} - f_{bin}^{hc}}, & f_{bin}^{hc} \leq k \leq f_{bin}^{he}; \\ 0, & k \geq f_{bin}^{he}; \end{cases} \tag{4.12}$$

Then, in mel-frequency warping, for each filter in the filterbank, the filter weight is multiplied with the corresponding power spectrum, and summed up all the product within the filter to obtain filter energy, and it is defined as[73][74]:

$$\tilde{E_k} = \sum_{k=1}^{L} \tilde{y_k} H_m(k), \qquad 1 \leq m \leq L \tag{4.13}$$

where E_k is the filters' energy, and $H_m(k)$ is the weight of the k^{th} energy spectrum bin, given by equation (4.12).

4.3.1.6 Discrete Cosine Transform (DCT)

As stated earlier in chapter three, a speech is resulted from vocal tract as convolution between the vocal tract impulse and glottal pulse (excitation components the speech). Speaker specific traits that used to represent an individual reside in the vocal tract impulse, and deconvoluting the two parts is the vital concern of this step.

Convolution is multiplication in time domain, but it is sum in frequency domain. Hence, by taking the inverse FFT or discrete cosine transform of the log of the magnitude spectrum, the glottal pulse, and the impulse response can be separated. Hence, in this case, the log Mel spectrum is converted back to the time domain, but not the original time domain, known as in the quefrency domain. The result is called the Mel Frequency Cepstrum Coefficients (MFCC). This cepstral representation of the speech spectrum provides a good representation of the local spectral properties of the signal for the given frame analysis. Since the Mel spectrum coefficients are real numbers,they can be converted to the time domain using the discrete cosine transform (DCT). And the MFCC can be computed using the equation defined below[60]:

$$MFCC_d = \sum_{k=1}^{D} \log_{10} \tilde{E_k} \cos[(d(k - \frac{1}{2}))\frac{\pi}{D}] \tag{4.14}$$

where $d = 1, 2, 3, ..., D$, is dimension of the feature vector MFCC, and E_k represents the filters energy, output of the k^{th} Mel-filter obtained from equation equation (4.13). This set of coefficients is called an acoustic vector. These acoustic vectors can be used to build a reference model during the enrollment or training phase and to Identify during the testing phase the voice characteristics of the given speaker. Thus, each input utterance is transformed into a sequence of acoustic vectors, to represent and recognize the speaker.

4.3.1.7 Dynamic Mel-Frequency Cepstral Coefficient (DMFCC)

The standard MFCC coefficients offer a good representation of the static cepstral properties of the framed speech. Also, a large amount of information resides in transition between frames of the speech too. As[75], in addition to the standard cepstrum coefficients, cepstrum difference coefficients are significant traits for speaker recognition. The usage of these transitional coefficients is inspired by their capability to capture dynamic cepstral information. Besides, in speaker recognition application, these transitional feature sets retain channel invariant and speaker specific information. For instance, on paper[76], these transitional coefficients had been tested for channel compensation of telephone speech on text-independent speaker identification task, and it had shown improvements. Transitional coefficients do not perform as good as static when they used by themselves; rather, they are normally utilized in combination with their static traits.

The dynamic features obtained by computing differences of the consecutive static features, i.e. delta MFCC (ΔMFCC) obtain from MFCC. And delta MFCC (ΔMFCC) features are considered as first-order differential (also known as velocity) and can be obtained using equation (4.15) while the Delta-Delta MFCC (ΔΔMFCC) features are considered as second-order differential coefficients(also known as acceleration)[76]:

$$\Delta MFCC_d = \frac{\sum_{\tilde{n}=1}^{\tilde{N}} n(MFCC_{d+\tilde{n}} - MFCC_{d-\tilde{n}})}{2\sum_{\tilde{n}=1}^{\tilde{N}} \tilde{n}^2} \tag{4.15}$$

where $\tilde{N}$ is depth of the feature vector, and its typical value is 2.
In 2007, on [77], an experiment has been carried out to reveal speaker identification performance versus modeling order of GMM using combined cepstral coefficients of MFCC. During the experiment, the incorporation of the first order, ΔMFCC coefficients to the static, MFCC coefficients outperformed than the the usage of static coefficients by themselves for all size of modeling order of GMM. However, the incorporation of both first order, ΔMFCC and second order, ΔΔMFCC coefficients gained not much advantage or improvement, and even slightly degrades the performance with the increases of the GMM's modeling order, compared to merely first order coefficients incorporation. Finally, based on their experimental results, the researchers of [77] postulated that, The use of first-order difference coefficients is sufficient to capture transitional information while maintaining reasonable dimensional complexity.

The final goal of the entire feature extraction task is obtaining a D-dimensional feature vectors, that can be represented the given speaker uniquely relying on his or her speech signal traits. For this thesis work, we have employed our implementation using appended acoustic feature vectors of the static MFCC with their corresponding first order dynamic, ΔMFCC coefficients for suspected criminal representation training, and actual criminal identification. The dimension of the feature vectors and related issues will present in the implementation part, (chapter Five). Based on the extraction context phase, these extracted set of feature vectors will serve to build a model for suspects or to identify a criminal by feeding it as an input to the back-end feature classification approach.

4.4 Back-end Feature Classification

As we have seen in the preceding section, the individual suspected speaker's speech corpora is transformed into an acoustic feature vectors of D-dimension. On this step, we will generate a reference model for each suspected speaker through training phase, and will discern the criminal through identification phase. As discussed under section 4.3, the feature extraction operation had been transformed the input speech signal into a D-dimensional acoustic feature vector of MFCC. Now, the next operation is modeling through training. The modeling task takes the extracted feature vector as training input and generates a reference model for each suspected speaker through training that able to represent each suspected speaker uniquely.

The problem of speaker recognition belongs to a much broader topic in scientific and engineering so-called pattern recognition. The goal of pattern recognition is to classify objects of interest into one of a number of categories (classes)[53]. The objects of interest are generally known as patterns, and in our case are sequences of acoustic feature vectors that are extracted from an input speech corpora. While the classes refer to the suspects.

With their respective set of state-of-the-arts (algorithms), Machine learning (ML) and Deep learning (DL) are the most well- known classifier approach categories. In general DL modeling approaches are more effective for pattern recognition applications. However, they are both corpora size and execution cost expensive. Which means modeling algorithms under DL approach category demanded a large size of corpora set, long training-testing duration and high-performance processing machines. This creates a big challenge to deal with such pattern recognition studies for low resourced languages, such as Amharic. On the other hand, modeling algorithms categorized under the ML approach, relatively can perform well with limited size of the corpora set and fair execution cost. As described in section 4.1, for this thesis work, we have been prepared a limited size of speech corpora. Consequently, we have preferred to employ ML based approach for the back-end feature classification operation.

As of [78], ML is a sub-filed of artificial intelligence(AI) that aims to offer capabilities for computers to learn without being explicitly programmed. Statistical techniques are applied for retrieving a model from observed data, rather than codifying a specific set of instructions that define the model for a given data. It describes a set of techniques that are commonly used to solve a variety of real-world problems with the help of computer systems which can learn to solve a problem instead of being explicitly programmed.

As general, the ML approach depicts a key facet of a human's cognition which refers to all processes by which the sensory input is transformed, reduced, elaborated, stored, recovered, and used[78]. We humans, process a vast amount of information by using an abstract knowledge that helps us to understand the incoming input. And due to their adaptive nature, ML models are able to mimic the cognitive abilities of a human being in an isolated manner. The field of statistics is focused on statistical learning, which is defined as a set of methods and algorithms to gain knowledge, predict outcomes, and make decisions by constructing models from a given dataset. And from the viewpoint of statistics, ML can be regarded as an implementation of statistical learning [79]. Within the field of computer science, ML has the focuses on designing efficient algorithms to solve problems with computational resources [80].

Although a mix of various techniques can be used by ML models, the learning techniques typically categorized as Supervised, Unsupervised and Reinforcement learning[e]. In Supervised learning, the learning algorithm is given labeled data and the desired output. For example, pictures of dogs labeled “dog” will help the algorithm to identify the rules to classify pictures of dogs. In Unsupervised learning, the data given to the learning algorithm is unlabeled, and the algorithm is asked to identify patterns from the input data. For example, a recommendation system of an e-commerce website where the learning algorithm discovers similar items often bought together. In reinforcement learning, the algorithm interacts with a dynamic environment that provides feedback in terms of rewards and punishments.

There are several Machine Learning state-of-the-arts for back-end feature classification operations, such as speaker recognition. In a broad view, ML based feature classification state-of-the-arts can be categorized into two main categories, as template and stochastic models[81].

In template models, the model is a statistical mean. I.e, the model for each speaker is a template, T of the training feature vectors $\{T_n\}$,$n = 1...N$. Where N is the number of frames per speaker. The likelihood score for each frame is computed by a distance formula, $d(T, T_n)$ between the template and testing feature vectors. The closer an input vector to the template model, and the higher the likelihood score is obtained for a given template[53].

Template models performance better in text-dependent speaker recognition, specifically in verification [82]. If they were employed to text-independent speaker recognition, a single model could be incapable of modeling all the acoustic information that a speaker generated. I.e., too many templates per speaker would be needed, and it would cause for performance degradation. Furthermore,unlike that of identification, verification is a 1 - to - 1 classification, only one template has to be compared with the testing feature vector. VQ, NN and DTW are some instance state-of-the-arts of a template model category.

In stochastic models, the model is expressed in terms of probabilities using probability density functions (PDFs)[81]. For instance, in speech or speaker recognition works, every observation corresponding to the given speech frame is considered to be random. This implies that every utterance generated by a speaker can be taken as a random sequence of feature vectors. The Stochastic state-of-arts try to build an accurate model for those sequences attending to the random sequence statistics such as it’s mean, variance or probability distribution. The shape of the feature vectors probability distribution corresponding to a given speaker differs from the others, and the stochastic models aim to compute the likelihood score of an utterance for each speaker model.

[e] Artificial Intelligence and Machine Learning, How machines learn?, accessed on May 17, 2020

GMM and HMM are the two well-known stochastic modeling algorithms. Due to their good sequential modeling (HMM models take into account a sequence of feature vectors per frame) capability HMMs are widely used in speech recognition, and they also commonly used in text-dependent speaker verification. On the other hand, GMM is considered as a single sate of HMM modeling, and it takes into account a single feature vector corresponding to a single frame. As [76], the individual component of a GMM can be used to represent speaker-dependent spectral shapes that are effective for modeling a given speaker identity.

In 1995, as Reynold and Rose have been shown, GMM is computationally more efficient than HMM in text-independent speaker identification task. Generally, as described, at section 2.1.4, since 1990s, especially after the detailed introduction of the GMM by Reynolds[40], it is a mainstream ML model for a text-independent speaker recognition, because of its flexibility, high efficiency and robustness. Also, it is insensitive to the temporal aspects of the speech, rather it models the distribution of acoustic observations from a speaker. Which means, GMM requires less amount of data to be trained, so that the memory requirement is less, and as well it has less computational complexity [83]. Since the aim of this thesis work is adopting a text-independent speaker identification technique for forensic speaker recognition, a stochastic model with GMM state-of-the-art is chosen as a back-end feature classification approach.

4.4.1 Gaussian Mixture Model (GMM)

As depicted on Figure 4.11, GMM is a PDF given by the weighted sum of Gaussian densities, called components of the model[84]. GMM belongs to the unsupervised classifiers category; i.e. the training data samples of the classifiers are not labeled to show their category membership and the targets are not provided. Instead, during training of the GMM classifier, the underlying PDFs of the observations are estimated. In the GMM classifier, the conditional-PDF of the observation vector with respect to the different classes is modeled as a linear combination of multivariate Gaussian PDFs.

In speaker recognition system, GMM used to model distribution of the acoustic feature vectors of every speaker[40]. GMM enable to smoothly approximate PDF of an arbitrary shape and portray distributed characteristics of different speakers' speech feature vector. Speech production is not deterministic, i.e. a particular sound is not produced by a speaker with exactly the same vocal tract shape, glottal flow due to context, co-articulation, anatomical and fluid dynamical variations. One way to model these variability is probabilistically via multidimensional Gaussian PDFs[85]. As of [76], GMM is a suitable approach in modeling a speaker for text-independent speaker identification applications. Because every mixture component (modeling order) of Gaussian in a GMM represents some broad acoustic classes and their density offers a smooth approximation to the observed sample distribution obtained from utterances of a speaker.

As we have seen from the preceding feature extraction stage, the suspects' speech corpora has been transformed into MFCC feature vectors of D dimension. As depicted on Figure in 4.9, in the GMM modeling technique, the distribution of the feature vector $\overrightarrow{v}$ is modeled using a mixture of M Gaussian components, and the Gaussian Mixture Model, λ for the modeling class can be defined by the weighted sum of M components D-variate Gaussian densities, and is given by[84]:

$$P(\overrightarrow{v}|\lambda) = \sum_{m=1}^{M} \overrightarrow{w_m} b_m(\overrightarrow{v}) \tag{4.16}$$

where $\overrightarrow{v}$ is a training feature vector of D-dimensional and $m = 1, 2, , 3, \ldots M$ is the number of mixture components; w_m is weight the mixtures , that fulfill $\sum_{m=1}^{M} \overrightarrow{w_m} = 1$ while $b_m(\overrightarrow{v})$ is the component Gaussian densities, each of them distributed according to a D-variate probability density function defined as:

$$b_m(\overrightarrow{v}) = \frac{1}{\sqrt{(2\pi)^D |\overrightarrow{\Sigma}_m|}} exp\{-\frac{1}{2}\,(\overrightarrow{v} - \overrightarrow{\mu}_m)^T\} \overrightarrow{\Sigma}_m^{-1} (\overrightarrow{v} - \overrightarrow{\mu}_m)\} \tag{4.17}$$

with mean vector $\overrightarrow{\mu}_m \epsilon R^D$ and co-variance matrix $\Sigma_m \epsilon R^{DxD}$. $(\overrightarrow{v} - \overrightarrow{\mu}_m)^T$ represents the transpose of vector $(\overrightarrow{v} - \overrightarrow{\mu}_m)$.

While modeling a suspect using GMM, the weight(w_m), mean vector(μ_m) and co-variance matrix($\overrightarrow{\Sigma}_m$) used to parameterized that suspect, and these parameters can collectively expressed as $\lambda_s = \{\overrightarrow{w_m}, \overrightarrow{\mu}_m, \overrightarrow{\Sigma}_m, 1 \leq m \leq M\}$. The co-variance matrix used to determine shape of the Gaussian, and it can be full or diagonal. In diagonal co-variance matrix, only the diagonal elements are taken and all the off-diagonal elements are set to zero. Diagonal co-variance matrix is both computationally faster and empirically more favorable than full co-variance matrices. It must be noted that using diagonal co-variance doesn't limit the representational power of GMMs. Because, any shape of full co-variance matrix can be achieved using multiple diagonal co-variance matrix components[76].

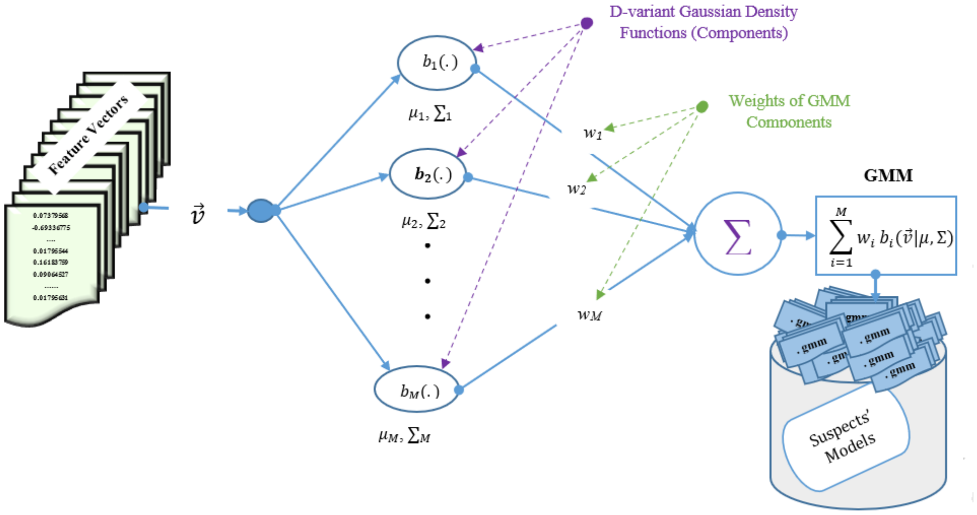


Figure 4.11: GMM with M number of modeling orders (mixture components)

4.4.1.1 Suspects' Model Training

As described earlier, while modeling a suspect using GMM, the complete Gaussian mixture density is parameterized by the $\overrightarrow{w_m}$, $\overrightarrow{\mu}_m$ and $\overrightarrow{\Sigma}_m$ from all component densities. In a speaker identification system, all the speakers represented by a GMM, and referred by their respective model. The parameters of the model are learn from the training data using a learning algorithm, which can achieve Maximum Likelihood estimation(MLE).

MLE is a method of estimating the parameters of a probability distribution through maximizing a likelihood function, so that under the assumed statistical model the observed data is most probable. The point in the parameter space that maximizes the likelihood function is called the maximum likelihood estimate[86]. MLE aims to find the model parameters which maximize the likelihood of GMM. For a sequence of training feature vector $\overrightarrow{V} = \{\overrightarrow{v_1}, \overrightarrow{v_2}, \overrightarrow{v_3}, ..., \overrightarrow{v_T}\}$, and its corresponding model λ parameters, the GMM distribution likelihood score can be defined as:

$$P(\overrightarrow{v}|\lambda) = \prod_{t=1}^{T} P(\overrightarrow{v_t}|\lambda) \tag{4.18}$$

This expression is a nonlinear function of the parameters λ , and so direct maximization is not possible (it is too tough to solve equation(4.18), because the number of equation and the unknown variables are not balanced).

For a sequence of training feature vectors $\overrightarrow{V}$, the maximum likelihood model parameters can estimated using the iterative approach, known as expectation-maximization (EM) algorithm. The EM algorithm iteratively refines the GMM parameters to monotonically increase the likelihood of the estimated model for the observed feature vectors. I.e, the basis idea of EM algorithm is, beginning with an initial model λ^m to estimate a new model and λ^{m+1} such that $p(V|\lambda^{m+1}) \geq p(V|\lambda^m)$. The new model then becomes an initial model for the next iteration, and the process would iterate till the occurrence of some threshold condition, such as $p(V|\lambda^{m+1}) - p(V|\lambda^m) \leq \epsilon$, where ϵ some small value used as threshold. For every EM iteration, the mixture weight, mean vector and covariance are computed[66]. Below are formulas to update the λ parameters of GMM while training using EM algorithm; and iterate till the occurrence of the convergence[40]: The a posteriori probability for acoustic class is given by:

$$P(m|\overrightarrow{v_t}, \lambda) = \frac{\overrightarrow{w_m} b_m(\overrightarrow{v})}{\sum_{m=1}^{M} \overrightarrow{w_m} b_m(\overrightarrow{v})} \tag{4.19}$$

The mixture weight (for each component):

$$\overrightarrow{w_m} = (\frac{1}{T}) \sum_{t=1}^{T} P(m|\overrightarrow{v_t}, \lambda) \tag{4.20}$$

The mean vector (for each component):

$$\overrightarrow{\mu_m} = \frac{\sum_{t=1}^{T} P(m|\overrightarrow{v_t}, \lambda)\overrightarrow{v_t}}{\sum_{t=1}^{T} P(m|\overrightarrow{v_t}, \lambda)} \tag{4.21}$$

The co-variance matrix (for each component):

$$\overrightarrow{\Sigma_m} = \frac{\sum_{t=1}^{T} P(m|\overrightarrow{v_m}, \lambda)\overrightarrow{v_m}^2}{\sum_{t=1}^{T} P(M|\overrightarrow{v_t}, \lambda)} - \overrightarrow{\mu}^2_m \tag{4.22}$$

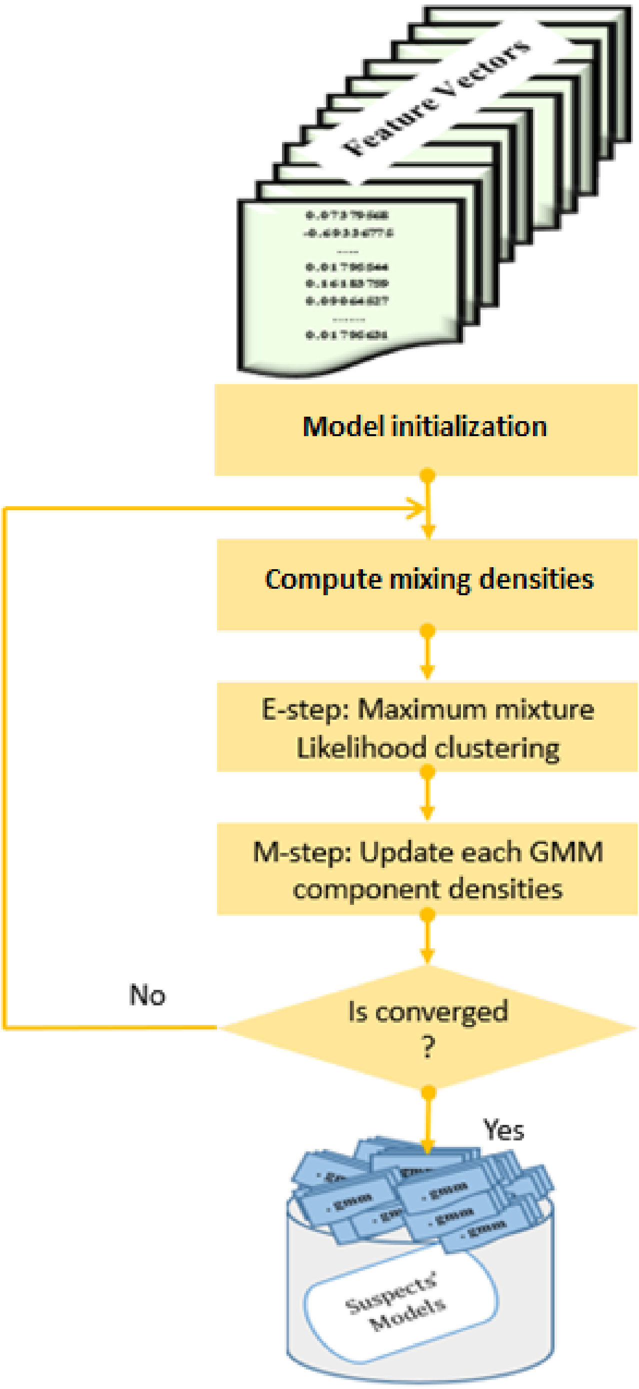


Figure 4.12: EM algorithm based Training

In general as depicted on Figure 4.12, the EM algorithm based training process can be summarized as:

- Step1: Receive the training input feature vector
- Step2: Initialize the model parameters ($\overrightarrow{w_m}, \overrightarrow{\mu_m}$ and $\overrightarrow{\Sigma_m}$)
- Step3: E-step: Compute the component mixing density for each feature vector v_t using equation (4.17)
- Step4: E-step: Compute the posterior probability for every feature vector v_t using equation (4.19)
- Step5: M-step: Update parameters of the model using the equations, (4.20) to (4.22)
- Step6: Check the convergence condition and back to step3 to iterate the training process till the occurrence of convergence or terminate the iteration and save the final model to the given directory.

when the gain in the probability between two consecutive steps is small enough, then the process would be terminated and the EM algorithm is said to be converged.

4.4.1.2 Criminal Identification

The identification subsystem is kernel of a speaker recognition system. It takes the role of brain compered to speaker identification done by humans. While performing the identification, initially, it keeps a register of the attributes of a given input voice like the human brain does in training. Then, in the testing stage, it uses the registered attributes to compare with the testing utterance attributes [85]. Like the recognition done by human, the result depends on how the system familiar with that voice. This is what the likelihood score of the speech signal represents[87].

As we have seen from the modeling section, for each individual suspect a reference model has been built and stored. I.e, all the suspects, $S_s, s = \{1, 2, 3, ...S\}$ are represented by parameters of the GMM, and referred by their respective model, $\lambda_s, s = \{\lambda_1, \lambda_2, \lambda_3, ..., \lambda_S\}$.

Here, the concern is obtaining a suspect's model with maximum log-likelihood score for the set of testing feature vectors v_t, extracted from testing utterance where $t = \{1, 2, 3, ..., T\}$ represented the frames. Now, from Bayes' rule, the recognized identity of a criminal $\tilde{C}$ can be defined by a maximum posterior probability[76]:

$$\tilde{C} = argmax_{1 \leq s \leq S} \frac{p(v_t|\lambda_s)p(\lambda_s)}{p(v_t)} \tag{4.23}$$

Considering equally likely suspects,($p(\lambda_s) = \frac{1}{S}$), So that, it is feasible to simplify equation equation (4.23) as:

$$\tilde{C} = \arg max_{1 \leq s \leq S} P(v_t|\lambda_s) \tag{4.24}$$

Taking into account independent feature vectors between observations and taking logarithm, the computational task given in equation (4.24) can be oversimplified to:

$$\tilde{C} = \arg max_{1\leq s\leq S}(\sum_{t=1}^{T} \log_{10}(P(\overrightarrow{v_t}|\lambda_s))) \qquad (4.25)$$

Here, the s^{th} suspect's log-likelihood score is given by $\sum_{t=1}^{T} \log_{10}(p(\overrightarrow{v}_t|\lambda_s))$, and $\tilde{C}$ is the identified criminal. The identified criminal $\tilde{C}$ with MLL score, obtained from equation (4.25). Figure 4.13 reveals a block diagram of feature matching process for the criminal identification process.

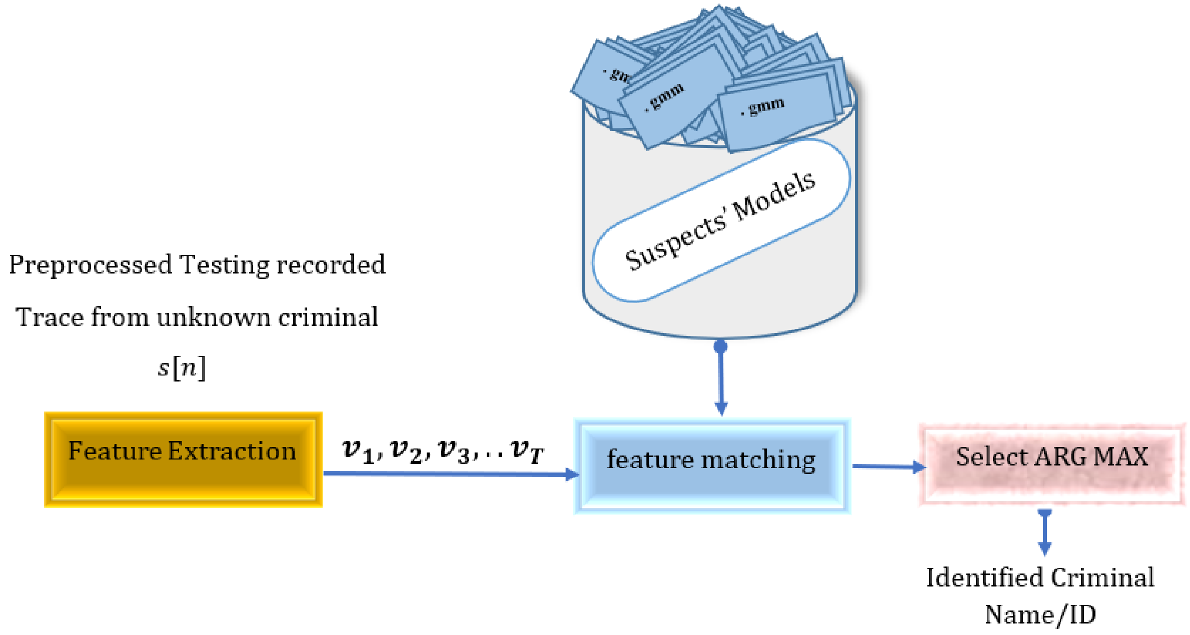


Figure 4.13: Block diagram for criminal identification process

CHAPTER 5

DESIGN IMPLEMENTATION

This chapter presented to reveal the implementation procure of the proposed forensic speaker identification (FSI) system. Figure 5.1 depicts the way how to collect training corpora from handed suspects (through interview at police department), testing voice trace evidence from an unknown criminal (from an ongoing mobile phone conversation at the receiver (most probably later on he/she would be the accuser) side), used techniques for pre-processing, front-end feature extraction and back-end feature classification (suspects modeling and criminal identification). As well, the used implementation tool and performance evaluating metric presented under this chapter.

As stated under section 2.1.3, and presented on Figure 5.1, during the implementation procedure, two consecutive phases are required, training and testing. For both phases, the input is a prepossessed speech corpora, which is collected from suspects, and unknown criminal for the respective phase. The preprocessed input speech makes to pass through MFCC feature extraction technique for obtaining speaker-specific feature vectors. During the training phase, the extracted feature vector from the respective suspected speaker speech signal feeds to the modeling algorithm,GMM. Then, theses feature vectors served to train (to obtain an optimum parameter values of mixture weight, mean vector and co-variance) the GMM using the EM algorithm. The training process is done to generate a speaker-specific model for every suspected speaker, and store the model into the models' database. During the testing phase, a feature vector wold be extracted from the trace evidence of the unknown criminal. Then, matching between the acoustic feature vectors of the unknown criminal voice trace evidence and the models of the training phase would be made. Lastly, after feature matching has been made by computing the maximum log-likelihood score, the proposed system will decide to identify the actual criminal among the handed suspects. For every part mentioned earlier, a more description presented is presented under sections 5.1 to 5.6.

5.1 Experimental Tools and Libraries

Python is a programming language for general use. It is widely used for desktop and web applications development. Also, it is possible to use python for performing complex scientific and numeric operations. For instance, researchers use python to accomplish Artificial Intelligence, Machine Learning, Deep Learning, Computer Vision and Natural Language Processing tasks[a].

[a] Why Python?, accessed on March, 11, 2020

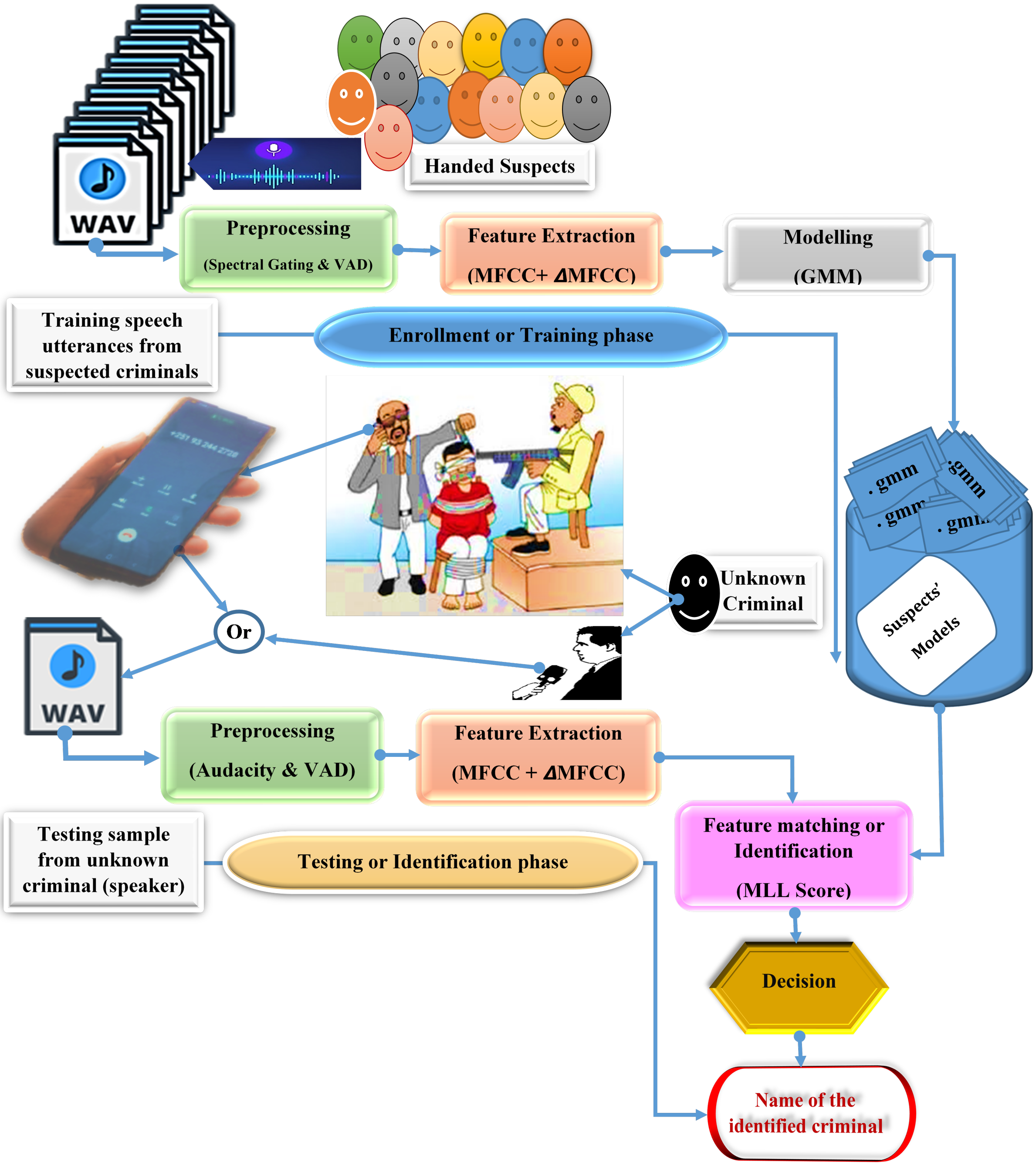


Figure 5.1: Implementation setup of the proposed FSI System

For this thesis work, we have used python for corpora pre-processing,feature extraction and feature classification tasks. The following python library packages employed for our implementation:

i. scipy.wavfile: is a python library used to read and write .wav audio files. For our case, we used this library to read the speech files from the given directory during training and testing processes.

ii. python speech features: is an open-source python library having multiple built-in audio processing methods related to feature engineering. It offers methods that are enable to extract the filter bank energies, logarithmic filterbank energies, spectral centroids and

Mel-Frequency Cepstral Coefficients (MFCC). For our case, we have used to extract the MFCC feature vectors.

iii. scikit-learn(sklearn): An open-source python library. It offers methods for various Machine Learning tasks such as clustering regression, classification, modeling, etc. For our case, we used Scikit-learn to train parameters of GMM by EM algorithm while modeling suspects during the enrollment phase.

iv. pickle: pickle is the standard way of serializing objects in python. Pickle operation can be used to serialize machine learning models, and save the serialized file format. Later we can load this file to serialize our model and use it to make a prediction. For our case, we used pickle module to pickle (store) the generated .gmm models to the models' database during training, and to de-pickle(retrieve) .gmm models from their location during identification while performing feature matching.

v. Others: Besides the library modules listed earlier, some additional python modules used for implementation were glob, matplotlib, numpy, Matplotlib, etc. for different purposes.

5.2 Training - Testing Corpora set

As presented under the design chapter, the corpora database for the proposed system implementation consisted of phonetically balanced word, sentence and paragraph levels of utterances from randomly selected 10 males and 10 females suspected speakers with each provided same 40-word, 10-sentence and 10-paragraph utterances with different texts.

Table 5.1: Training - Testing Corpora set

Level of Utterance	Training Data (NTUPCS*NSC)	Testing Data (NTTPCS*NSC)	(Training + Testing) Data
From Mobile phone conversation record			
WLU	33*20 = 660	7*20 = 140	800
SLU	6*20 = 120	4*20 = 80	200
PLU	6*20 = 120	4*20 = 80	200
From Microphone record			
WLU	33*20 = 660	7*33 =140	800
SLU	6*20 = 120	4*20 = 80	200
PLU	6*20 = 120	4*20 = 80	200
Total utterance	**1,800 (75%)**	**600(25%)**	**2400(100%)**

Where, NTUPS - is number of training utterance per suspected criminal,NSC - is number of suspected criminals, NTTPSC - is number of testing trials per suspected criminal.

Since a text-independent speaker identification is adopted for our FSI system, a subset of utterances is used for training the speaker specific model for each suspect. The training utterances with different text are same for all suspects. The other unseen subset of utterances used for testing. Also, the testing utterances were same for all suspects. So, as revealed from Table 5.1, we have set the training-testing corpora ratio 75%-25% as

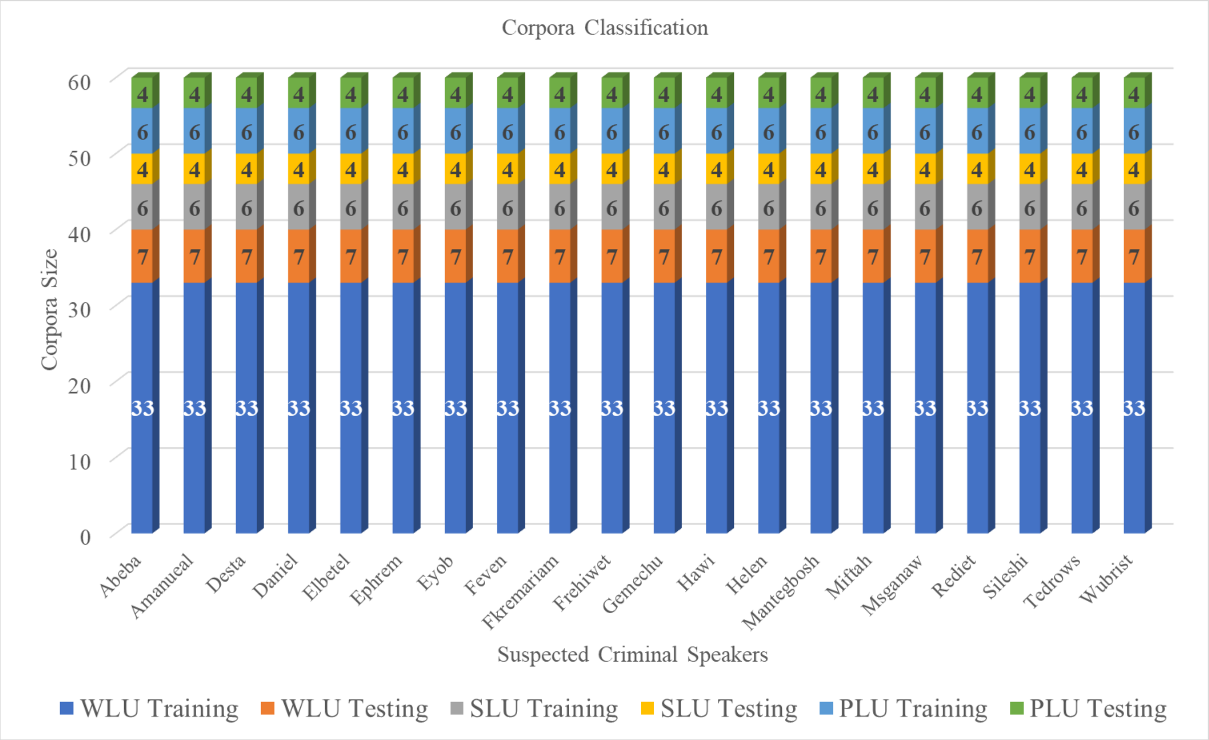


Figure 5.2: Training - Testing Corpora set for the respective level of utterance

per the protocol used by[88]; i.e. out of the entire speech corpora, 1800 (75% of the total corpora) utterances used for training to build a reference model for suspects and the rest 600 (25% of the total corpora) utterances used for testing.

Besides the total training-testing corpora set done above, furthermore, we have set the individual suspect's corpora as revealed in Figure 5.2. i.e. out of 40-WLU, 10-SLU and 10-PLU utterances, each individual suspected speaker has 33-WLU, 6-SLU and 6-PLU utterances for training purpose, and 7-WLU, 4-SLU and 4-PLU unseen (trace evidence) utterances for testing purpose.

5.3 Corpora Preprocessing

As stated under the design chapter, the preprocessing stage helps to enhance quality of the corpora; so that used to create suitability for the subsequent feature extraction and classification stages. To do so, we used background noise removal and silence truncation techniques.

5.3.1 Spectral Noise Gate based Background Noise Removal

To remove the background noise from our corpora we have used a spectral gating based noise removal technique. We have framed the input speech signal into 30ms length, and employed the steps listed under section 4.2.1. Figure 5.3, reveals sample input noised and the de-noised (bottom) speech signals. The de-noised speech signal saved to the given destination in the original file format.

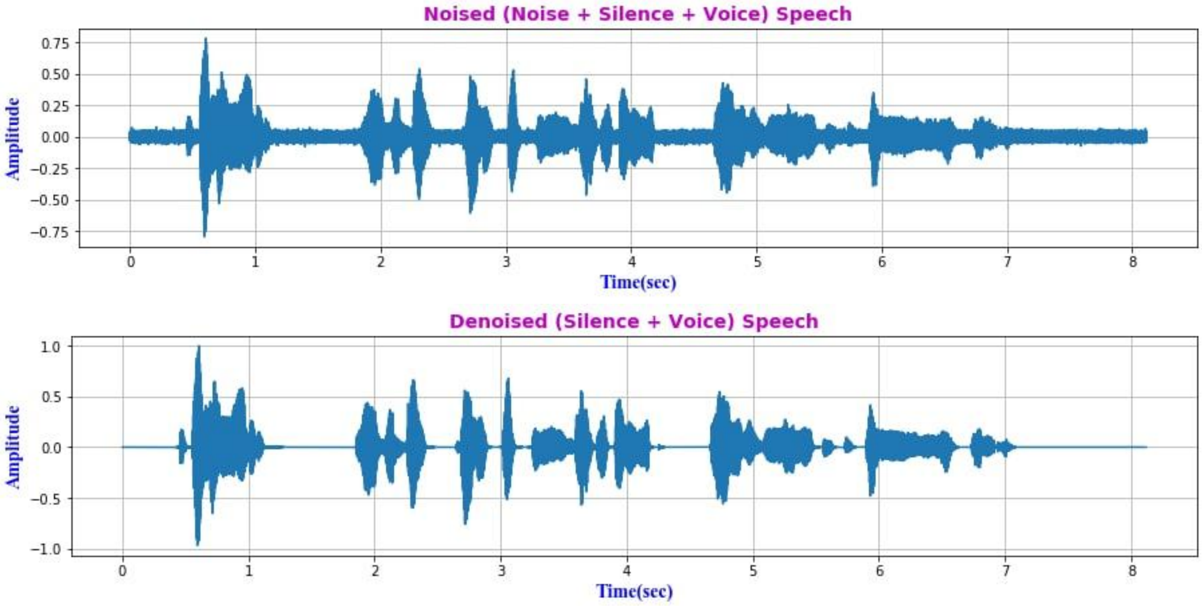
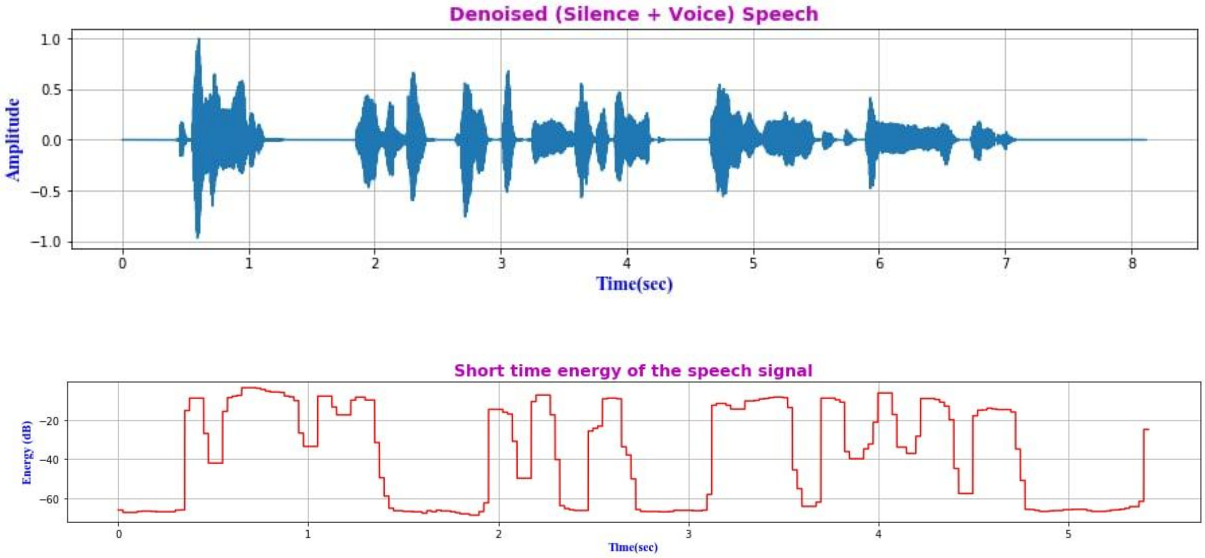


Figure 5.3: Noised and De-noised speech signal (see top to bottom)

5.3.2 VAD based Silence Truncation

As stated earlier under section 4.2.2, silence truncation is a technique used to truncate the silent regions of an audio file. And to do so, we used short time energy-based VAD algorithm, and implemented using the steps given at Algorithm 1 (see section 4.2.2).

While the implementing, the silence truncation code expected to read the de-noised speech file from the given directory, segmenting into frames, computing frames energy, converting to decibel and employing VAD array to each frame using sliding window (we have used 30ms length frame). Finally, the frames having silence, regions below the specified threshold value (we have used -30dB, by trial and test) are discarded, and then, as shown from Figure 5.4, all frames which contain the voiced region compressed back to the original file format and saved to the destination folder.



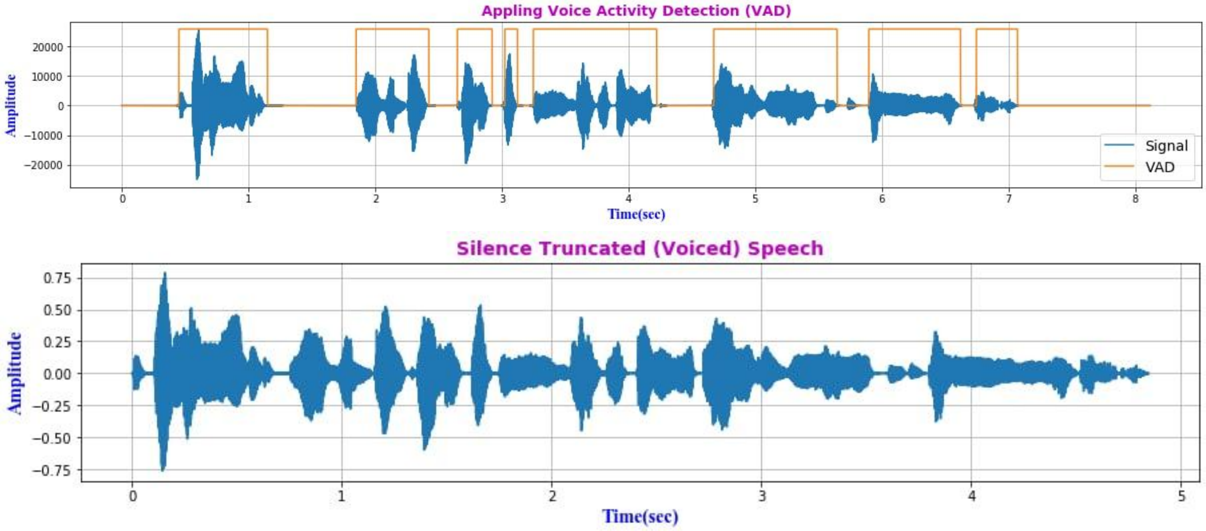


Figure 5.4: Short time based VAD implementation for silence truncation

Now, our speech files are ready to serve as an input for the feature extraction step.

5.4 Front-end Feature Extraction using MFCC

As discussed under section 4.3, MFCC can closely mimic the human auditory system, and outperformed than the other cepstral feature extraction techniques in speaker recognition operations. Thus, we have chosen MFCC as a front-end feature extraction technique. The process of extracting an acoustic feature vectors from audio files will implement following the procedure which was discussed under section 4.3.1.

Pre-emphasis: As described under 4.3.1.1, α is the pre-emphasis factor of equation (4.4), and its value ranges from 0.9 to 1. But, in most previous speaker recognition studies the typical value of α is taken as 0.97. So that we set this value for our case too.

Framing: As stated in section 4.3.1.2, a 20 to 40ms frame length is convenient in obtaining the quasi-stationary feature vectors of a speech from STFT. For our case, we used a frame size about 30ms, which is an average of the minimum and maximum intervals $(\frac{20+40}{2})$ ms and overlap about 50% of the frame length, which is 15ms i.e., the next frame shifting length is (30 -15) = 15ms.

Windowing: As described under section 4.3.1.3, Hamming window is the common windowing function in speaker recognition, and also it has a better combination with the MFCC feature extraction approach. Thus, once the speech signal is framed and overlapped, we have employed the Hamming window (which was presented in equation (4.5)) with frame size, (30ms) to retain a smooth transition between the consecutive frames through tapering to zero endings of the speech frames.

Fast Fourier Transform(FFT): During this step, each windowed frame will be transformed into the frequency domain using FFT and keeping the magnitude of the spectrum. As discussed in 4.3.1.4, FFT is the fast implementation of DFT. To do so, primarily, we have calculated the data point per frame, and set the size of the FFT point. From the corpora sampling frequency (f_s) and frame length, the data point per frame calculated as, 0.03*44100 = 1323 data points(samples) per frame. Then, to include this data point value during the FFT, we used 2048-point FFT with 725 zero padding. Then, this is step resulted a spectrum, $\tilde{y_i}[k]$ and its magnitude, $\tilde{y_k}$ of the speech frame using equations (4.7) and (4.8) respectively.

Mel-Frequency Warping: For the Hertz to Mel-scale conversion, the lower and higher frequencies used are f_{min} = 0Hz and f_{max} = 22050Hz. Then, to map the power spectrum value obtained from the preceding FFT step to mel scale values, a triangular filterbank with 40 filters, spaced 0.0 to 3,923.34 mel-scale (0, 2, 6, 9, 13, 17, 21, 26, 31, 37, 43, 50, 57, 65, 74, 83, 94, 105, 117, 130, 145, 160, 178, 196, 217, 239, 263, 289, 317, 349, 382, 419, 459, 503, 550, 602, 658,720, 786, 859, 938 and 1024) is used, and the resulted values were obtained from equation (4.9), inverse of equation of equation (4.9) and equation (4.10). Then, the filters' energy, $\tilde{E_k}$ computed with equation (4.12).

Discrete Cosine Transform (DCT): As stated under 4.3.6, in this step the vital intention is to extract the desired feature vectors, known as cepstral coefficients. And performing DCT using equation (4.12), is convenient to do so. In the feature extraction process, one vital concern is determining the suitable dimension (number of MFCC per frame) of the feature vector. We have determined dimension of the MFCC coefficients using the rule of thumb. As stated in [47], the rule of thumb regarding the number of coefficients says that coefficients up to $\left\lfloor \frac{f_s+\frac{f_s}{2}}{2} \right\rfloor$ are useful. Where f_s is the sampling frequency of the speech corpora in kHz. Thus, the dimension of the extracted feature vectors, D of $MFCC$ obtained as:

$$D = \left\lfloor \frac{f_s + \frac{f_s}{2}}{2} = \frac{3 * f_s}{4} \right\rfloor \tag{5.1}$$

As described in the corpora collection and preparation section, we sampled the utterances with 44.1kHz sampling rates for both the mobile phone and microphone recorded corpora. Thus, for the speech corpora with 44.1kHz, $D = \left\lfloor (\frac{3*44.1}{4}) \right\rfloor = 33MFCC$ coefficients per windowed frame.

The above computation helped us for obtaining 33-dimensional MFCC coefficient values from each frame of speech utterance files. As existed related works proved, for instance on [59], the first MFCC, $MFCC_0$ comprises insignificant speaker-specific information, thus we have discarded it. Hence, we used features $MFCC_1$ to $MFCC_{33}$.

ΔMFCC: As stated earlier under section 4.3.7, besides the standard static cepstral coefficients, the usage of delta (delta) coefficients as an appended feature vector helps to gain the finer details of the speech signal, and a significant improvement in the recognition performance. Hence, once we had been extracted the standard static MFCC coefficients, we have computed their corresponding first order, $MFCC$ coefficients, $MFCC_1$ to $MFCC_{33}$, and appended together to obtain the desired feature vector dimension per frame.

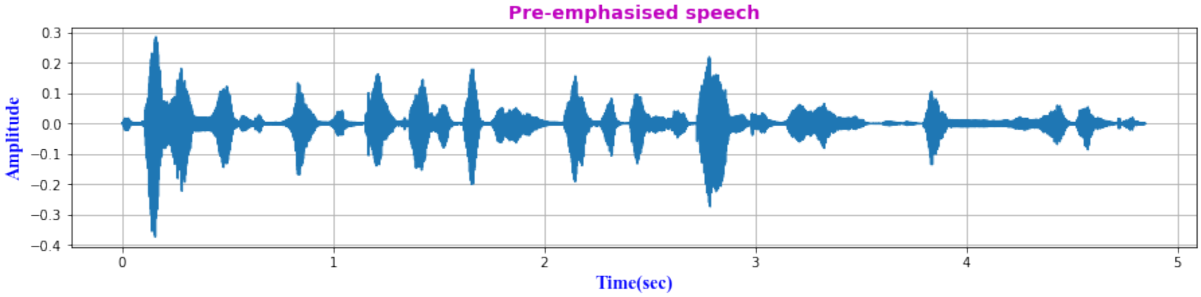
As we have seen from the framing step, we have taken a 30ms frame size with 50% overlapping duration. But the length of the corpora files is substantially larger than the frames. Thus, we have taken the means of the cepstral coefficients for each speech utterance. Figure 5.6, reveals the MFCC based feature extraction implementation procedures' results.

We have done the implementation of *MFCC* feature extraction using a python library module, *python_speech_features*[b]. As state under section 5.2 (*ii*), this library module enabled to extract *MFCC* acoustic feature vectors from the given speech utterance with a desired dimension. Below, the module's method definition presented with its parameters, and their respective default values:
python_speech_featuresḃase.mfcc(signal, samplerate=16000, winlen=0.025, winstep,0.01, num=13,nfilt=26, nfft=512, lowfreq=0, highfreq= samplerate/2, preemp=0.97, ceplifter= 22, appendEnergy = True/False, winfunc=<function<lambda»)

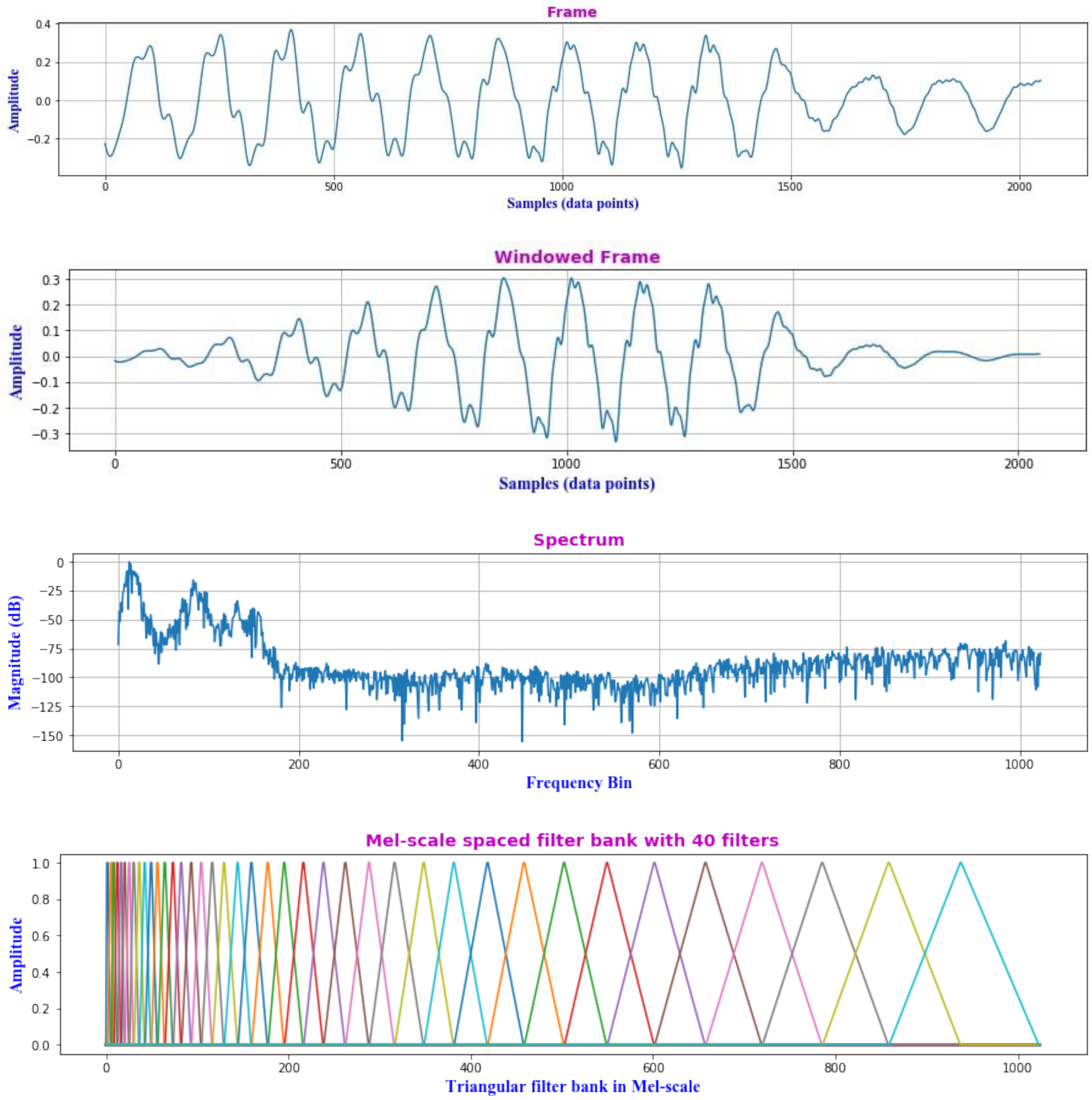
We used the above open-source package with parameter modifications as revealed below.

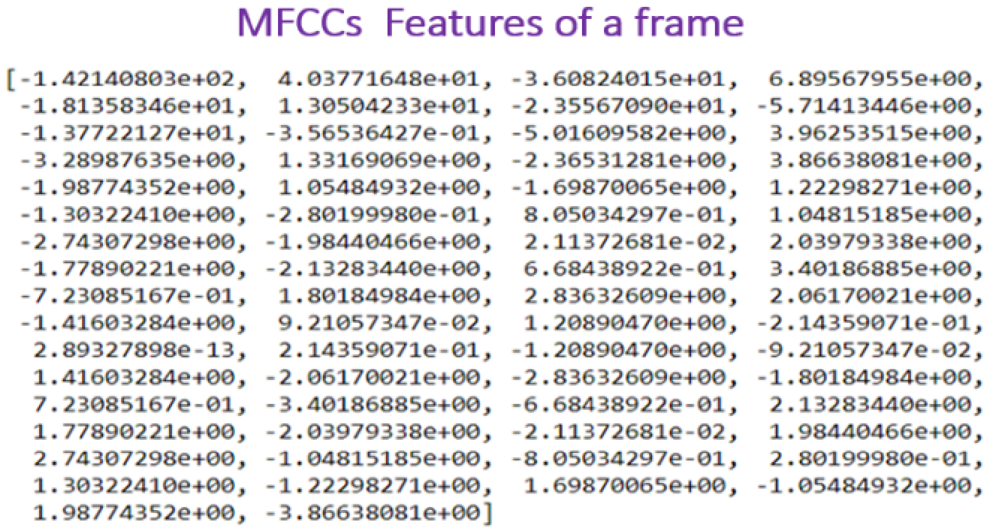
Table 5.2: Parameters of MFCC and their values used for our feature extraction

List of Parameters	Taken Values	
Sampling Frequency (samplerate)	44100Hz	
Window length (winlen)	30ms	
Overlapping length(winstep)	15ms	
Number of MFCC	33 per frame	*A 66-Dimension feature vector per frame*
Number of ΔMFCC	33 per frame	
Low frequency (lowfreq)	0Hz	
High frequency (highfreq)	22050Hz	
Number of FFT (nfft)	2048 sample points of FFT from each frame	
Number of filters (cepfilter)	40 filters	
Number of lifters (ceplifter)	34 lifters	
Window function (winfunc)	Hamming	
Pre-emphasis (preemph)	$\alpha = 0.97$	
AppendEnergy	False	



[b] Welcome to python_speech_features's documentation!, accessed on May 23, 2020

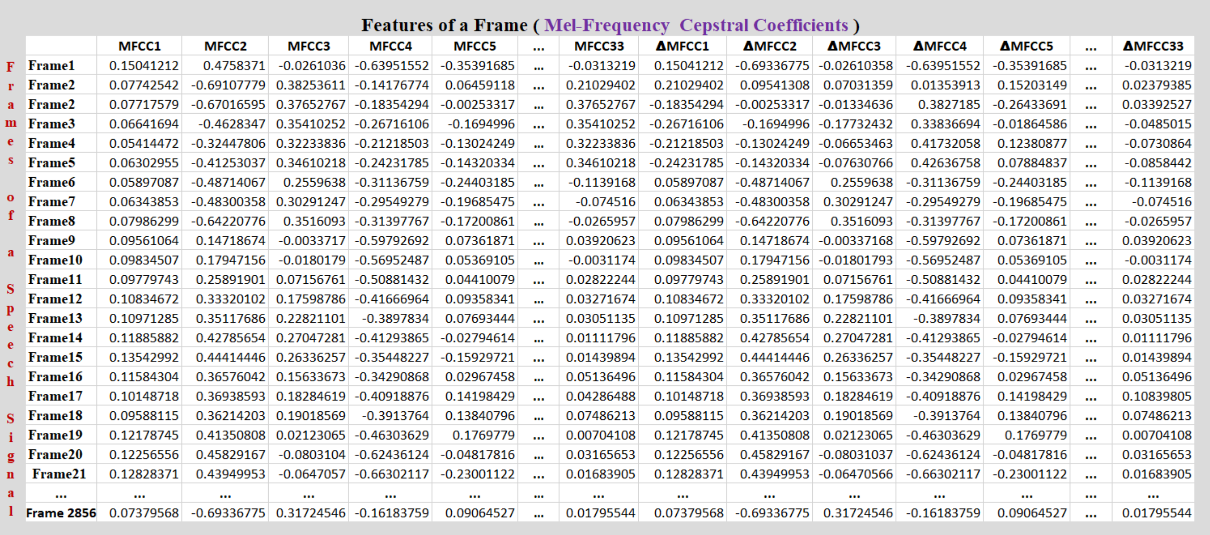



MFCCs Features of a frame

```
[-1.42140803e+02,  4.03771648e+01, -3.60824015e+01,  6.89567955e+00,
 -1.81358346e+01,  1.30504233e+01, -2.35567090e+01, -5.71413446e+00,
 -1.37722127e+01, -3.56536427e-01, -5.01609582e+00,  3.96253515e+00,
 -3.28987635e+00,  1.33169069e+00, -2.36531281e+00,  3.86638081e+00,
 -1.98774352e+00,  1.05484932e+00, -1.69870065e+00,  1.22298271e+00,
 -1.30322410e+00, -2.80199980e-01,  8.05034297e-01,  1.04815185e+00,
 -2.74307298e+00, -1.98440466e+00,  2.11372681e-02,  2.03979338e+00,
 -1.77890221e+00, -2.13283440e+00,  6.68438922e-01,  3.40186885e+00,
 -7.23085167e-01,  1.80184984e+00,  2.83632609e+00,  2.06170021e+00,
 -1.41603284e+00,  9.21057347e-02,  1.20890470e+00, -2.14359071e-01,
  2.89327898e-13,  2.14359071e-01, -1.20890470e+00, -9.21057347e-02,
  1.41603284e+00, -2.06170021e+00, -2.83632609e+00, -1.80184984e+00,
  7.23085167e-01, -3.40186885e+00, -6.68438922e-01,  2.13283440e+00,
  1.77890221e+00, -2.03979338e+00, -2.11372681e-02,  1.98440466e+00,
  2.74307298e+00, -1.04815185e+00, -8.05034297e-01,  2.80199980e-01,
  1.30322410e+00, -1.22298271e+00,  1.69870065e+00, -1.05484932e+00,
  1.98774352e+00, -3.86638081e+00]
```

Figure 5.5: Procedures of MFCC (pre-emphasis to ΔMFCC, see top to bottom)

Finally, through this entire feature extraction process, the original time-domain utterances are transformed into a 66-dimensional feature vector set per frame, appended from the 33$MFCCs$ and their corresponding first order 33$\Delta MFCC$ coefficients. And this used to train parameters of the GMM, and construct a reference model for the corresponding suspected speaker during the enrollment phase, or performing a feature matching (comparison) during the testing phase.



Features of a Frame (Mel-Frequency Cepstral Coefficients)

Frames of a Speech Signal

	MFCC1	MFCC2	MFCC3	MFCC4	MFCC5	...	MFCC33	ΔMFCC1	ΔMFCC2	ΔMFCC3	ΔMFCC4	ΔMFCC5	...	ΔMFCC33
Frame1	0.15041212	0.4758371	-0.0261036	-0.63951552	-0.35391685	...	-0.0313219	0.15041212	-0.69336775	-0.02610358	-0.63951552	-0.35391685	...	-0.0313219
Frame2	0.07742542	-0.69107779	0.38253611	-0.14176774	0.06459118	...	0.21029402	0.21029402	0.09541308	0.07031359	0.01353913	0.15203149	...	0.02379385
Frame2	0.07717579	-0.67016595	0.37652767	-0.18354294	-0.00253317	...	0.37652767	-0.18354294	-0.00253317	-0.01334636	0.3827185	-0.26433691	...	0.03392527
Frame3	0.06641694	-0.4628347	0.35410252	-0.26716106	-0.1694996	...	0.35410252	-0.26716106	-0.1694996	-0.17732432	0.33836694	-0.01864586	...	-0.0485015
Frame4	0.05414472	-0.32447806	0.32233836	-0.21218503	-0.13024249	...	0.32233836	-0.21218503	-0.13024249	-0.06653463	0.41732058	0.12380877	...	-0.0730864
Frame5	0.06302955	-0.41253037	0.34610218	-0.24231785	-0.14320334	...	0.34610218	-0.24231785	-0.14320334	-0.07630766	0.42636758	0.07884837	...	-0.0858442
Frame6	0.05897087	-0.48714067	0.2559638	-0.31136759	-0.24403185	...	-0.1139168	0.05897087	-0.48714067	0.2559638	-0.31136759	-0.24403185	...	-0.1139168
Frame7	0.06343853	-0.48300358	0.30291247	-0.29549279	-0.19685475	...	-0.074516	0.06343853	-0.48300358	0.30291247	-0.29549279	-0.19685475	...	-0.074516
Frame8	0.07986299	-0.64220776	0.3516093	-0.31397767	-0.17200861	...	-0.0265957	0.07986299	-0.64220776	0.3516093	-0.31397767	-0.17200861	...	-0.0265957
Frame9	0.09561064	0.14718674	-0.0033717	-0.59792692	0.07361871	...	0.03920623	0.09561064	0.14718674	-0.00337168	-0.59792692	0.07361871	...	0.03920623
Frame10	0.09834507	0.17947156	-0.0180179	-0.56952487	0.05369105	...	-0.0031174	0.09834507	0.17947156	-0.01801793	-0.56952487	0.05369105	...	-0.0031174
Frame11	0.09779743	0.25891901	0.07156761	-0.50881432	0.04410079	...	0.02822244	0.09779743	0.25891901	0.07156761	-0.50881432	0.04410079	...	0.02822244
Frame12	0.10834672	0.33320102	0.17598786	-0.41666964	0.09358341	...	0.03271674	0.10834672	0.33320102	0.17598786	-0.41666964	0.09358341	...	0.03271674
Frame13	0.10971285	0.35117686	0.22821101	-0.3897834	0.07693444	...	0.03051135	0.10971285	0.35117686	0.22821101	-0.3897834	0.07693444	...	0.03051135
Frame14	0.11885882	0.42785654	0.27047281	-0.41293865	-0.02794614	...	0.01111796	0.11885882	0.42785654	0.27047281	-0.41293865	-0.02794614	...	0.01111796
Frame15	0.13542992	0.44414446	0.26336257	-0.35448227	-0.15929721	...	0.01439894	0.13542992	0.44414446	0.26336257	-0.35448227	-0.15929721	...	0.01439894
Frame16	0.11584304	0.36576042	0.15633673	-0.34290868	0.02967458	...	0.05136496	0.11584304	0.36576042	0.15633673	-0.34290868	0.02967458	...	0.05136496
Frame17	0.10148718	0.36938593	0.18284619	-0.40918876	0.14198429	...	0.04286488	0.10148718	0.36938593	0.18284619	-0.40918876	0.14198429	...	0.10839805
Frame18	0.09588115	0.36214203	0.19018569	-0.3913764	0.13840796	...	0.07486213	0.09588115	0.36214203	0.19018569	-0.3913764	0.13840796	...	0.07486213
Frame19	0.12178745	0.41350808	0.02123065	-0.46303629	0.1769779	...	0.00704108	0.12178745	0.41350808	0.02123065	-0.46303629	0.1769779	...	0.00704108
Frame20	0.12256556	0.45829167	-0.0803104	-0.62436124	-0.04817816	...	0.03165653	0.12256556	0.45829167	-0.08031037	-0.62436124	-0.04817816	...	0.03165653
Frame21	0.12828371	0.43949953	-0.0647057	-0.66302117	-0.23001122	...	0.01683905	0.12828371	0.43949953	-0.06470566	-0.66302117	-0.23001122	...	0.01683905
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
Frame 2856	0.07379568	-0.69336775	0.31724546	-0.16183759	0.09064527	...	0.01795544	0.07379568	-0.69336775	0.31724546	-0.16183759	0.09064527	...	0.01795544

Figure 5.6: A 66-D MFCC feature vector (from suspected 19)

5.5 Back-end Feature Classification using GMM

As discussed under sections 4.4.1 and 4.4.2, our back-end feature classification stage consisted of suspects modeling and criminal identification operations.

5.5.1 Modeling the Suspects using EM Algorithm

Following extracting the acoustic feature vector from the preprocessed speech corpora, the proposed system trained to build a reference model for each suspect. i.e. During the enrollment phase, the modeling algorithm takes the feature vectors extracted from speech signals and generates model to represent each individual.

As described in the design chapter, in GMM based model training process, the main task is estimating the parameters' value. In our training process, the GMM parameters estimated using an iterative expectation maximization(EM) algorithm from the training data point. We have used k-means clustering technique to initialize the parameters. To minimize the training complexity, we have used a diagonal covariance matrix (each component with its diagonal matrix). But before initializing and estimating the Gaussian distribution parameters' value, determining the optimum modeling order (M) of a GMM mixture is a crucial task to model a given speaker adequately .

As [76], there is no a standard approach to determine the optimum number of mixture components a priori. For speaker modeling, the objective is to choose the minimum number of components necessary to adequately model a speaker for good speaker identification. Choosing too few mixture components can produce a speaker model that does not accurately model the distinguishing characteristics of a speaker's distribution. Choosing too many components can reduce performance when there are a large number of model parameters relative to the available training data and can also result in excessive computational complexity both in training and identification.

Thus, preceding research works determined experimentally, and we too enforced to follow that approach to determine the optimum number of mixture components. We will perform an experiment to determine the optimum number of mixture for the respective level of utterances in the next chapter, preceding to the other experiments.

As stated in section 5.2, *sklearn* is an open-source python library. It enables to train the GMM's parameters, and estimate their maximum-likelihood distribution. So that, we used this library package for training the proposed system and model the suspects.

A model is built for each suspect by using feature vectors extracted from his or her speech signal at the previous stage. Each model is generated and stored in the form of .gmm file format. Finally, at the end of the modeling process, each suspect will have a single voiceprint reference model in the database of models to represent that particular individual uniquely, and later on, these models are used to identify the unknown criminal speaker during the identification phase. Table 5.3 depicts modeling characteristics and their description used for our training task. The equations given from (4.18) to (4.22) used to perform the training operation as of the steps presented on Figure 4.12.

Table 5.3: GMM parameters and used techniques for modeling the supects

List of parameters	Description
Number of modeling orders	Determined experimentally
Co-variance matrix type	Diagonal
Model parameters	$\lambda^{(m)} = \{\overrightarrow{w}_m, \overrightarrow{\mu}_m, \overrightarrow{\Sigma}_m\}$
Parameter initialization technique	k-means clustering
Training algorithm	Expectation Maximization(EM)
Convergence threshold (ϵ)	$(\lambda^{(m+1)} - \lambda^{(m)}) < 0.001$

5.5.2 Identifying the Criminal using MLL Score

This is the testing phase of the proposed system. At this moment the trace evidence of the unknown criminal speaker is compared to all the known suspected speakers' reference model which has been built during the training phase. The log-likelihood for each model of every suspected speaker was calculated in the model training phase and stored as a database. So, in this identification or testing phase, the stored models are used for matching, and the system makes a decision about the identity of the criminal through comparing the unknown criminal speaker's utterance (trace evidence) acoustic feature vector with all models in the database, and select the best matching model, i.e. the suspected speaker with the highest log-likelihood score would be identified as an actual criminal among modeled suspects. This has been done using equation (4.25).

5.6 Performance Evaluating Metrics

Once the proposed system is designed, during testing, the effectiveness of that system measured with performance metric(s). For speaker recognition, there are many measurement metrics, but they are slightly different for different types of speaker recognition. Detection Error Trade-offs (DET) curve, Equal Error Rate (EER) and Detection Cost Function (DCF) are common evaluation metrics for speaker verification and open-set speaker identification tasks while identification rate (IDR) is the most widely used evaluation metric for a close-set speaker identification systems[8].

IDR is referred to the expected proportion that the test utterances are correctly identified from the set of enrolled speakers. For each test utterance, the speaker achieving the highest score among the set of enrolled speakers is regarded as an identified speaker. And where U_c and U_i represented the number of correctly and incorrectly classified utterances among the given total testing trials respectively, the identification rate (IDR) can be define as[7]:

$$IdentificationRate(IDR) = \frac{U_c}{U_c + U_i} \tag{5.2}$$

As of [60], accuracy (also known as traditional accuracy) is one evaluation metric of Machine Learning models. And it measures the classifier's performance by determining the ratio of the correct number of trials, which are achieved by the classifier over the total number of trials. The formula for accuracy is defined as:

$$Accuracy(A) = \frac{TP + TN}{TP + TN + FN + FP} \tag{5.3}$$

While considering a binary class classification, equation (5.3) can be taken as a very good evaluation measurement definition of IDR. But, while taking into account a multi-class classification, equation (53) is not considered correct as the True Negatives(TN) are 'true' only from the perspective of one class. I.e, a TN is an incorrectly classified instance and cannot be considered as the right classification. Thus, the idea of IDR, defined on equation (5.2), can be redefined in a more vivid way as classification accuracy(CA)[60]:

$$ClassificationAccuracy(CA) = \frac{TP}{TP + TN + FN + FP} \tag{5.4}$$

As it can be observe from equation(5.4), in classification accuracy only the true positive values(TP) are considered in computing the system's IDR. When there is a balanced number of corpora per class, classification accuracy is a very good metric in evaluating multi-class classification model[60].

Confusion matrix is a 2-D matrix used to describe the overall performance of the given model. The rows and columns of a matrix are marked by the classes. The diagonal cells correspond to observations that are correctly identified and represent the true positives in a multi- class confusion matrix. The off-diagonal cells correspond to incorrectly identified observations. True positive (TP), the correct identification of a speaker, False-positive (FP), the incorrect identification of a speaker, True negative(TN), the correct identification of the incorrect speaker and False-negative (FN), the incorrect identification of the incorrect speaker are the four possible outcomes of a confusion matrix. As shown in equations (5.3) and (5.4), to measure performance of the given model, the evaluation metrics computation relies on the four possible outcomes.

CHAPTER 6

EXPERIMENTAL RESULTS AND DISCUSSION

In this chapter experimental results presented with their corresponding discussion. To evaluate performance of the proposed forensic speaker identification system, five experiments were done at word-level utterances (WLU), sentence-level utterances (SLU) and paragraph-level utterances (PLU) of speech corpora. The experiments were carried out to examine the impact of modeling order of GMM, level of the utterance of training and testing corpora, crossed levels of utterance between training and testing corpora, recording device mismatch between training and testing corpora and population size over an identification performance.

6.1 Experimental Setup

As described earlier, we have carried out five experiments. The first experiment (see Experiment 1 under section 6.2) has been done to observe the impact of modeling order(number of mixture components of GMM), and determine the optimum modeling order for the three level of utterances (WLU, SLU and PLU). The second experiment(see Experiment 2.1 to 2.3 under section 6.2) has been carried out to see the impact of corpora level of utterance over identification rate, and determine the preferable level of utterances of the training-testing corpora. The third experiment (see Experiment 3 under section 6.2) has been performed to observe the impact of crossed level training-testing utterance. The subsequent fourth experiment (see Experiment 4 under section 6.2) has been done to observe the impact of multi-modality (recording device mismatch for training-testing scenario) over IDR. The last experiment (see Experiment under section 6.2) has been conducted to see the impact of population size (for our case, number of suspects) over IDR.

The experiments performed according to the parameters descriptions and configurations as presented under design and implementation sections, chapter four and five respectively. That means, The experiments have been carried out based on the MFCC feature extraction procedures as presented under section 4.3.1, Figure 4.7 with parameters configuration given under section 5.4, Table 5.2. While the feature classification operation performed based on GMM through EM procedures as presented under section 4.4.1, Figure 4.12 with parameters configuration given under section 5.5, Table 5.3.

The experiments have been done for three (word, sentence and paragraph)level of utterances with the training-testing ration given under section 5.2, Table 5.1 and Figure

5.2. The training-testing corpora set have been preprocessed as of the techniques given under sections 4.2.1 and 4.2.2 with parameters configuration given under sections 5.3.1 and 5.3.2. As discussed under section 5.6, the identification rate(IDR) and classification accuracy are the same concepts, and they are convenient metrics to evaluate performance of a close-set speaker identification system under balanced number of testing corpora per class. Thus, we evaluated our model using the IDR metric. As described under section 5.1, our experiments have been done using python on Pycharm environment.

6.2 Experimental Results

Experiment 1: The impact of modeling order of GMM on identification performance

As stated under section 5.5.1, to model the given speaker adequately using his or her acoustic feature vector, determining the optimum modeling order (number of mixture components), M of a GMM is one vital task; but it is a difficult problem yet. Because there is no objective way to determine the optimum modeling order prior to the training.

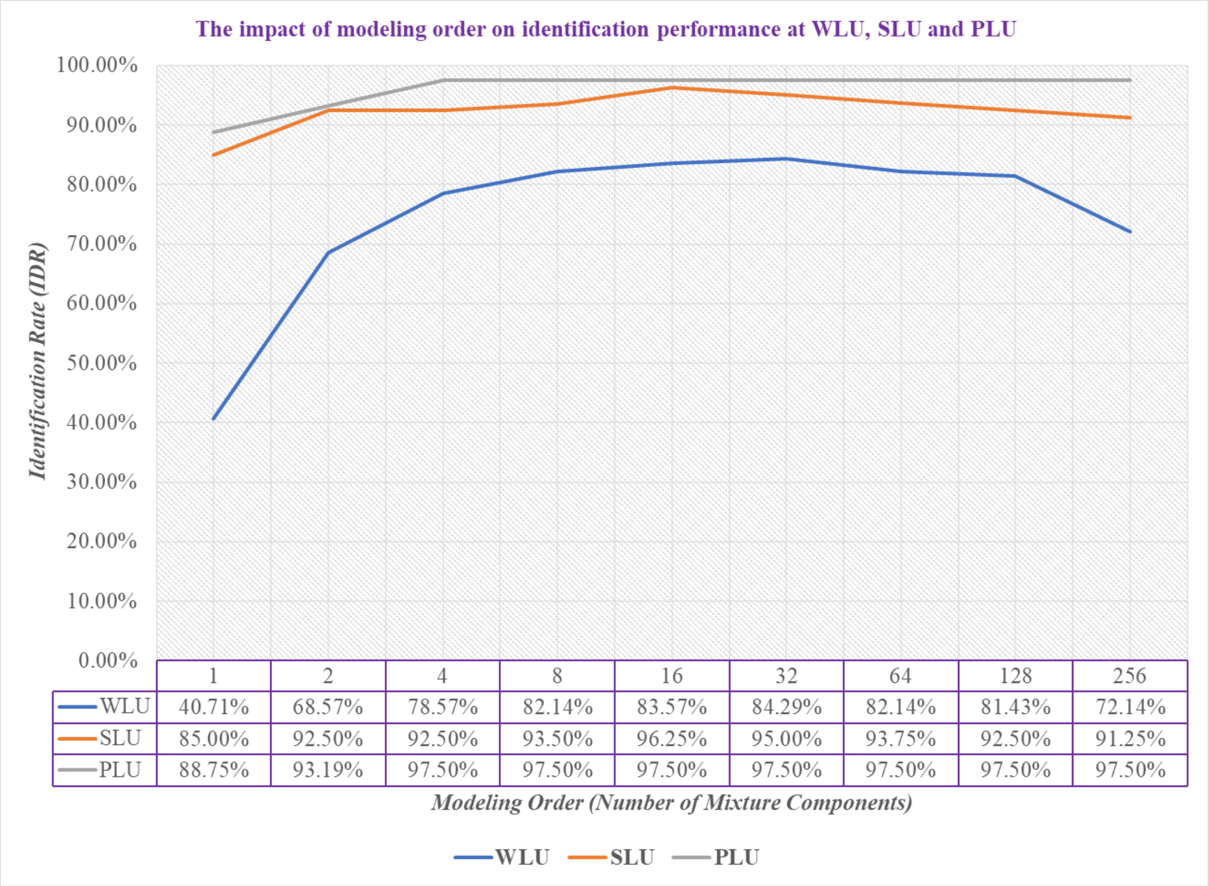
In GMM based speaker modeling task, the aim is to select a minimum number of modeling orders that enable to adequately model a speaker for better recognition performance. As [76], choosing too few modeling orders can produce a speaker model that does not accurately model the distinguishing characteristics of a speaker's distribution. On the contrary, choosing too many modeling orders relative to limited training data can induce many vacant parameters to be modeled, and this would be caused to degrade performance. Thus, the number of modeling order has to be many enough to provide good model but few enough to avoid computational complications in both the training and testing stages. As well the modeling order has a direct impact on the time needed for training and testing. This inferred that the smallest modeling order that provide acceptable results should be taken, and the feasible way is determining experimentally.

For this thesis work implementation, knowing the optimum modeling order for each level of utterances helps us to perform the other experiments in the proper ways with the aim of attaining a better identification performance for each level of utterances. Hence, to determine the optimum modeling order for the three levels of utterances, we have experimented with three levels of utterance separately on our speech corpora with different number of modeling orders. The experiments performed repeatedly for an increasing number of modeling orders (M) with power of 2. Table 6.1 and Figure 6.1, reveal the IDR versus modeling orders of the GMMM at WLU, SLU, and PLU training-testing utterances with training and testing duration for each respective experiment.

As mentioned earlier, the above experiments carried out to determine the optimum modeling order for WLU, SLU and PLU levels of utterance. We have done 9-executions per level of utterance (27-executions for the three levels of utterance), and from the obtained

Table 6.1: The impact of modeling order of GMM over Identification performance

M	Average Identification Rate (IDR) at								
	WLU			SLU			PLU		
	Duration (second)		IDR	Duration (second)		IDR	Duration (second)		IDR
	Training	Testing		Training	Testing		Training	Testing	
1	20.17	146.33	40.71%	29.03	96.12	85.00%	63.15	117.45	88.75%
2	23.47	146.68	68.57%	36.79	92.42	92.50%	109.33	112.86	93.19%
4	39.81	149.24	78.57%	45.28	90.94	92.50%	144.56	113.09	**97.50%**
8	71.95	147.80	82.14%	61.29	92.83	93.500%	250.97	114.63	97.50%
16	94.83	148.53	83.57%	97.98	91.17	**95.00%**	443.14	115.43	97.50%
32	180.45	149.27	84.29%	165.76	91.74	95.00%	752.48	121.09	97.50%
64	336.55	147.04	82.14%	336.18	92.83	93.75%	1447.64	145.43	97.50%
128	480.67	147.43	81.43%	607.53	97.19	92.50%	3572.91	151.54	97.50%
256	1015.74	147.56	72.14%	1012.27	99.53	91.25%	6200.79	192.45	97.50%



	1	2	4	8	16	32	64	128	256
WLU	40.71%	68.57%	78.57%	82.14%	83.57%	84.29%	82.14%	81.43%	72.14%
SLU	85.00%	92.50%	92.50%	93.50%	96.25%	95.00%	93.75%	92.50%	91.25%
PLU	88.75%	93.19%	97.50%	97.50%	97.50%	97.50%	97.50%	97.50%	97.50%

Figure 6.1: The impact of modeling order of GMM over IDR

results illustration (Table 6.1 and Figure 6.1), the proposed system achieved its maximum IDR at different modeling orders for the three levels of utterance. i.e. for the WLU experiment, when the modeling order(M) set as 1, 2,4,8,16 and 32, the system revealed an increment in identification accuracy with various incremental rates, and the system scored its maximum IDR as 84.29%, when the GMM modeling order set to 32. Afterward, when the modeling order exceeded 32, the system began to decline its IDR gradually. Similarly, for the SLU experiment, when the number of components set as 1, 2, 4, 8 and 16 the system revealed an increment in IDR with various incremental rates. The system scored its maximum IDR as 95.00%, when the GMM modeling order was set to 16 and 32. Afterward, when the modeling order exceeded 32, the system began to decline its IDR with slower rates compared to WLU's case. On the other hand, for the PLU experiment, when modeling order set as 1, 2 and 4, the system revealed an increment in IDR with various incremental rates. The system scored its maximum IDR as 97.50% for the modeling orders of 4, 8,16, 32, 64,128 and 256.

Relying on the above experimental results, we preferred to select the minimum modeling order with maximum accuracy for the respective level of utterance at the expense of training and testing duration. We have selected 32 (see from Table 6.1, which is marked with blue), 16 (see from Table 6.1, which is marked with red), and 4 (see from Table 6.1, which is marked as green), modeling order of GMM for WLU, SLU and PLU respectively. Thus, to attain a better IDR performance for the next experiments of the given level of utterances, we will employ their respective modeling order to WLU, SLU and PLU.

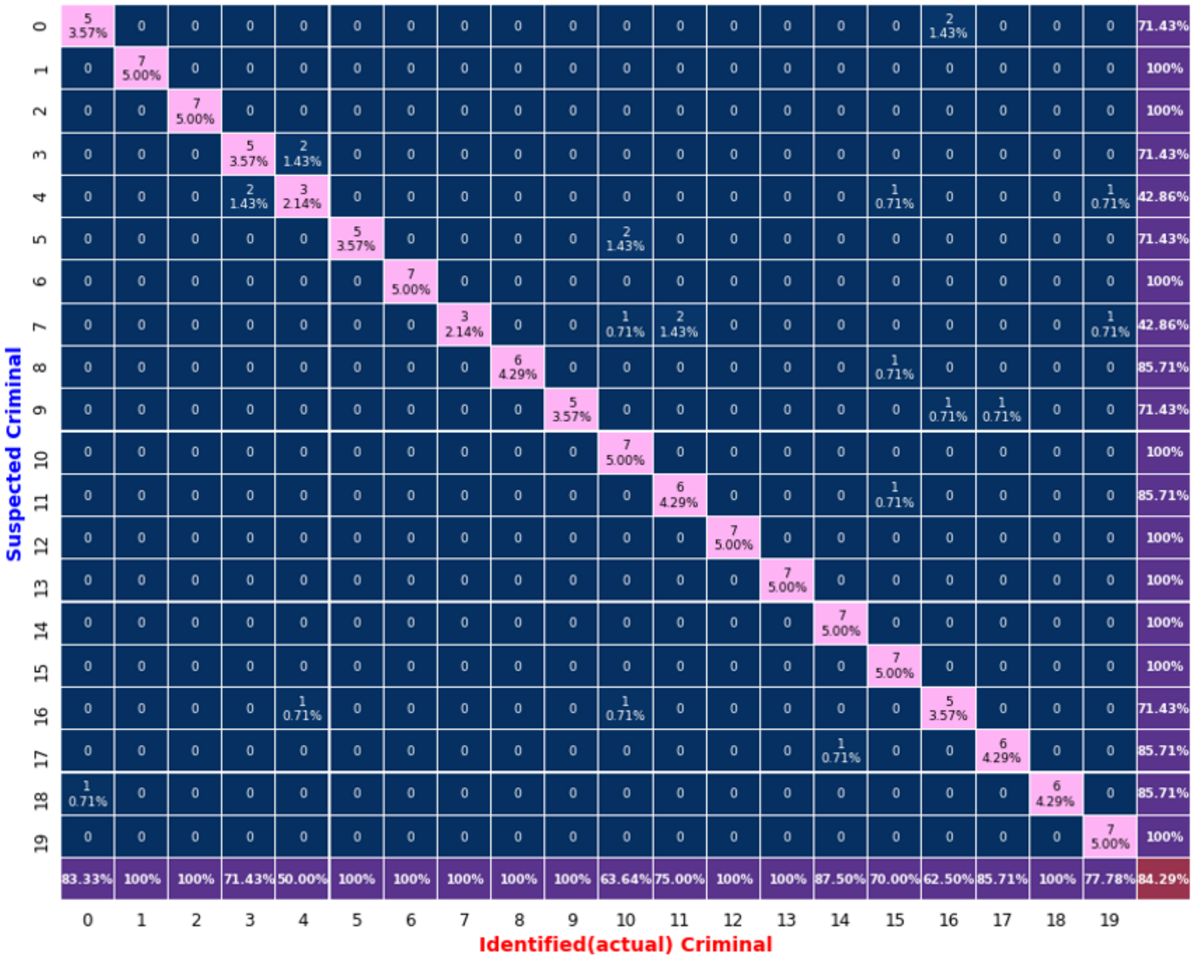
Experiment 2: The impact of levels of utterance on identification performance

Here, we have done three experiments (using word, sentence and paragraph levels of utterances for training and testing.) to observe the impact of the level(length) of utterance, and to determine the optimum level of utterance for forensic speaker recognition applications. We used 32, 16 and 4 modeling orders for WLU,SLU and PLU respectively. For the respective level of utterance an experiment was conducted as follow:

Experiment 2.1: Identification Rate of the proposed system at WLU

As designated under section 5.2 (see Figure 5.2), for this experiment, we used 40- world level utterances (WLU) for each individual of the 20 suspects. On the experiment, 33 of the total word-level utterances were used during the enrollment phase to train the system and build a reference model for the given suspect. Once the model is built and stored, the rest unseen 7-word level utterances are used during the identification as the testing trace. i.e. for each suspect, we made 7-trials to measure the identification rate of system. Figure 6.2 reveals an average IDR for the respective suspected among the total trials of the testing process and overall identification performance.

From the confusion matrix given in Figure 6.2, the diagonal cells illustrate the number of testing trace utterances that were correctly identified their respective suspect as an criminal. While the off-diagonal cells illustrate the number of testing trace utterances that were wrongly identified(number of confused or wrong prediction for the given suspect identified as an criminal). On the off-diagonal, empty (zero value) cell means no confusion has occurred for the given particular suspected to be identified as criminal. The column on the last right reveals the percentages of how many trace utterances correctly identified from the total trace utterances of the given suspected. For example, the fifth suspected, Elbetel (Suspect 4) has been identified 42.86% $[\frac{3*100}{7}]$ correctly, and 57.14% (28.57% +14.285% +14.285%) confused with Desta(suspected 3), Wubrist (suspected 19) and



Suspected Criminal

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	
0	5 3.57%	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2 1.43%	0	0	0	71.43%
1	0	7 5.00%	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100%
2	0	0	7 5.00%	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100%
3	0	0	0	5 3.57%	2 1.43%	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	71.43%
4	0	0	0	2 1.43%	3 2.14%	0	0	0	0	0	0	0	0	0	0	1 0.71%	0	0	0	1 0.71%	42.86%
5	0	0	0	0	0	5 3.57%	0	0	0	0	2 1.43%	0	0	0	0	0	0	0	0	0	71.43%
6	0	0	0	0	0	0	7 5.00%	0	0	0	0	0	0	0	0	0	0	0	0	0	100%
7	0	0	0	0	0	0	0	3 2.14%	0	0	1 0.71%	2 1.43%	0	0	0	0	0	0	0	1 0.71%	42.86%
8	0	0	0	0	0	0	0	0	6 4.29%	0	0	0	0	0	0	1 0.71%	0	0	0	0	85.71%
9	0	0	0	0	0	0	0	0	0	5 3.57%	0	0	0	0	0	0	1 0.71%	1 0.71%	0	0	71.43%
10	0	0	0	0	0	0	0	0	0	0	7 5.00%	0	0	0	0	0	0	0	0	0	100%
11	0	0	0	0	0	0	0	0	0	0	0	6 4.29%	0	0	0	1 0.71%	0	0	0	0	85.71%
12	0	0	0	0	0	0	0	0	0	0	0	0	7 5.00%	0	0	0	0	0	0	0	100%
13	0	0	0	0	0	0	0	0	0	0	0	0	0	7 5.00%	0	0	0	0	0	0	100%
14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	7 5.00%	0	0	0	0	0	100%
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	7 5.00%	0	0	0	0	100%
16	0	0	0	0	1 0.71%	0	0	0	0	0	1 0.71%	0	0	0	0	0	5 3.57%	0	0	0	71.43%
17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1 0.71%	0	0	6 4.29%	0	0	85.71%
18	1 0.71%	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	6 4.29%	0	85.71%
19	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	7 5.00%	100%
	83.33%	100%	100%	71.43%	50.00%	100%	100%	100%	100%	100%	63.64%	75.00%	100%	100%	87.50%	70.00%	62.50%	85.71%	100%	77.78%	84.29%

Identified(actual) Criminal

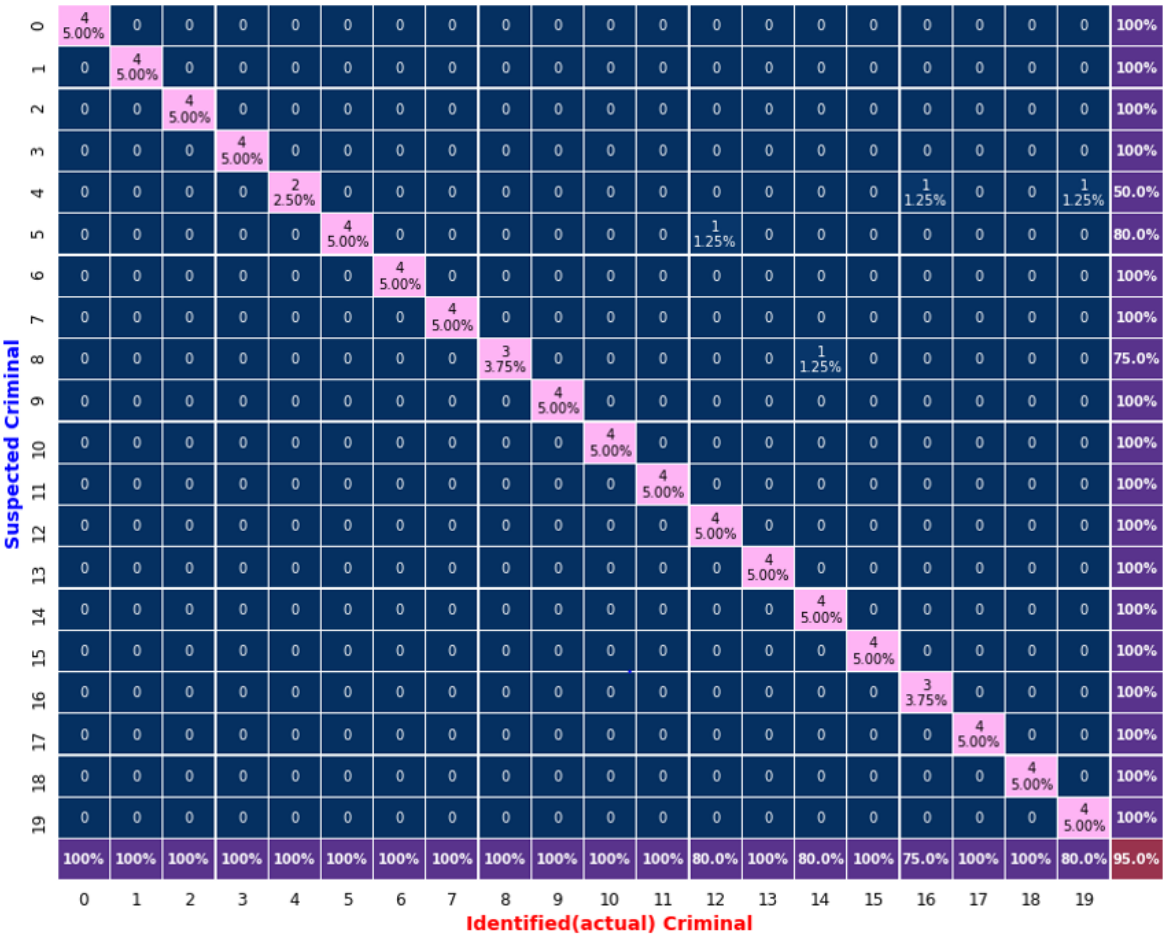
Figure 6.2: Confusion Matrix for WLU testing

Msganaw (suspected 15). The red cell in the bottom right illustrates the model's overall IDR of the proposed system . Finally, as it could be observed from the obtained result, the proposed system achieved 84.29% overall IDR at WLU.

Experiment 2.2: Identification Rate of the proposed system at SLU

As designated under section 5.2 (see Figure 5.2), for this experiment, we used 10-sentence level utterances for every individual of the 20 suspects. 6 of the total utterances used for the enrollment phase to train the system and build a reference model for each suspected. While the rest unseen 4- sentence level utterances were used during the identification phase as the testing trace; for each suspected made 4-trials to evaluate the proposed system identification performance. Figure 6.3 reveals the average IDR for the respective suspects among the total trials of the testing process and the overall identification performance of the proposed system using SLU.

From the confusion matrix given in Figure 6.3, the diagonal cells illustrate the number of testing trace utterances that were correctly identified their respective suspected criminal speaker as a criminal. While the off-diagonal cells illustrate the number of testing trace utterances that were wrongly identified(number of confused or wrong prediction for the given suspected identified as a criminal). On the off-diagonal, empty (zero value) cell



Suspected Criminal

Identified(actual) Criminal

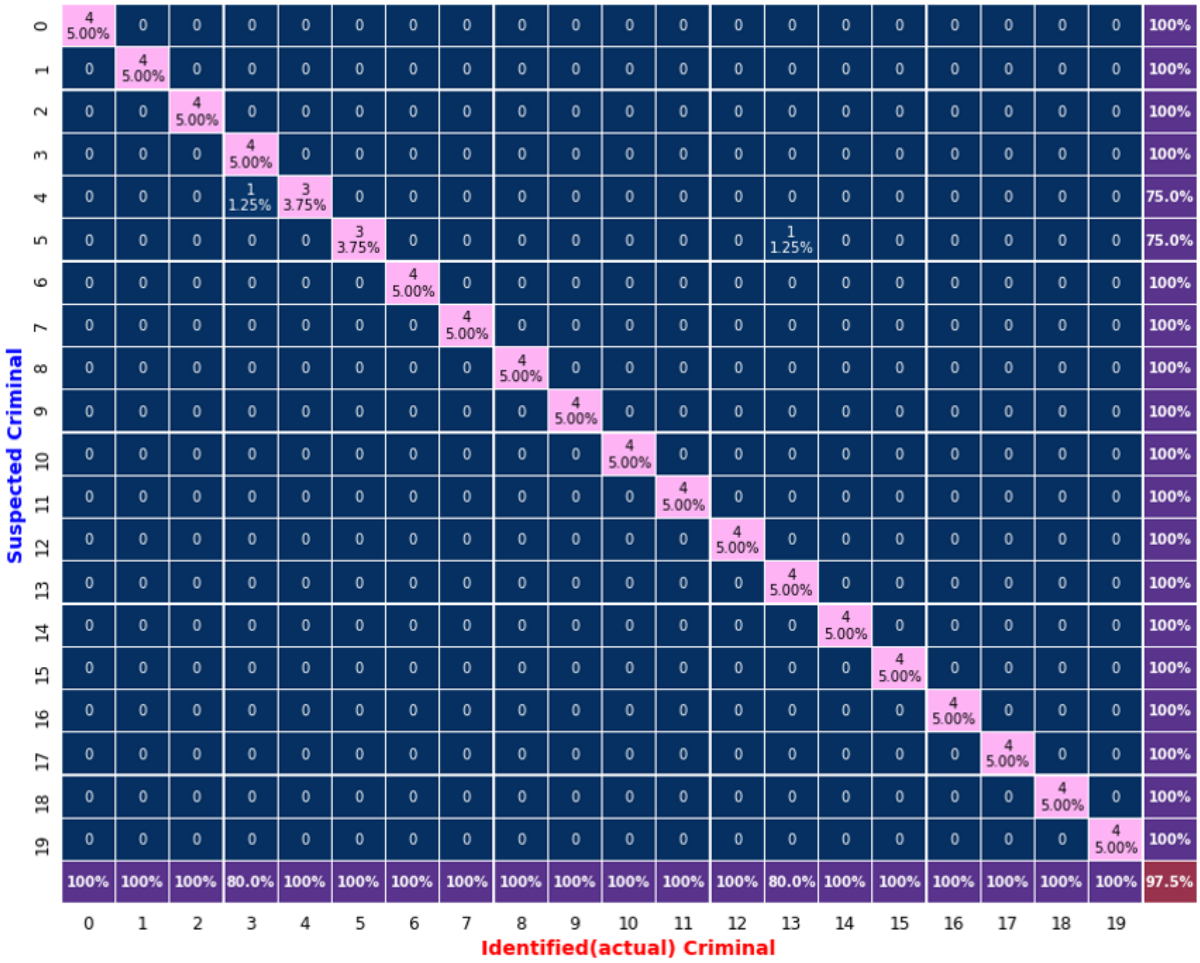
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	
0	4 5.00%	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100%
1	0	4 5.00%	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100%
2	0	0	4 5.00%	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100%
3	0	0	0	4 5.00%	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100%
4	0	0	0	0	2 2.50%	0	0	0	0	0	0	0	0	0	0	0	1 1.25%	0	0	1 1.25%	50.0%
5	0	0	0	0	0	4 5.00%	0	0	0	0	0	0	1 1.25%	0	0	0	0	0	0	0	80.0%
6	0	0	0	0	0	0	4 5.00%	0	0	0	0	0	0	0	0	0	0	0	0	0	100%
7	0	0	0	0	0	0	0	4 5.00%	0	0	0	0	0	0	0	0	0	0	0	0	100%
8	0	0	0	0	0	0	0	0	3 3.75%	0	0	0	0	0	1 1.25%	0	0	0	0	0	75.0%
9	0	0	0	0	0	0	0	0	0	4 5.00%	0	0	0	0	0	0	0	0	0	0	100%
10	0	0	0	0	0	0	0	0	0	0	4 5.00%	0	0	0	0	0	0	0	0	0	100%
11	0	0	0	0	0	0	0	0	0	0	0	4 5.00%	0	0	0	0	0	0	0	0	100%
12	0	0	0	0	0	0	0	0	0	0	0	0	4 5.00%	0	0	0	0	0	0	0	100%
13	0	0	0	0	0	0	0	0	0	0	0	0	0	4 5.00%	0	0	0	0	0	0	100%
14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4 5.00%	0	0	0	0	0	100%
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4 5.00%	0	0	0	0	100%
16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3 3.75%	0	0	0	100%
17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4 5.00%	0	0	100%
18	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4 5.00%	0	100%
19	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4 5.00%	100%
	100%	100%	100%	100%	100%	100%	100%	100%	100%	100%	100%	100%	80.0%	100%	80.0%	100%	75.0%	100%	100%	80.0%	95.0%

Figure 6.3: Confusion Matrix for SLU testing

means no confusion has occurred for the given particular suspected to be identified as criminal. The column on the last right reveals the percentages of how many trace utterances correctly identified from the total trace utterances of the given suspected. For example, the fifth suspected, Elbetel (Suspect 4) has been identified 50.0% [$\frac{2*100}{4}$] correctly,and 50.0% (25.0%+25.0%) confused with Rediet (Suspect 16) and Wubrist (Suspect 19). The red cell in the bottom right illustrates the model's overall IDR of the proposed system. Finally, as it could be observed from the obtained result, the proposed system achieved 95.0% overall IDR at SLU.

Experiment 2.3: Identification Rate of the proposed system at PLU

This experiment conducted using 10-paragraph level utterances for each individual of the 20 suspects. 6 of the total paragraph-level utterances used during the enrollment phase to train the system and build a reference model for every suspect. Once the model is built, to test the system the rest unseen 4-paragraph level utterances are used during the identification phase as the testing trace. Which means, for each suspected made 4-trials. Figure 6.4 reveals the average IDR for a suspected among the total trials of the testing process and the overall performance of the proposed system at PLU.



	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	
0	4 5.00%	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100%
1	0	4 5.00%	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100%
2	0	0	4 5.00%	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100%
3	0	0	0	4 5.00%	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100%
4	0	0	0	1 1.25%	3 3.75%	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	75.0%
5	0	0	0	0	0	3 3.75%	0	0	0	0	0	0	0	1 1.25%	0	0	0	0	0	0	75.0%
6	0	0	0	0	0	0	4 5.00%	0	0	0	0	0	0	0	0	0	0	0	0	0	100%
7	0	0	0	0	0	0	0	4 5.00%	0	0	0	0	0	0	0	0	0	0	0	0	100%
8	0	0	0	0	0	0	0	0	4 5.00%	0	0	0	0	0	0	0	0	0	0	0	100%
9	0	0	0	0	0	0	0	0	0	4 5.00%	0	0	0	0	0	0	0	0	0	0	100%
10	0	0	0	0	0	0	0	0	0	0	4 5.00%	0	0	0	0	0	0	0	0	0	100%
11	0	0	0	0	0	0	0	0	0	0	0	4 5.00%	0	0	0	0	0	0	0	0	100%
12	0	0	0	0	0	0	0	0	0	0	0	0	4 5.00%	0	0	0	0	0	0	0	100%
13	0	0	0	0	0	0	0	0	0	0	0	0	0	4 5.00%	0	0	0	0	0	0	100%
14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4 5.00%	0	0	0	0	0	100%
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4 5.00%	0	0	0	0	100%
16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4 5.00%	0	0	0	100%
17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4 5.00%	0	0	100%
18	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4 5.00%	0	100%
19	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4 5.00%	100%
	100%	100%	100%	80.0%	100%	100%	100%	100%	100%	100%	100%	100%	100%	80.0%	100%	100%	100%	100%	100%	100%	97.5%

Figure 6.4: Confusion Matrix for PLU testing

From Figure 6.4, the diagonal cells illustrate the number of testing trace utterances that were correctly identified their respective suspected as a criminal. While the off-diagonal cells illustrate the number of testing trace utterances that were wrongly identified(number of confused or wrong prediction for the given suspected identified as a criminal). On the off-diagonal, empty (zero value) cell means no confusion has occurred for the given particular suspected criminal to be identified as a criminal. The column on the last right reveals the percentages of how many trace utterances correctly identified from the total trace utterances of the given suspected. For example, the first suspected, Abeba (Suspect 0) has been identified 100.0% [$\frac{4*100}{4}$] perfectly,and no confused with any other suspects. The red cell in the bottom right illustrates the model's overall IDR of the proposed system. Finally, as it could be observed from the obtained result, the proposed system achieved 97.50% overall IDR at PLU.A sample of the individual's full identification trial result was provided in the appendix section (under appendix C).

As we have seen from the above three experiments (Figure 6.2 to Figure 6.4), we have achieved different identification performances for the three levels of utterances corpora. And from the interpretation of the results, the system's identification performance revealed a direct affiliation with the levels of utterances of the corpora; which means that, to identify each suspected uniquely the system performed better on models that were obtained from long level utterances. i.e. the proposed system performed well on models gained from sentence-level utterances than word-level utterances, and as well for models obtained from paragraph-level utterances than sentence and word level utterances of the individual suspected speaker. To reveal this vividly, Table 6.2 and Figure 6.5 presented the result summary of the previous three experiments, (from Experiment 2.1 to Experiment 2.3).

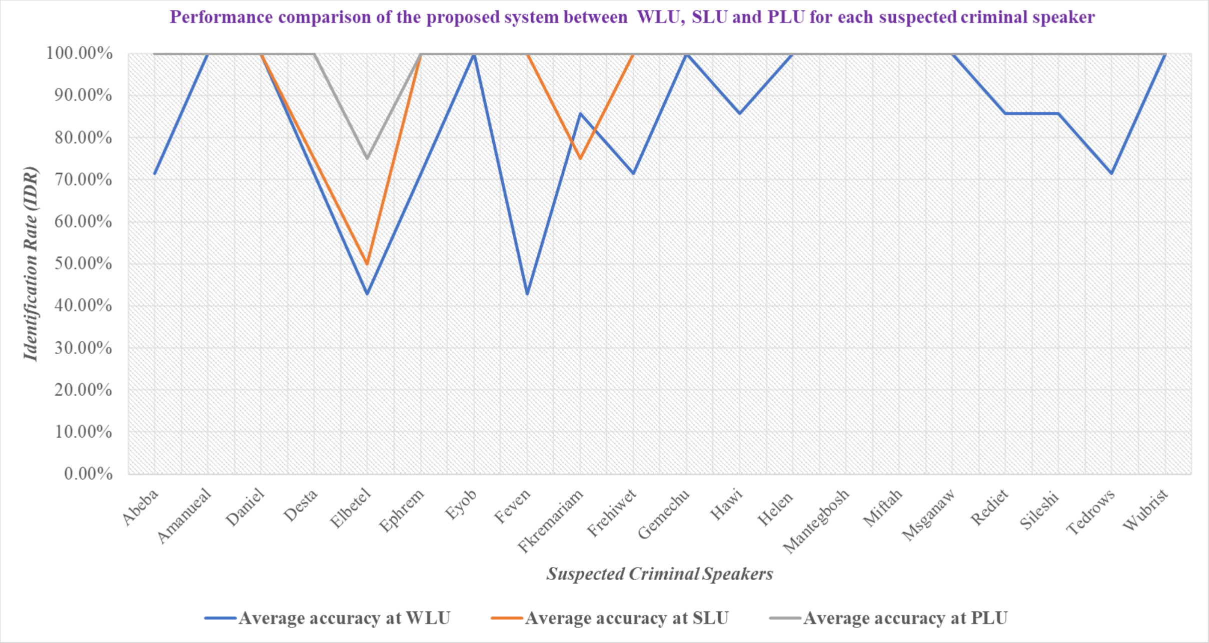


Figure 6.5: Performance comparison between WLU, SLU and PLU for individual suspect

From the result comparison revealed on Figure 6.5, the level of utterance has its impact on the suspected speaker identification performance. The proposed Forensic Speaker Recognition system depicted a better average identification performance for almost all suspected criminal speakers when they represented with models that were obtained from the long level utterances, which is from paragraph-level of utterances. For instance, our proposed system, identified Desta (suspect 3), 42.86%, 50.0% and 75.0% for WLU, SLU and PLU respectively. This infers that, when the training and testing corpus level (length) is getting more, the system's learning ability also can increase to capture the fine detail vocal tract characteristics of the individual speaker that can enable to discriminate one speaker from the others.

Experiment 3: Impact of crossed levels of utterance on IDR

In real forensic corpora, there might be a mismatch between training (suspected speaker reference) and testing (traces) channels, which could seriously degrade the forensic speaker recognition system performances [89]. Among the mismatch, a channel can be recording environment channel mismatch, recording device channel mismatch, time channel mismatch, background channel mismatch. This particular experiment was conducted to examine the impact crossed levels of utterances of the training and testing corpora. During the experiment, we have considered the following nine particular scenarios:

- WLU - WLU: In this scenario, the word-level utterance speech corpora play the roles of both training and testing.
- WLU - SLU: In this scenario, the word-level utterance corpora play the role of training while the sentence level utterance corpora used for testing.
- WLU - PLU: In this scenario, the word-level utterance corpora plays the role of training while the paragraph-level utterance corpora used for testing
- SLU - SLU: In this scenario, the sentence-level utterance speech corpora play the roles of both training and testing.
- SLU - WLU: In this scenario, the sentence-level utterance speech corpora play the role of training while the word-level utterance corpora used for testing.
- SLU - PLU: In this scenario, the sentence level-utterance speech corpora play the role of training while the paragraph-level utterance corpora used for testing.
- PLU - WLU: In this scenario, the paragraph-level utterance corpora play the role of training while the word-level utterance corpora used for testing.
- PLU - SLU: In this scenario, the paragraph-level utterance speech corpora play the role of training while the sentence-level utterance corpora used for testing.
- PLU - PLU: In this scenario, the paragraph-level utterance speech corpora play the roles of both training and testing.

Table 6.2: The impact of crossed level of utterances of training and testing on IDR

Training -Testing Scenario	Identification Rate (IDR)
WLU-WLU	84.29%
WLU-SLU	15.00%
WLU-PLU	13.75%
SLU-WLU	43.57%
SLU-SLU	95.00%
SLU-PLU	75.50%
PLU-WLU	42.14%
PLU-SLU	76.25%
PLU-PLU	97.50%

The results of this experiment analyzed by categorized into three as follow:

Category1: Scenarios marked with red color-this category represented the scenarios with short- long utterance training-testing circumstances, and the results are the lowest(worst). i.e. WLU- SLU, WLU-PLU, SLU- PLU with 15.00%,13.75%,75.50% respectively.

Category2: Scenarios marked with blue color-this category represented the scenarios long- short utterance training-testing circumstances, and the results are better than the preceding category's circumstances. i.e. SLU-WLU, PLU-WLU, PLU-SLU with 43.57%, 42.14%,76.25% respectively.

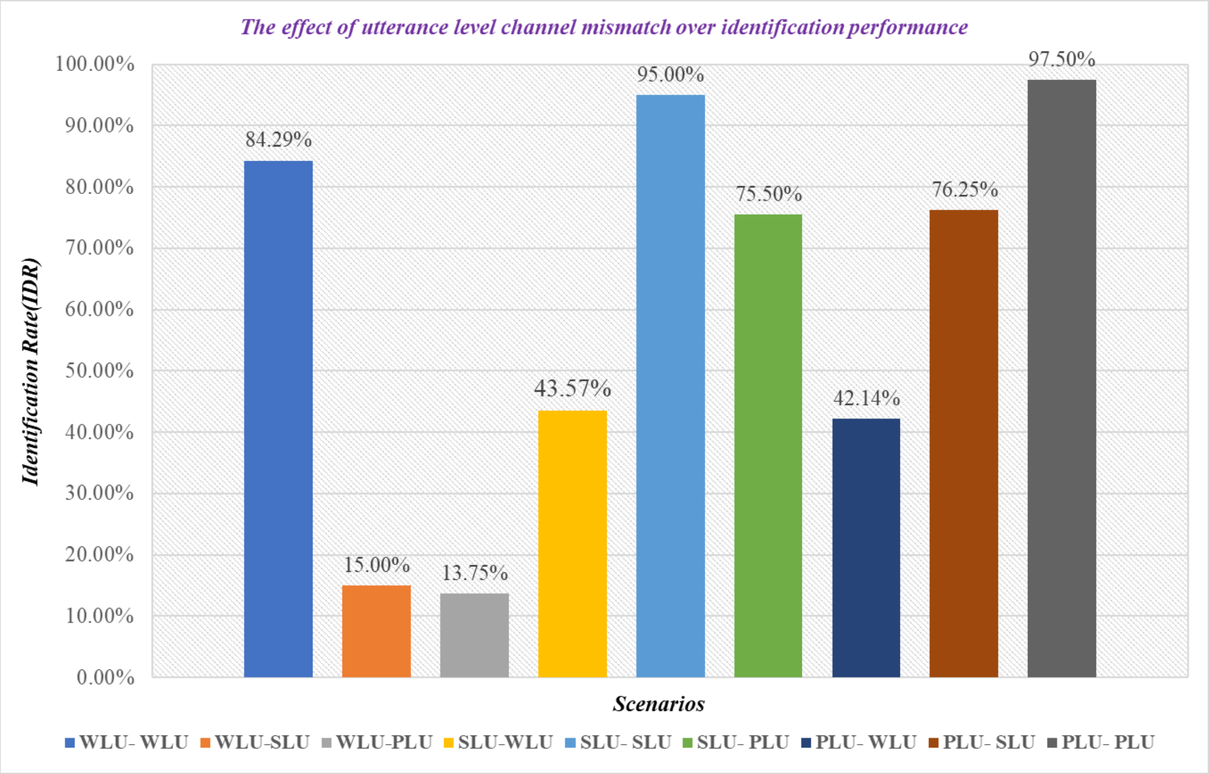


Figure 6.6: Identification performances for level of utterance channel mismatch scenarios

Category3: Scenarios marked with purple color: this category represented the scenarios with a parallel level of utterance training - testing circumstances, and the results are best of the preceding categories' circumstances. i.e. WLU-WLU,SLU-SLU,PLU-PLU with 84.29%,95.00%,97.50% respective identification accuracy.

As we have seen from Experiment 2.1 to Experiment 2.3, and Experiment 3(category3:) results, the proposed system achieved 84.29%, 95.00% and 97.50% respective IDR for WLU, SLU and PLU. On those experiments, we have been used a parallel level of utterances for both training and testing purposes. However, as it could see from Experiment 3 (Table 6.2 and Figure 6.6), when the level of utterances crossed for the training and testing purposes, the system revealed a degraded identification performance. These degraded identification performances implied that, to have a better identification accuracy, it is recommended to prepare, and use corpus set for the training purpose from the concerned entities (such as suspected speakers) relying on the trace evidence level of utterance/s for testing, which is recorded from the given event (in our case, from the crime commissioning moment).

Experiment 4: Impact of the recording device mismatch on IDR

For instance, the trace utterances of the unknown criminal might be recorded from a telephone conversation at the receiver side, but the training speech corpora of the suspects might be recorded using an interview microphone by the policeman or forensic expert. In such a case, there is a recording device mismatch between the training and testing corpora. This experiment was carried out to observe this channel mismatch impact on the identification performance. During the experiment, we used the speech recorded through an interview microphone as training input, and the speech from a mobile phone conversation as a testing trace.

Table 6.3: The impact of recording device mismatch on IDR

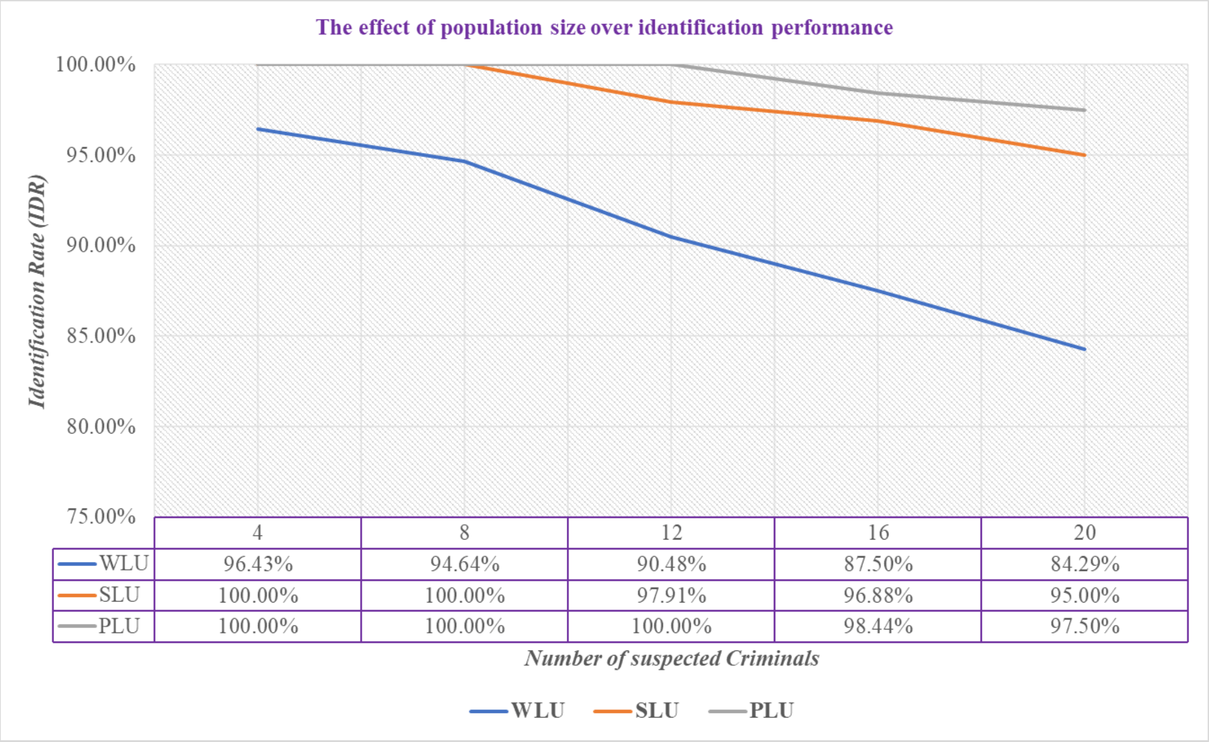
Training Corpora (Microphone recorded)	Testing Corpora (Mobile phone recorded)	Identification Rate (IDR)
WLU	WLU	47.86%
SLU	SLU	61.25%
PLU	PLU	78.75%

As it could be observed from the results, Table 6.3, the recording device channel mismatch has shown a significant performance degradation for all the respective levels of utterance compared to the result which is conducted using a parallel recording channel. This is due to the fact that, when the human voice passed through a network, there is a chance that the speech to compressed with the transmission network, and that could be a factor to create a certain deviation of the individual voice when recorded from a telephone conversation, and a microphone interview. This experiment results recommended as the training corpora and testing trace evidence must be recorded with a parallel recording channel for better identification performance.

Experiment 5: Impact of the size of suspects on IDR

The size of speakers population to be examined is one difficulty factor in the recognition task[76]. In this experiment, the impact of an increasing number of suspected criminals on the identification rate of the proposed system was tested as a function of number of suspected criminal speakers. As reveals from Figure 6.7, the number of suspected speakers taken is 4 to 20.

From the indication of this experiment results (Figure 6.7), for a fewer number of suspects, the IDR is better for all the three levels of utterances. For instance, when the number of suspects is 4, the IDR is 96.43%, 100.00% and 100.00% for WLU, SLU and PLU respectively. But when the number of suspects made to increase, the identification performance of the proposed system depicted a gradual degradation. For instance, when the number of suspects raised to 20, the IDR degraded to 84.29%, 95.00% and 97.50% for WLU, SLU and PLU respectively. From this observation, the feasible conclusion in all the sets of speakers is that irrespective of the levels of utterance, as the number of suspects increases, the performance in identifying the actual criminal degrades. This is due to the fact that, as the number of suspects that the system must distinguish increases, the computation complexity of the system is increased, and this computation complexity, in turn, created confusion to match the features extracted from the testing trace evidence with the respective suspect's reference model accurately and figure out the actual criminal speaker easily, and this led the system to decreases in identification performance. As of the experiment which had been done on[53], as the number of speakers increased, the performance of the classifier decreased. And as the researcher justified that, when the number of speakers increased, there would be more number of reference models, as a



	4	8	12	16	20
WLU	96.43%	94.64%	90.48%	87.50%	84.29%
SLU	100.00%	100.00%	97.91%	96.88%	95.00%
PLU	100.00%	100.00%	100.00%	98.44%	97.50%

Figure 6.7: The effect of population size over-identification accuracy

result, the classifier's accuracy in matching the feature array with the appropriate model declines. And also, as the number of speakers increases, the probability of having similar templates increases. This increment in similarity makes the system to pass a wrong decision on the identity of the individuals. Hence, the probability of wrongly identification increases.

6.2.1 6-Fold Cross-Validation

We have used a stratified K-fold cross-validation technique, and implemented as follow, initially, we split our training corpora set into 6-folds (in fact the default is 10-fold, but for our case it created corpora imbalance problem per fold, so for the sake of fairness among the suspects, we preferred to use a K-value that enabled us to split the corpora equal number of set per fold, which is 6). Hence, each fold made to held the same percentage of corpora set for each class, suspect. Then, we trained our model using 5-folds (80% of the training corpora set) as training set and validated using the remaining testing fold, 1-fold (20%). As shown in Figure 6.8, we have done the training and testing task six -times.Then, after performed the 6-fold cross-validation, as shown from table 6.4, we got x,y and c average accuracy for WLU, SLU and PLU.

Table 6.4 reveals IDR for the three levels of utterances by averaging every performance of each fold and by taking their accuracy while validating after the training has been done.

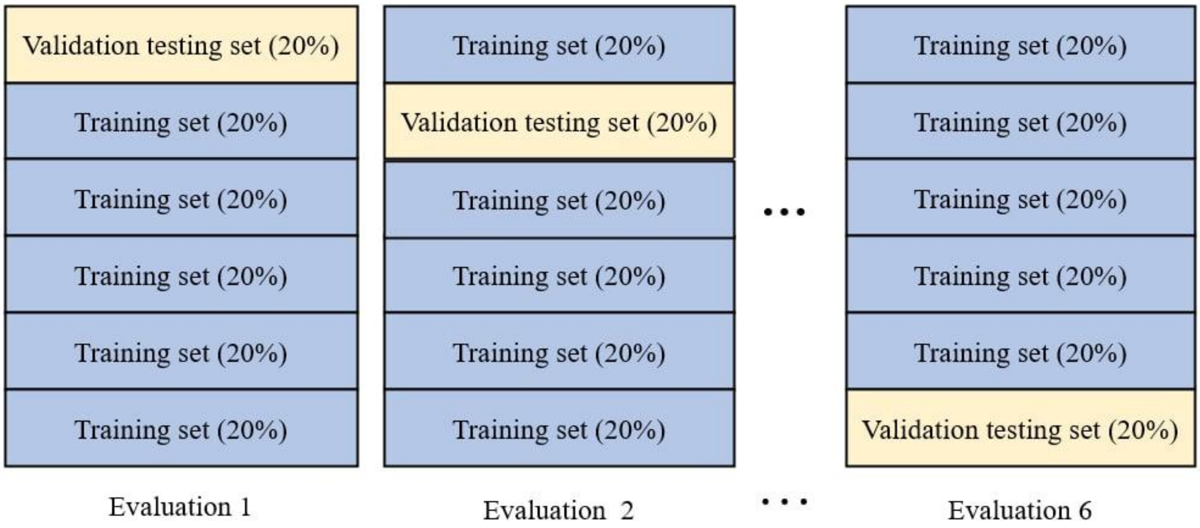


Figure 6.8: Corpora representation for 6-fold cross validation

Table 6.4: Result of the cross validation testing

Corpora level	Evaluations of 6-fold cross validation						Average IDR
	Eval-1	Eval-2	Eval-3	Eval-4	Eval-5	Eval-6	
WLU	86.67%	84.17%	81.67%	79.17%	82.50%	87.50%	83.61%
SLU	90.00%	85.00%	95.00%	95.00%	100.00%	80.00%	90.83%
PLU	95.00%	90.00%	100.00%	95.00%	90.00%	95.00%	94.17%

6.3 Discussion

Figure 6.9 reveals the overall results for the WLU, SLU, and PLU. The observed results implied that long utterances are more convenient to model each speaker adequately and efficiently to perform the identification task. In 2019, a group of researchers has been done a study[90]. In the experiment section, they experimented to see the impact of variable length of input speech on speaker recognition accuracy. As of a result, the recognition accuracy increased as the length of the speech of the speaker increased. And the authors reason out that, the increased length of speech provides an increased details of the speaker's voice characteristics which in turn helps in computing MFCC that are close to the Speaker's model formed during the training phase.

Interestingly, in our experiment, we achieved a better performance of identifying the actual criminal at the usage of PLU corpora for both training and testing purposes than the WLU and SLU circumstances. Thus, we can conclude that the training and testing level of utterances greatly affects the proposed system recognition performance, and long utterance preferable for achieving a better recognition performance with training and testing time expense.his is due to the reason that when the speech corpora level of the utterance getting longer, it can offer a better input feature vector representation of the given individual's vocal tract shape in training-testing phases. Also,as we can observe from the figure, for uni-modal training-testing recording scenarios, the proposed

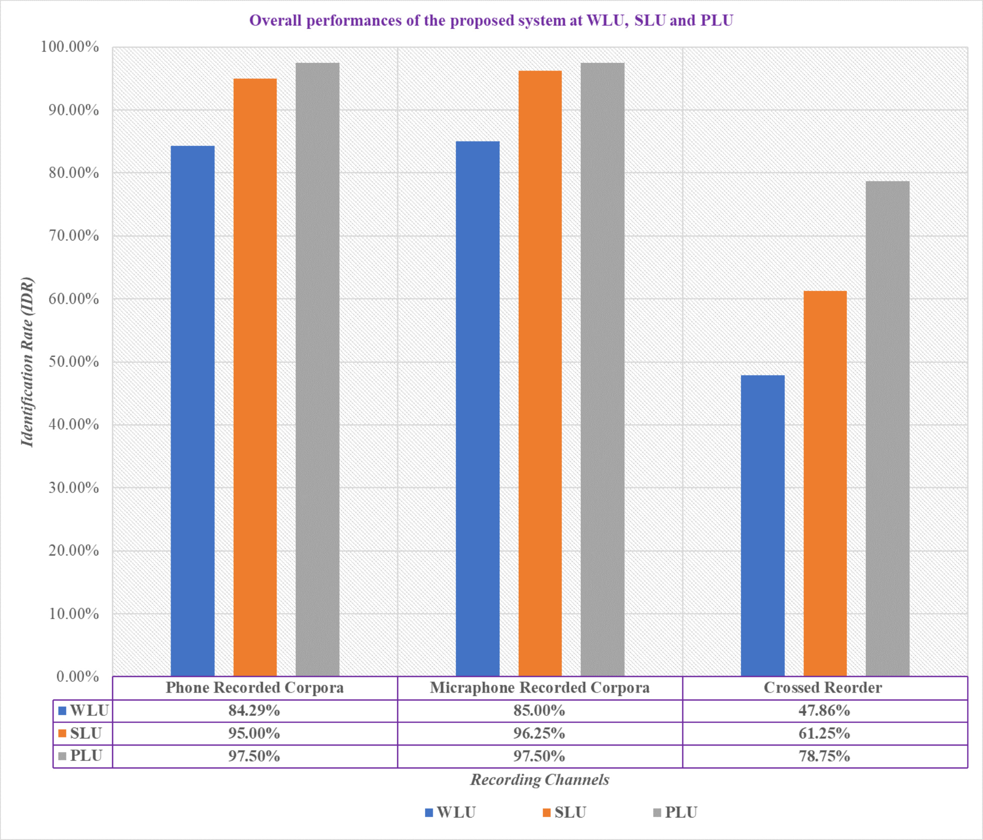


Figure 6.9: Overall performance of the proposed system at WLU, SLU and PLU

system achieved better performances while for crossed training-testing recording device mismatch, the performance is challenging yet.

6.4 Summary of this thesis with related works

As mentioned under section 2.3.2, we could not found any local work on forensic speaker recognition. Hence, we incorporated works on general-purpose speaker recognition for Amharic languages and FSR works done for foreign languages. In fact, we cannot compare these works directly with our work since there are many deviations (deviation in corpora size, language and level of utterance, feature extraction approach technique and size, modeling technique, implementation tool, and number of population) between the works mentioned as a related works and ours. This summary is presented here just to show the scenarios we have taken compared to the given related works from the perspective of level of utterance, corpora size, feature size, implementation tool and recording mechanisms. This study examine the impact of level of utterance over recognition performance, and revealed corpora level of utterance preference for subsequent related studies and systems development. As well, unlike the cited related works (except the first related work, the rest had been implemented relying upon the default feature vector dimension), this study implemented using a high dimension feature vector, determined from sampling rate of the corpus.

Table 6.5: Summary of this thesis work with related works

Language Research Year	Level of utterance	Number of speaker	Number of utterances	Feature Extraction approach	Feature vector size	Modeling approach	Tool	Result
English [49] 2017	SLU	6	30	GFCC	64	GMM-UBM	MATLAB	16.67% EER
Hindi [50] 2017	WLU	15	150	MFCC	13	VQ and GMM	MATLAB	[85.49%,94.12%] IDR [VQ, GMM] for TD [77.64%,86.27%] IDR [VQ, GMM] for TI
Malayalam [51] 2018	PLU	18	1232	MFCC and PNCC	13	i-vector and PLDA	MATLAB	49.89%, 41.0%, 46.64%, 30.37%] IDR for [White,Vehicle,Factory,Babble] using MFCC. [49.89%, 41.0%, 46.64%, 30.37%] IDR for [White, Vehicle, Factory, Babble] using PNCC
Mexican [52] 2019	PLU	120	1200	MFCC	13	GMM	MATLAB	[93.57%,97.14%] IDR at [noisy,clean] for males [95.28%,96.92%] IDR at [noisy,clean] for females
Amharic [53] 2015	SLU	50	500	MFCC + ΔMFCC + ΔΔMFCC	39	VQ and GMM	MATLAB	[74.2%,84.3%] IDR for [VQ ,GMM] respectively.
Amharic [54] 2017	SLU	90	270	GFCC + MFCC	39	VQ,GMM and BPNN	MATLAB	[69.0%,78.3%,84.7%] IDR for [VQ, GMM, BPNN] of 30,60 and 90 speakers
Amharic [55] 2017	SLU	100	300	GFCC + MFCC	39	GMM + BPNN	MATLAB	92.7% accuracy had been achieved for 100 speakers respectively
Amharic [56] 2019	SLU	10	200	MFCC + ΔMFCC + ΔΔMFCC	39	ANN	MATLAB	[96%, 96.7%, 97.3%] IDR for [15,20,25] number of hidden neurons respectively.
Amharic [Ours] 2021	WLU, SLU, PLU	20	2400	MFCC + ΔMFCC	66	GMM	PYTHON	[84.29% 95.00%, 97.50%] IDR for [WLU,SLU, PLU] of mobile phone conversation corpora [84.29%,96.25%, 97.50%] IDR for [WLU,SLU,PLU] of microphone recorded speech corpora

CHAPTER 7

CONCLUSION AND RECOMMENDATION

7.1 Conclusion

Bio-metrics is the measurement and statistical analysis of people's unique physical and behavioral traits. Speaker recognition is the task of recognizing an individual based on his or her voice trait. Forensic science is the use of scientific and technical methods for revealing the occurrence of a crime and determine identity of the doer. Forensic speaker recognition is an established term when the speaker recognition techniques are adopted to forensic applications on the basis of voice bio-metrics. In this study, a text-independent speaker identification techniques has been adopted to design a FSR system for criminal investigation. To conduct the study, efforts have been distributed into corpora collection, corpora preprocessing, front-end feature extraction and back-end feature classification design elements.

To carried out this study, an Amharic language speech corpora prepared from an ongoing mobile phone conversation and microphone interview reading speech records. Usually, while committing crimes through speech, the criminals spoke out randomly to harass, ransom, irritate, frustrate, etc. the receiver. Thus, to involve such a realistic context scenario, while collecting and preparing the speech corpora, we have tried to encompass most of the Amharic Fidels' sound. Once the collection and preparation has been done, to enhance quality of the corpora, Spectral Gating based noise reduction and VAD based silence truncation mechanisms employed as preprocessing techniques. Then, MFCC and GMM utilized as a front-end feature extraction and back-end feature classification approaches respectively. To obtain a speaker-specific acoustic feature vectors for each suspect, the first 33-dimensional static MFCCs feature vectors appended with their first order, ΔMFCCs. Dimension of the extracted feature vector per frame determined from the sampling rate of the corpora using the rule of thumb. Then, once having the feature vector, to generate a reference model for the suspects and identify the criminal, a GMM used with EM learning algorithm and MLL score for the enrollment and identification operations respectively.

From the vantage point of answering the research question of this study, during the experiment section more emphasis was provided in examining the impact of the level of utterance over recognition (identification) performance. That is why, all the experiments had been carried out using three levels of utterances (word, sentence, and paragraph levels of utterances), as well efforts have been applied to address the issues of modeling order, level of utterance, crossed level of utterance, recording channel mismatch (multi-modality) and population size over a recognition performance.

As the executed experimental results revealed, in all recognition (identification) trial scenarios, the proposed system outperformed for a speech corpora prepared from longer training-testing utterances. I.e, the system outperformed for sentence-level utterances compared to word-level utterances, and in turn, for paragraph-level utterances compared to sentence-level training-testing utterances.

Finally from this study, we can conclude that apart from selecting fitting feature extracting and modeling approaches, the level (length) of utterance also has a substantial impact in determining performance of a given FSR system; and longer utterances enable to accomplish a better recognition performance with the expense of longer training-testing time. This is due to the reason that when the speech corpora level of the utterance getting longer, it can offer a better input feature vector representation of the given individual's vocal tract shape in training-testing phases. And as well, the proposed system performed better for parallel level of utterances and uni-modal recording training-testing scenarios. However,the proposed system poorly performed for crossed levels of utterances and multi-modal recording training-testing scenarios yet. Hence, improving these poor performances can be the next research direction of this study. Also, even if we have a plenty local languages, due to corpora set constraint, we couldn't examine the language dependency scenario.

7.2 Recommendation

Due to constraints related to corpora size, our system has been implemented using a Machine Learning state-of-the-art on a few suspects with a limited corpora size. There are several ways that this study could be extended. For instance, designing this system using a more advanced approach, such as Deep Learning state-of-the-arts with a large size of speech corpora that helps to boost recognition performance. Dealing the issue of threshold values, and implementing the proposed system with an open-set scenario is another view to make it more genuine from a Forensic perspective since there is a possibility that the criminal to be out of the handed, modeled suspects. Addressing the matters of twines, recording time mismatch, abnormal conditions (alcoholic and health impacts) and speaker diaraziation are also another critical potential issues in FSR studies.

Reference

[1] Bishnu S Atal. Automatic recognition of speakers from their voices. *Proceedings of the IEEE*, Vol. 64(4), pp. 460-475, 1976.

[2] Ling Feng. Speaker recognition. Master's thesis, Technical University of Denmark (DTU), Denmark, 2004.

[3] Andrzej Drygajlo. Forensic automatic speaker recognition [exploratory dsp]. *IEEE Signal Processing Magazine*, Vol. 24(2), pp. 132-135, 2007.

[4] Christophe Champod and Didier Meuwly. The inference of identity in forensic speaker recognition. *Speech communication*, Vol. 31(2-3), pp. 193-203, 2000.

[5] Joseph P Campbell, Wade Shen, William M Campbell, Reva Schwartz, Jean-Francois Bonastre, and Driss Matrouf. Forensic speaker recognition. *IEEE Signal Processing Magazine*, Vol. 26(2), pp. 95-103, 2009.

[6] Siddique Latif, Muhammad Usman, Sanaullah Manzoor, Waleed Iqbal, Junaid Qadir, Gareth Tyson, Ignacio Castro, Adeel Razi, Maged N Kamel Boulos, Adrian Weller, et al. Leveraging data science to combat covid-19: A comprehensive review. *IEEE Transactions on Artificial Intelligence*, Vol. 1(1), pp. 85-103, 2020.

[7] Viva Voce and Class Record. Ug criminology syllabus department of criminology, summary chart–academic inputs: 2015-2018.

[8] Thomas Fang Zheng and Lantian Li. *Robustness-related issues in speaker recognition.* 2017.

[9] Tomi Kinnunen and Haizhou Li. An overview of text-independent speaker recognition: From features to supervectors. *Speech communication*, Vol. 52(1), pp. 12-40, 2010.

[10] A Daniel Yarmey, Meagan J Yarmey, and Leah Todd. Frances mcgehee (1912-2004): The first earwitness researcher. *Perceptual and motor skills*, Vol. 106(2), pp. 387-394, 2008.

[11] Frances McGehee. The reliability of the identification of the human voice. *The Journal of General Psychology*, Vol. 17(2), pp. 249-271, 1937.

[12] Lawrence George Kersta. Voiceprint identification. *Nature*, Vol. 196, 1962.

[13] Sandra Pruzansky. Pattern-matching procedure for automatic talker recognition. *The Journal of the Acoustical Society of America*, Vol. 35(3), pp. 354-358, 1963.

[14] James E Luck. Automatic speaker verification using cepstral measurements. *The Journal of the Acoustical Society of America*, Vol. 1(4), pp. 1026-1032, 1969.

[15] Bishnu S Atal. Effectiveness of linear prediction characteristics of the speech wave for automatic speaker identification and verification. *the Journal of the Acoustical Society of America*, Vol. 55(6), pp. 1304-1312, 1974.

[16] Sadaoki Furui. Cepstral analysis technique for automatic speaker verification. *IEEE Transactions on Acoustics, Speech, and Signal Processing.*, Vol. 29(2), pp. 254-272, 1981.

[17] Hynek Hermansky, B Hanson, and Hisashi Wakita. Perceptually based linear predictive analysis of speech. In *ICASSP'85. IEEE International Conference on Acoustics, Speech, and Signal Processing.*, volume Vol. 10, pp. 509-512. IEEE, 1985.

[18] Steven Davis and Paul Mermelstein. Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. *IEEE transactions on acoustics, speech, and signal processing*, Vol. 28(4), pp. 357-366, 1980.

[19] Md Jahangir Alam, Tomi Kinnunen, Patrick Kenny, Pierre Ouellet, and Douglas O'Shaughnessy. Multitaper mfcc and plp features for speaker verification using i-vectors. *Speech communication*, Vol. 55(2), pp. 237-251, 2013.

[20] Lawrence Rabiner and B Juang. An introduction to hidden markov models. *ieee assp magazine*, Vol. 3(1), pp. 4-16, 1986.

[21] Anil K Jain, Jianchang Mao, and K Moidin Mohiuddin. Artificial neural networks: A tutorial. *Computer*, Vol. 29(3), pp. 31-44, 1996.

[22] Tahira Mahboob, Memoona Khanum, Malik Sikandar Hayat Khiyal, and Ruqia Bibi. Speaker identification using gmm with mfcc. *International Journal of Computer Science Issues (IJCSI)*, Vol. 12(2), pp. 126, 2015.

[23] David Sierra Rodríguez. Text-independent speaker identification. *AGH University Of Science and Technology Krakow, Kraków*, 2008.

[24] Najim Dehak, Patrick J Kenny, Réda Dehak, Pierre Dumouchel, and Pierre Ouellet. Front-end factor analysis for speaker verification. *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19(4), PP. 788-798, 2010.

[25] George E Dahl, Dong Yu, Li Deng, and Alex Acero. Context-dependent pre-trained deep neural networks for large-vocabulary speech recognition. *IEEE Transactions on audio, speech, and language processing*, Vol. 20(1),pp. 30-42, 2011.

[26] Yun Lei, Nicolas Scheffer, Luciana Ferrer, and Mitchell McLaren. A novel scheme for speaker recognition using a phonetically-aware deep neural network. In *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1695–1699. IEEE, 2014.

[27] Zhiyuan Tang, Lantian Li, and Dong Wang. Multi-task recurrent model for speech and speaker recognition. In *2016 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA)*, pages 1–4. IEEE, 2016.

[28] Nilu Singh, RA Khan, and Raj Shree. Applications of speaker recognition. *Procedia engineering*, Vol. 38, 2012.

[29] Anil K Jain, Karthik Nanda kumar, and Abhishek Nagar. Biometric template security. *EURASIP Journal on advances in signal processing*, Vol. 2008, pp. 1-17, 2008.

[30] Rob Fitzpatrick, Mark Raven, and Peter Self. Using forensics to inspire the next generation of regolith, soil and clay scientists. In *Proceedings, Combined Australian Regolith Geoscientists Association and Australian Clay Minerals Society Conference, Mildura (Victoria, Australia)*, 2012.

[31] Andrzej Drygajlo and Rudolf Haraksim. Biometric evidence in forensic automatic speaker recognition. In *Handbook of Biometrics for Forensic Science*. Springer, 2017.

[32] Nengheng Zheng, Ning Wang, Tan Lee, and PC Ching. Speaker verification using complementary information from vocal source and vocal tract. In *International Symposium on Chinese Spoken Language Processing*, pages pp. 518–528. Springer, 2006.

[33] Frank K Soong, Aaron E Rosenberg, Bling-Hwang Juang, and Lawrence R Rabiner. Report: A vector quantization approach to speaker recognition. *AT&T technical journal*, Vol. 66 (2), pp. 14-26, 1987.

[34] Lin and Shuxun Wang. A kernel method for speaker recognition with little data. In *2006 8th international Conference on Signal Processing*, volume Vol. 1. IEEE, 2006.

[35] Pillay Surosh Govindasamy. *Voice biometrics under mismatched noise conditions.* PhD thesis, University of Hertfordshire, United Kingdom, 2011.

[36] Christopher JC Burges. A tutorial on support vector machines for pattern recognition. *Data mining and knowledge discovery*, Vol. 2(2), pp. 121-167, 1998.

[37] Homayoon Beigi. Speaker recognition. In *Fundamentals of Speaker Recognition*, pages pp. 543–559. Springer, 2011.

[38] Sinith M S, Anoop Salim, K. Sankar, K. Narayanan, and Vishnu Soman. A novel method for text-independent speaker identification using mfcc and gmm. pages pp. 292 – 296, 2010.

[39] Arthur P Dempster, Nan M Laird, and Donald B Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society: Series B (Methodological)*, Vol. 39(1), pp. 1-22, 1977.

[40] Douglas A Reynolds. Automatic speaker recognition using gaussian mixture speaker models. volume Vol. 8(2), pp. 173-192. IEEE, 1995.

[41] Hector NB Pinheiro, Tsang Ing Ren, George DC Cavalcanti, Tsang Ing Jyh, and Jan Sijbers. Type-2 fuzzy gmm-ubm for text-independent speaker verification. In *2013 IEEE International Conference on Systems, Man, and Cybernetics*, pages pp. 4328–4331. IEEE, 2013.

[42] Huo Chun Bao and Zhang Cai Juan. The research of speaker recognition based on gmm and svm. In *2012 international conference on system science and engineering (ICSSE)*, pages pp. 373–375. IEEE, 2012.

[43] Abdul Wahab, Goek See Ng, and Romy Dickiyanto. Speaker authentication system using soft computing approaches. *Neurocomputing*, Vol. 68 pp. 13-37, 2005.

[44] Fatma zohra Chelali, Amar Djeradi, and Rachida Djeradi. Speaker identification system based on plp coefficients and artificial neural network. *environments*, Vol. 1, pp. 2, 2011.

[45] Tomi Kinnunen and Haizhou Li. An overview of text-independent speaker recognition: From features to supervectors. *Speech communication*, Vol. 52 (1), pp. 12-40, 2010.

[46] Larry P Heck, Yochai Konig, M Kemal Sönmez, and Mitch Weintraub. Robustness to telephone handset distortion in speaker recognition by discriminative feature design. *Speech Communication*, Vol. 31 (2), pp. 181-192, 2000.

[47] Rotate Clockwise Rotate Counterclockwise, Zoom Out, Zoom In, Geoffrey Hinton, Li Deng, Dong Yu, George Dahl, Abdel-rahman Mohamed, Navdeep Jaitly, Andrew Senior, et al. Deep neural networks for acoustic modeling.

[48] Yun Lei, Nicolas Scheffer, Luciana Ferrer, and Mitchell McLaren. A novel scheme for speaker recognition using a phonetically-aware deep neural network. In *2014 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pages pp. 1695–1699. IEEE, 2014.

[49] Mohammed Algabri, Hassan Mathkour, Mohamed A Bencherif, Mansour Alsulaiman, and Mohamed A Mekhtiche. Automatic speaker recognition for mobile forensic applications. *Mobile Information Systems*, 2017.

[50] Ankur Maurya, Divya Kumar, and RK Agarwal. Speaker recognition for hindi speech signal using mfcc-gmm approach. *Procedia Computer Science*, Vol. 125, pp. 880-887, 2018.

[51] Arathy Ajit, Anu George, and Leena Mary. I-vectors for forensic automatic speaker recognition. 2018.

[52] Abel Herrera-Camacho, Adrián Zúñiga-Sainos, Gerardo Sierra-Martínez, José Trangol-Curipe, Margarita Mota-Montoya, and Adonay Jarquín-Casas. Design and testing of a corpus for forensic speaker recognition using mfcc, gmm and mle. In *Proceedings of the 2019 International Conference on Video, Signal and Image Processing*, 2019.

[53] Aykefam Azene. Text-independent speaker identification for the amharic language. Master's thesis, Bahir Dar University, Ethiopia, 2015.

[54] Abrham Debasu Mengistu and Dagnachew Melesew Alemayehu. Text independent amharic language speaker identification in noisy environments using speech processing techniques. *Indonesian Journal of Electrical Engineering and Computer Science*, Vol. 5(1), pp. 109-114, 2017.

[55] Abrham Debasu Mengistu. Automatic text independent amharic language speaker recognition in noisy environment using hybrid approaches of lpcc, mfcc and gfcc. *International Journal of Advanced Studies in Computers, Science and Engineering*, Vol. 6(5), pp. 8, 2017.

[56] Belayneh Gizachew. Artificial neural network based amharic language speaker recognition. Master's thesis, ASTU, Ethiopia, 2019.

[57] Homayoon Beigi. Speaker recognition: Advancements and challenges. *New trends and developments in biometrics*, 2012.

[58] Hasan Muaidi, Ayat Al-Ahmad, Thaer Khdoor, Shihadeh Alqrainy, and Mahmud Alkoffash. Arabic audio news retrieval system using dependent speaker mode, mel frequency cepstral coefficient and dynamic time warping techniques. *Research Journal of Applied Sciences, Engineering and Technology*, Vol. 7(24), pp. 5082-5097, 2014.

[59] Hiroaki Sakoe and Seibi Chiba. Dynamic programming algorithm optimization for spoken word recognition. *IEEE transactions on acoustics, speech, and signal processing*, Vol. 26(1), pp. 43-49, 1978.

[60] Abhishek Manoj Sharma. Speaker recognition using machine learning techniques. 2019.

[61] Bethlehem Mengistu. N-gram-based automatic indexing for amharic text. *Addis Ababa University*, 2002.

[62] Andrey Barinov. Voice samples recording and speech quality assessment for forensic and automatic speaker identification. In *Audio Engineering Society Convention 129*. Audio Engineering Society, 2010.

[63] Muhammad Asadullah and Shibli Nisar. A silence removal and endpoint detection approach for speech processing. *Sarhad University International Journal of Basic and Applied Sciences*, Vol. 4, 2017.

[64] Chandar Kumar, Faizan Ur Rehman, Shubash Kumar, Atif Mehmood, and Ghulam Shabir. Analysis of mfcc and bfcc in a speaker identification system. 2018.

[65] Ta-Wen Kuan, An-Chao Tsai, Po-Hsun Sung, Jhing-Fa Wang, and Hsien-Shun Kuo. A robust bfcc feature extraction for asr system. *Artif. Intell. Research*, Vol. 5 (2), pp. 14-23, 2016.

[66] Taabish Gulzar, Anand Singh, and Sandeep Sharma. Comparative analysis of lpcc, mfcc and bfcc for the recognition of hindi words using artificial neural networks. *International Journal of Computer Applications*, Vol. 101(12),pp. 28-32, 2014.

[67] Faizan ur Rehman, Chandar Kumar, Shubash Kumar, Atif Mehmood, and Umair Zafar. Vq based comparative analysis of mfcc and bfcc speaker recognition system. In *2017 International Conference on Information and Communication Technologies (ICICT)*. IEEE, 2017.

[68] Chandar Kumar, Faizan Ur Rehman, Shubash Kumar, Atif Mehmood, and Ghulam Shabir. Analysis of mfcc and bfcc in a speaker identification system. In *2018 International Conference on Computing, Mathematics and Engineering Technologies (iCoMET)*. IEEE, 2018.

[69] Ramisetty Rajeswara Rao, V. Prasad, and Akkaladevi Nagesh. Performance evaluation of statistical approaches for text independent speaker recognition using source feature. *Computing Research Repository - CORR*, 2011.

[70] Andrey Barinov. Voice samples recording and speech quality assessment for forensic and automatic speaker identification. In *Audio Engineering Society Convention 129*. Audio Engineering Society, 2010.

[71] Bidhan Barai, Subhadip Basu, Mita Nasipuri, Debayan Das, and Nibaran Das. Vq/gmm based speaker identification with emphasis on language dependency. 2018.

[72] Rajeev Ranjan and Abhishek Thakur. Analysis of feature extraction techniques for speech recognition system. *International Journal of Innovative Technology and Exploring Engineering*, 2019.

[73] Arun G Rajsekhar. Real time speaker recognition using mfcc and vq. *Department of Electronics & Communication Engineering National Institute of Technology Rourkela*, 2008.

[74] Fang Zheng, Guoliang Zhang, and Zhanjiang Song. Comparison of different implementations of mfcc. *Journal of Computer science and Technology*, Vol. 16 (6), pp. 582-589, 2001.

[75] Frank K Soong and Aaron E Rosenberg. On the use of instantaneous and transitional spectral information in speaker recognition. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, Vol. 36 (6), pp. 871-879, 1988.

[76] Douglas A Reynolds and Richard C Rose. Robust text-independent speaker identification using gaussian mixture speaker models. *IEEE transactions on speech and audio processing*, Vol. 3 (1), pp. 72-83, 1995.

[77] Chee-Ming Ting, Sh-Hussain Salleh, Tian-Swee Tan, and AK Ariff. Text independent speaker identification using gaussian mixture model. In *2007 International Conference on Intelligent and Advanced Systems*. IEEE, 2007.

[78] Ulric Neisser. Cognitive psychology appleton-century-crofts. *New York*, Vol. 351, 1967.

[79] Olivier Bousquet, Ulrike von Luxburg, and Gunnar Rätsch. *Advanced Lectures on Machine Learning: ML Summer Schools 2003, Canberra, Australia, February 2-14, 2003, Tübingen, Germany, August 4-16, 2003, Revised Lectures*, volume Vol. 3176. Springer, 2011.

[80] Mehryar Mohri, Afshin Rostamizadeh, and Ameet Talwalkar. Foundations of machine learning. adaptive computation and machine learning. *MIT Press*, Vol. 13 (1), pp. 981-1006, 2012.

[81] Kauleshwar Prasad, Piyush Lotia, and MR Khan. A review on text-independent speaker identification using gaussian supervector svm. *International Journal of u- and e-Service, Science and Technology*, Vol. 5 (1), pp. 71-82, 2012.

[82] Kirandeep Kaur and Neelu Jain. Feature extraction and classification for automatic speaker recognition system – a review. volume Vol. 5(1), pp. 1-6, 2015.

[83] D. Reynolds. Gaussian mixture models. In *Encyclopedia of Biometrics*, volume Vol. 741, pp. 659-663, 2009.

[84] Rafael González Ayestarán. Text-independent speaker identification, master thesis. 2008.

[85] John R Deller, John G Proakis, and John HL Hansen. *Discrete-time processing of speech signals.* PhD thesis, 2000.

[86] G Alastair Young. *Mathematical Statistics: An Introduction to Likelihood Based Inference Richard J. Rossi John Wiley & Sons, 2018, ebook ISBN: 978-1-118-77104-4, LCCN 2018010628 (ebook).* PhD thesis, 2019.

[87] David Sierra Rodríguez. Text-independent speaker identification. *AGH University Of Science and Technology Krakow, Kraków*, 2008.

[88] Douglas A Reynolds. Speaker identification and verification using gaussian mixture speaker models. *Speech communication*, Vol. 17(1), pp. 91-108, 1995.

[89] Achmad Fanany Onnilita Gaffar, Rheo Malani, Agusma Wajiansyah, Arief Bramanto Wicaksono Putra, et al. A multi-frame blocking for signal segmentation in voice command recognition. In *2020 International Seminar on Intelligent Technology and Its Applications (ISITIA)*. IEEE, 2020.

[90] Holambe Deshpande, Mangesh and Raghunath. Robust speaker identification in babble noise. volume Vol. 6(10), pp. 19. Citeseer, 2011.

Appendix A: Text Data

i.Word Level Text Data:

1. ባዶ
2. አንድ
3. ሁለት
4. ሶስት
5. አራት
6. አምስት
7. ስድስት
8. ሰባት
9. ስምነት
10. ዘጠኝ
11. ህዝባዊነት
12.ቅልቅል
13. ምርምር
14. ስነውበት
15. አነፍጠኛ
16. ብሽሽቅ
17. አንድነት
18. ትምክተኛ
19. ክምችት
20. ጠባብ
21. መኸር
22. አክራሪ
23. ውዥንር
24. ዘረኛ
25. ውግዘት
26. ጭፍን
27. ቅኝት
28. ውይይት
29. ጎጠኛ
30. ጅማሬ
31. ጳጳስ
32. ፅናት
33. ፖሊስ
34. ቋንቋ
35. አኗኗር
36. አሟሟት
37. አኳኋን
38. ሯጭ
39. ቧንቧ
40. አዋዋሏ

ii. Sentence Level Text Data:

1. ሄሎ ትሰማኛለህ የልጅህ ህይወት እጄ ላይ ነው፤ በሰላም እንዲለቀቅ ከፈለክ ጊዜ ሳትፈጅ እቃውን የተባባልነው ቦታ አስቀመጥ ፤ ሌላ ነገር ብታስብ ግን በልጅህ ህይወት እንደፈረድህ እወቅ፡፡

2. ሄሎ ወዳጄ ካንተ ባላውቅም ሰርክ በእግርህ ከመመላለስ አንድ ቀን በእጅህ ብትመጣ ውሎ ሳያድር ጉዳይህ መስመር መያዙ አይቀርም ብዬ አምናለሁ፡፡

3. ሄሎ ትሰሚኛለሽ ምርጫው ያንችነው፤ ውጤትሽን ለመፃፍ እስክረቢቶው እጅሽ ላይ ነው፤ ስለዚህ ይህን አጉል መግደርደርሽን ትተሽ ሳይረፍድ አስበሽበት ብትደውይልኝ ስልኬ ክፍት ነው፡፡

4. ሄሎ ሄሎ የምልህን ትሰማኛለህ? ሁላችንም ተያይዘን የዘበጥያ መቀመቅ ከመውረዳችን በፊት የመዝገቡ ዱካ የሚጠፋበትን መንገድ ሳትውል ሳታድር ማመቻቸቱን አስብበት፡፡

5. ሄሎትሰማኛለህ? ይህን አጓጉል ሃገር ሃገር የሚል ወግ አጥባቂነትህን ትተህ በምልህ ነገር መስማማቱ ለሁላችንም ይበጃልና ደግሜ እስክ ደውል ነገሬ ብለህ አስብበት፡፡

6. ባክህ ለማያውቅህ ታጠን ድሮም ጥበት የዘራችሁ መገለጫ መሆኑን ጠንቅቄ አውቃለሁ፡፡

7. ነፍጠኝነት የደም ውርሳችሁ መሆኑን መች አጣሁትና፡፡

8. ትሰማኛለህ? ይህ ጉዳይ ከሁለታችን ቢያፈተልክ ያን ጊዜ የህወትህ ፍፃሜ መሆኑን አትዘንጋ፡፡

9. አክብሮቴን ከፍራቻ ቆጥረህ ተፈታትነሽኛል ፤አሁን ግን ትግስቴ ስለተሟጠጠ ደም ከመቃባታችን በፊት አይንህን ከሷ ላይ አንሳ ብዬሃለሁ፡፡

10. ሄሎ ትሰማለህ? አንዲት ፊርማ ለመጫር እጅህ ወባ እንደያዘው እንዲህ ከተንዘረዘረ በቀርቡ የልጅእና ባለቤትህን አስከሬን ስናስታቅፍህ መላ አካልህ ከመንዘፍዘፉ በፊት የምንልህን ተኝተህ አስብበት፤ እያንዳንዷ ሰከንድ ባለፈች ቁጥር የመከራ ዘመንህ መቅረቡን አትዘነጋ፡፡

iii.Paragraph Level Text Data:

1. በዓለም ላይ እጅግ ውድ የሆነ ዋጋ የሚያስከፍለው ነገር እውነት ነው፡፡ ባንፃሩ ውሸት እጅግ በጣም ርካሽ

ነው፡፡ ርካሽነቱ የሚመነጨው ለመቀያየር ስለሚቻል ነው፡፡ መቀያየሪያ እንደልብ የሚገኝለት ዕቃ ርካሽ እንደሚሆ ነው፡፡ እውነት ግን እንኳንስ በሌላ ነገር በሌላ እውነት እንኳን ልትቀየር አትችልም፡፡ እውነተኞች በፈንጅ ላይ እንደሚራመድ ወታደር ናቸው፡፡ እነሱ ተሠዉተው ሌሎች ያለ አደጋ እንዲሻገሩ ያደጋሉ፡፡ ትውልድ መሻገር የሚያቅተው በፈንጅ ላይ ለመራመድ የሚቆርጡ እውነትኛ ሰዎችን ሲያጣ ነው፡፡

2. ሀገር ሰላም እንድትሆን ሰላም ሆናም በብልፅግና ጎዳና እንድትራመድ ከተፈለገ ሁለት ነገሮች ያስፈልጓታል። ህዝብን የሚያዳምጥ መንግስትና መንግስትን የሚያዳም ጥህዝብ። ማዳመጥ ከመስማት ይለያል። መስማት ጆሮ ለተፈጠረለት ሁሉ የሚቻል ነዉ። ይህን ፅሁፍ ስናነብ እንኳ ስንትና ስንት ድምፆችን ፈልገንም ሳንፈልግም እንሰማለን ። ማዳመጥ ግን ሶስት ነገሮችን ይፈልጋል ። ይሁነኝ ብሎ መስማት ሰምቶ ማስተዋል አስተዉሎም መመለስን።

3. ሀገር እንድትሠለጥን፣ እንድትዘምንና የሁሉም እንድትሆን ትግል የሚደረገው በጥሎማለፍ ዋንጫ እየተበሻሸቁ ለመጓዝ አይደለም፡፡ ለውጡ ሁሉንም እንደ ዐቅሙና እንደ ድርሻው ካልጠቀመ፤ አብሻቂና በሻቂ ቡድን ከፈጠረ ‹አገር አጥፉ አረም› ሥር እየሰደደ ነውና ልንነቃ ይገባል፡፡ አረሙን ያጠኑት ሊቃውንት እንደሚሉት ‹አገር አጥፉ › አረም ሥር ከሰደደ በኋላ መንቀሉ ራሱ ሌላ ጉዳት ያመጣል ፡፡ ማጥፋት የሚቻለው በመከላከል ነው፡፡ የተነቀለ ሲመስል እንኳን መርዛማ ዘሩን ምድር ላይ በትኖ ደብቆ ያስቀምጠዋል፡፡ ጊዜ ሲያመቸውም ወረራውን ይጀምራል፡፡

4. በርግጥ ዛፉን መትከል የነበረብን የመጀመሪያው ምርጥ ጊዜ አልፏል፡፡ አሁን ሽሚያው መሆን ያለበት ሁለተኛው ምርጥጊዜ እንዳያልፍ ነው፣ ጉንዳን መኽር እንዲመጣ ስትጸልይ ያያት ዝንጀሮ ‹መኽር ቢደርስ ለሰው እንጂ ላንቺ ምን ያደርግልሻል? › ቢላት ‹ሞኝ፤ ሰውም ያጭዳል ጉንዳንም ይሰበስባል፤ ዋናው መኽሩ ይምጣ› አለቺው ይባላል፡፡ ዋናው መኽር መምጣቱ ነው ፡፡የምንፈልገውን ማድረግ የኛ ድርሻነው፡፡ ለሀገርና ለወገን እንዲጠቅም አድርጎ መኽሩን መሰብሰብ የኛ ፋንታ ነው፡፡ እየደጋገሙ መሸወድና እየደጋገሙ ማልቀስ አመላችን ሆኗል፡፡ ትናንትን ከመርገም እንውጣና የተሻለ ዛሬና የሚናፈቅ ነገን እንገንባ፡፡ ትናንትን በመውቀስ ትናንትን ማሸነፍ አይቻልም ፡፡ ትናንትን ለማሸነፍ የተሻለው ብልሃት ዛሬን በተሻለ መሥራት ነው፡፡ኢትዮጵያ ማለት እኔና እናንተ በምናውቃት ልክ ብቻ ያለች ሀገር አይደለችም። ኢትዮጵያ ከምድር ሀገራት ሁሉ እጅግ ምስጢራዊና ረቂቅ ሀገር ናት። ይሄን ስላችሁ ደግሞ በግምት ወይም በስሜት ተገፍቼ አይደለም። በብዙ ማስረጃዎች ነው እንጅ። ይሄን ጽሁፍ ስታነቡ ውስጣችሁ የማይቀበልላችሁ፣ የኢትዮጵያ ነገር የማይዋጥላችሁ፣ ተራ አሉቧልታ ወይም አፈታሪክ የሚመስላችሁ፣ ጎሊናችሁ ኢትዮጵያን በጠባብነት በዘረኝነት ወይም በገንዘብና በክህደት የታወረ አይን የምትመለከቱ ጥቂቶች ትኖራላችሁ። አልፈርድባችሁም ። ምክንያቱም የኢትዮጵያ ማንነት አልገባችሁምና ነው።

5. የምታገኘው ለሌላው ባደረግከው ልክ ነው፡፡ እባብ ሰጥተህ ርግብ፣ ድንጋይ ሰጥተህ ዳቦ፣ እሬት ሰጥተህ ማር ልታገኝ አትችልም ፤አንተ ብቻ ብልጥ ልትሆን አትችልም ፤ማሾውን አጥፍተህ ብርሃን ልታገኝ አትችልም ፤ሌላው ገድለህ አንተ በሰላም ልትኖር አትችልም፡፡ ማንኛውም ድርጊት ተመሳሳይና ተመጣጣኝ የሆነ ምላሽ አለው፡፡ አንተ እዚህ ለብቻህ ቤትህ ውስጥ ሆነህ ክፉ ስታደርግ ሌላውም በቤቱ ብቻውን ሆኖ ክፉውን ይመልስሃል፡፡ ‹እዛም ቤት እሳት አለ› እንዳሉት አለቃ ገብረ ሐና፡፡

6. የምታገኘው ለሌላው ባደረግከው ልክ ነው፡፡ እባብ ሰጥተህ ርግብ፣ ድንጋይ ሰጥተህ ዳቦ፣ እሬት ሰጥተህ ማር ልታገኝ አትችልም ፤አንተ ብቻ ብልጥ ልትሆን አትችልም ፤ማሾውን አጥፍተህ ብርሃን ልታገኝ አትችልም ፤ሌላው ገድለህ አንተ በሰላም ልትኖር አትችልም፡፡ ማንኛውም ድርጊት ተመሳሳይና ተመጣጣኝ የሆነ ምላሽ አለው፡፡ አንተ እዚህ ለብቻህ ቤትህ ውስጥ ሆነህ ክፉ ስታደርግ ሌላውም በቤቱ ብቻውን ሆኖ ክፉውን ይመልስሃል፡፡ ‹እዛም ቤት እሳት አለ› እንዳሉት አለቃ ገብረ ሐና፡፡

7. አስቀድሜ የሃገራችን መሪ ከሆኑበት ጊዜ አንስቶ ለሰሯቸው ስራዎች እና ላከወኗቸው መልካም ተግባራት ያለኝን ጥልቅ አክብሮት ለመግለፅ እወዳለሁ፡፡ በተለይ ከሶስት ወራት በፊት በማንም የኢትዮጲያዊም ሆነ ኤርትራዊ ዜጋ

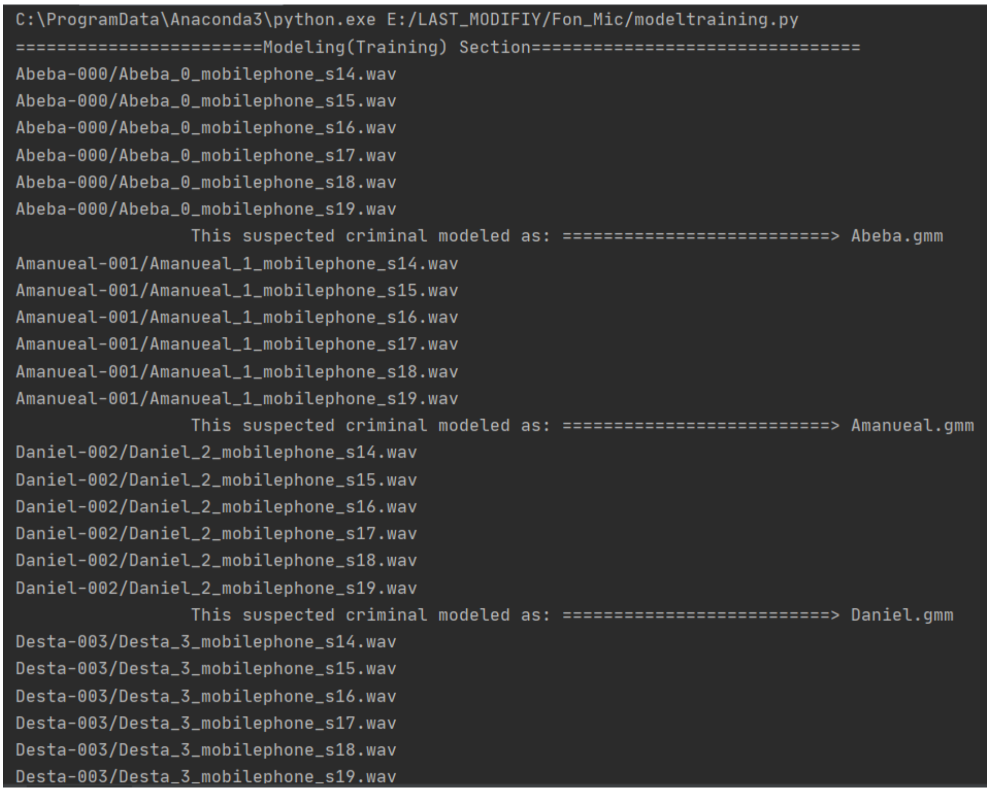
የማይቻል እና የማይሳካ ይመስል የነበረውን የሁለቱን ሀገሮች ዕርቅ ጉዳይ የፈቱበት መንገድ መብረቃዊ የሚባል እና ምንጊዜም ከስሞት ጋር ተያይዞ ሲዘከር የሚኖር ማንም ሊፍቀው የማይችል አንፀባራቂ ስኬቶ ነው፡፡ ለዚህም ታሪኮ የሁለቱም ሃገር ህዝቦች በልባቸው ከፃፉት ቀጥሎ ብዙ ቁሳዊ እና ስነጥበባዊ ማስታወሻዎች በማኖር ሲዘክሩት እንደሚኖሩ እርግጠኛ ነኝ፡

8. የህዝብን ኽንጉርጉነት፤ ማስወገድ አይቻልም፡፡ ከሰፊው ህዝብ ውስጥ፤ ጥቂት ስለህዝብ የሚሠው ይፈልቃሉ፡፡ በአንጻሩ ከሰፊው ህዝብ ውስጥ፤ ጥቂት ግለሰቦች ለራስ ጥቅም የሚሸነፉ፤ ህዝብ ሊያጠፉ የሚችሉ ይፈጠራሉ፡፡ የግጭት ወይም የጦርነት ተዋናዮች ቢያንስ አንዱ ግንባር ወይም አንዱ ተጋጣሚ ህዝባዊነት ስሜት አይኖረውም፡፡ አንዳንዴ እንደ ዕድል ሆኖ ሁለቱ ግንባር የህዝባዊነት ስሜት የሌላቸው ሊሆኑ ይችላሉ፡፡

9. ሁለት ጎሳዎች በአደንና በእርሻ ቦታ ወይም በድንበር ምክንያት በተፈጠረ ችግር ወደ ጦርነት ይገባሉ፡፡ የእነዚህ ጥንታውያን ሕዝብ ሽማግሌዎች ግን ጦርነቱ በአንደኛው አሸናፊነት እንዲጠናቀቅ አይፈቅዱም፡፡ ጦርነቱ ሲደረግ ለተወሰነ ጊዜ ብቻ ሆን ብለው ይታገሡታል፡፡ የዚህን ምክንያቱን ሲያስረዱ ደግሞ ‹ሁለቱም ወገኖች የጦርነቱን አስከፊ ገጽታ መቅመስ አለባቸው› ይላሉ፡፡ የችግር መፍቻ መንገዳቸው ወደባሰ ችግር እየወሰዳቸው መሆኑን ከምክርና ከትምህርት ይልቅ በተግባር እንዲያዩት ጊዜ ይሰጧቸዋል፡፡ በኋላ ግን በመካከል ይገባሉ፡፡ ‹ጦርነቱ በአንደኛው ወገን አሸናፊነት መጠናቀቅ የለበትም ›የሚል እምነት አላቸው፡፡ ሁለቱም ወገኖች ቀን ጣላቸው፤ ችግር ለያያቸው፤ መንገድ አጣላቸው እንጂ ወንድማማቾች ናቸው፡፡ የእነዚህ ወንድማማቾች ትግል በአንደኛው አሸናፊነት ከተጠናቀቀ ሰላም አይገኝም፡፡ አሸናፊው ጨቋኝ፣በቀለኛ፣ጉልበተኛና ዘራፊ ሆኖ ይቀራል፡፡ ተሸናፊው ደግሞ ቂመኛ፣ ቀን ጠባቂ፣ በጥላቻ የተሞላና ባዕድ ሆኖ ይኖራል፡፡ የጊዜ ጉዳይ ይሆናል እንጂ አሸናፊና ተሸናፊ ቦታ መቀያየራቸው አይቀርም፡፡ የሕዝቡም ችግር ይቀጥላል፡፡

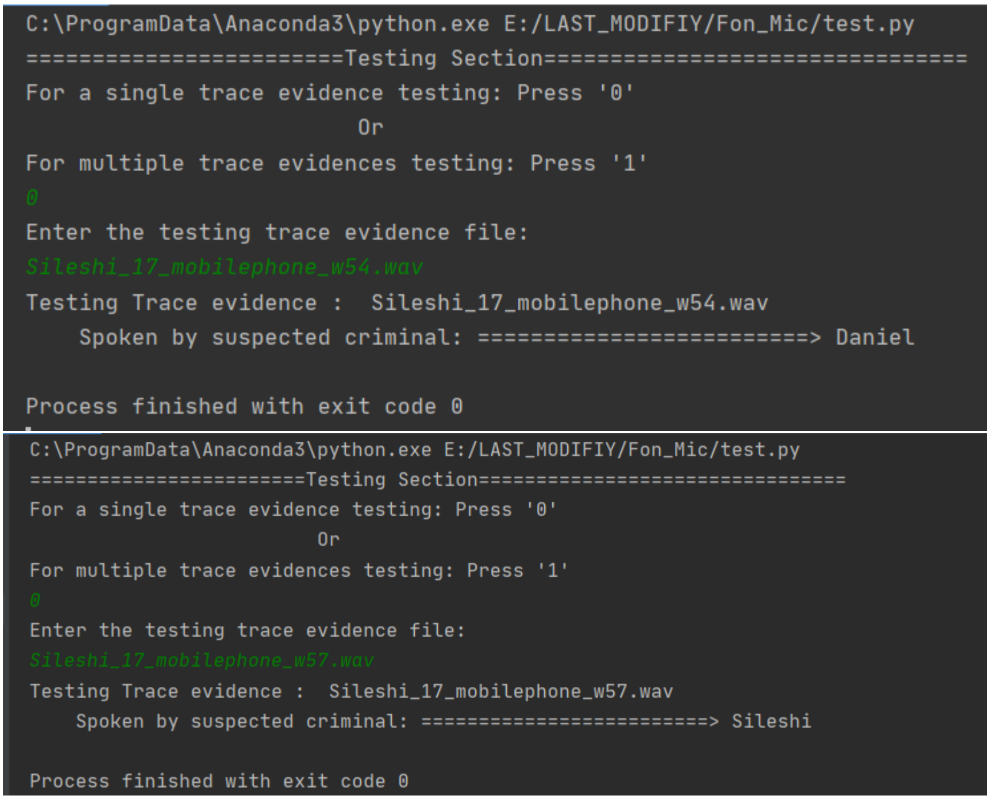
10. አንድ ሕዝብ ታስተዳድረኛለህ ብሎ የመረጠውን መንግስት ማንነትና ምንነት በብዕሩም ሆነ በአንደበቱ ለመግለጽ ይፋዊ መብት ሊኖረው እንደሚገባ ግልጽ ነው። ይህን ሃቅ የሰለጠኑትም ሆኑ የ3 ኛ ዓለም ሃገሮች መንግስታት ፉርሽ አይሉትም ። ጥያቄው ያለው በ3 ኛው ዓለም በተለይም በአፍሪካ የሚገኙ ሃገሮች ሕዝብ የሚያስተዳድረውን መንግስት የመምረጥ እድል አለው ወይ? የሚለው ነጥብ ላይ ይመስለኛል። መልሱ ደግሞ የለም ነው። በአፍሪካ ሃገሮች መንግስታት የሚያስተዳድሩትን ህዝብ ይመርጣሉ እንጂ ህዝብ የሚያስተዳድረውን መንግስት ሲመርጥ አልታየም።

Appendix B: Sample Training and Testing screenshot and Results



```
C:\ProgramData\Anaconda3\python.exe E:/LAST_MODIFIY/Fon_Mic/modeltraining.py
========================Modeling(Training) Section================================
Abeba-000/Abeba_0_mobilephone_s14.wav
Abeba-000/Abeba_0_mobilephone_s15.wav
Abeba-000/Abeba_0_mobilephone_s16.wav
Abeba-000/Abeba_0_mobilephone_s17.wav
Abeba-000/Abeba_0_mobilephone_s18.wav
Abeba-000/Abeba_0_mobilephone_s19.wav
                 This suspected criminal modeled as: ==========================> Abeba.gmm
Amanueal-001/Amanueal_1_mobilephone_s14.wav
Amanueal-001/Amanueal_1_mobilephone_s15.wav
Amanueal-001/Amanueal_1_mobilephone_s16.wav
Amanueal-001/Amanueal_1_mobilephone_s17.wav
Amanueal-001/Amanueal_1_mobilephone_s18.wav
Amanueal-001/Amanueal_1_mobilephone_s19.wav
                 This suspected criminal modeled as: ==========================> Amanueal.gmm
Daniel-002/Daniel_2_mobilephone_s14.wav
Daniel-002/Daniel_2_mobilephone_s15.wav
Daniel-002/Daniel_2_mobilephone_s16.wav
Daniel-002/Daniel_2_mobilephone_s17.wav
Daniel-002/Daniel_2_mobilephone_s18.wav
Daniel-002/Daniel_2_mobilephone_s19.wav
                 This suspected criminal modeled as: ==========================> Daniel.gmm
Desta-003/Desta_3_mobilephone_s14.wav
Desta-003/Desta_3_mobilephone_s15.wav
Desta-003/Desta_3_mobilephone_s16.wav
Desta-003/Desta_3_mobilephone_s17.wav
Desta-003/Desta_3_mobilephone_s18.wav
Desta-003/Desta_3_mobilephone_s19.wav
```

Sample of Modeling(Training)



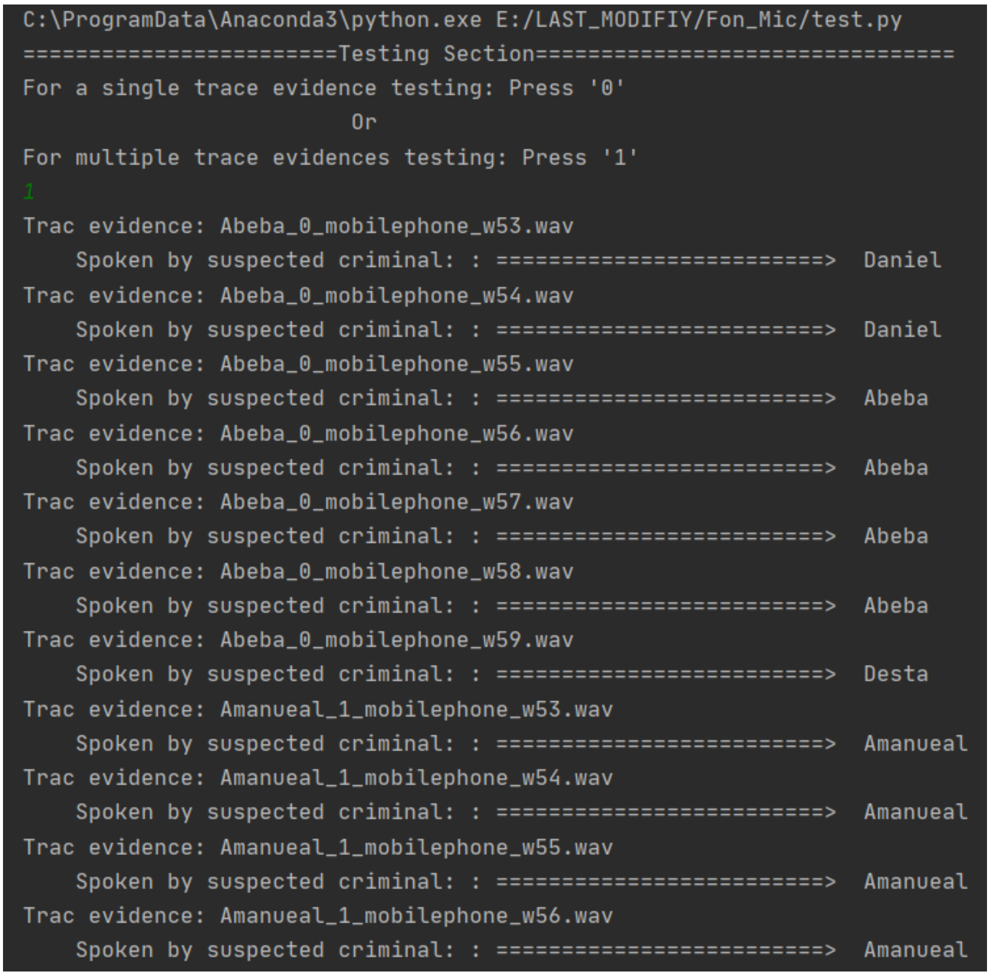
```
C:\ProgramData\Anaconda3\python.exe E:/LAST_MODIFIY/Fon_Mic/test.py
========================Testing Section=============================
For a single trace evidence testing: Press '0'
                         Or
For multiple trace evidences testing: Press '1'
0
Enter the testing trace evidence file:
Sileshi_17_mobilephone_w54.wav
Testing Trace evidence :  Sileshi_17_mobilephone_w54.wav
    Spoken by suspected criminal: =========================> Daniel

Process finished with exit code 0
```

```
C:\ProgramData\Anaconda3\python.exe E:/LAST_MODIFIY/Fon_Mic/test.py
========================Testing Section================================
For a single trace evidence testing: Press '0'
                         Or
For multiple trace evidences testing: Press '1'
0
Enter the testing trace evidence file:
Sileshi_17_mobilephone_w57.wav
Testing Trace evidence :  Sileshi_17_mobilephone_w57.wav
    Spoken by suspected criminal: =========================> Sileshi

Process finished with exit code 0
```

Sample Screenshot for single trace evidence testing



```
C:\ProgramData\Anaconda3\python.exe E:/LAST_MODIFIY/Fon_Mic/test.py
========================Testing Section=================================
For a single trace evidence testing: Press '0'
                         Or
For multiple trace evidences testing: Press '1'
1
Trac evidence: Abeba_0_mobilephone_w53.wav
    Spoken by suspected criminal: : =========================>  Daniel
Trac evidence: Abeba_0_mobilephone_w54.wav
    Spoken by suspected criminal: : =========================>  Daniel
Trac evidence: Abeba_0_mobilephone_w55.wav
    Spoken by suspected criminal: : =========================>  Abeba
Trac evidence: Abeba_0_mobilephone_w56.wav
    Spoken by suspected criminal: : =========================>  Abeba
Trac evidence: Abeba_0_mobilephone_w57.wav
    Spoken by suspected criminal: : =========================>  Abeba
Trac evidence: Abeba_0_mobilephone_w58.wav
    Spoken by suspected criminal: : =========================>  Abeba
Trac evidence: Abeba_0_mobilephone_w59.wav
    Spoken by suspected criminal: : =========================>  Desta
Trac evidence: Amanueal_1_mobilephone_w53.wav
    Spoken by suspected criminal: : =========================>  Amanueal
Trac evidence: Amanueal_1_mobilephone_w54.wav
    Spoken by suspected criminal: : =========================>  Amanueal
Trac evidence: Amanueal_1_mobilephone_w55.wav
    Spoken by suspected criminal: : =========================>  Amanueal
Trac evidence: Amanueal_1_mobilephone_w56.wav
    Spoken by suspected criminal: : =========================>  Amanueal
```

Sample screenshot for multiple trace evidence testing with a single execution

Table 1: Sample result for word level utterance trace evidence

Testing Trace	Suspected Criminal speaker	Identified Criminal	Identification Rate(IDR)
Elbetel_4_mobilephone_w53.wav	Elbetel	Elbetel	42.86%
Elbetel_4_mobilephone_w54.wav	Elbetel	**Desta**	
Elbetel_4_mobilephone_w55.wav	Elbetel	Elbetel	
Elbetel_4_mobilephone_w56.wav	Elbetel	**Desta**	
Elbetel_4_mobilephone_w57.wav	Elbetel	Elbetel	
Elbetel_4_mobilephone_w58.wav	Elbetel	**Wubrist**	
Elbetel_4_mobilephone_w59.wav	Elbetel	**Msganaw**	

Table 2: Sample result for Sentence level utterance trace evidence

Testing Trace	Suspected Criminal speaker	Identified Criminal	Identification Rate(IDR)
Elbetel_4_mobilephone_s10.wav	Elbetel	**Rediet**	50.0%
Elbetel_4_mobilephone_s11.wav	Elbetel	Elbetel	
Elbetel_4_mobilephone_s12.wav	Elbetel	Elbetel	
Elbetel_4_mobilephone_s13.wav	Elbetel	**Wubrist**	

Table 3: Sample result for paragraph level utterance trace evidence

Testing Trace	Suspected Criminal speaker	Identified Criminal	Identification Rate(IDR)
Elbetel_4_mobilephone_p6.wav	Elbetel	Elbetel	75.0%
Elbetel_4_mobilephone_p7.wav	Elbetel	Elbetel	
Elbetel_4_mobilephone_p8.wav	Elbetel	**Desta**	
Elbetel_4_mobilephone_p9.wav	Elbetel	Elbetel	