VOICE TO TEXT CONVERSION APP WITH SPEAKER RECOGNITION

By

Ang Sea Zhe

A REPORT

SUBMITTED TO

Universiti Tunku Abdul Rahman

in partial fulfillment of the requirements

for the degree of

BACHELOR OF COMPUTER SCIENCE (HONOURS)

Faculty of Information and Communication Technology

(Kampar Campus)

JAN 2022

UNIVERSITI TUNKU ABDUL RAHMAN

REPORT STATUS DECLARATION FORM

Title: VOICE TO TEXT CONVERSION APP WITH SPEAKER RECOGNITION

Academic Session: JAN 2022

I ANG SEA ZHE

(CAPITAL LETTER)

declare that I allow this Final Year Project Report to be kept in

Universiti Tunku Abdul Rahman Library subject to the regulations as follows:

1. The dissertation is a property of the Library.
2. The Library is allowed to make copies of this dissertation for academic purposes.

(Author's signature)

Address:
32, Jalan Kangar 86, Kawasan
19 Off Jalan Kapar 41400,
Klang, Selangor.

Date: 20/4/2022

Verified by,

(Supervisor's signature)

Tou Jing Yi

Supervisor's name

Date: 22/4/2022

FACULTY OF INFORMATION AND COMMUNICATION TECHNOLOGY

UNIVERSITI TUNKU ABDUL RAHMAN

Date: ____20/4/2022_____

SUBMISSION OF FINAL YEAR PROJECT

It is hereby certified that _____ ***Ang Sea Zhe*** _____ (ID No: __***18ACB01470***__) has completed this final year project entitled "__Voice To Text Conversion App With Speaker Recognition __" under the supervision of ____Mr. Tou Jing Yi _____ (Supervisor) from the Department of ____Computer Science____, Faculty of _Information and Communication Technology__ .

I understand that University will upload softcopy of my final year project / dissertation/ thesis* in pdf format into UTAR Institutional Repository, which may be made accessible to UTAR community and public.

Yours truly,

_____

Ang Sea Zhe

*Delete whichever not applicable

DECLARATION OF ORIGINALITY

I declare that this report entitled "**VOICE TO TEXT CONVERSION APP WITH SPEAKER RECOGNITION**" is my own work except as cited in the references. The report has not been accepted for any degree and is not being submitted concurrently in candidature for any degree or other award.

Signature : 

Name : ANG SEA ZHE

Date : 20/4/2022

ACKNOWLEDGEMENTS

I would like to express my sincere thanks and appreciation to my supervisors, Mr. Tou Jing Yi who has given me this bright opportunity to engage in a research-based project on speech recognition. It is my first step to establish a career in IT field. A million thanks to you.

Finally, I must say thanks to my parents and my family for their love, support and continuous encouragement throughout the course.

ABSTRACT

This project is a final year project of a computer science student. Voice recognition is a field that is still quite underdeveloped. There are still a lot of obstacles that are needed to be overcome before voice recognition system can identify all the speakers correctly under all kind of conditions. This would be helpful in speaker verification field, and also a speech recognition system that is personalized to the user.

In this paper, Mel Frequency Cepstral Coefficient and delta of it are used to describe the vocal traits of a person. Mel Frequency Cepstral Coefficient is popular in this field to describe the phenomes of voice. After that, Gaussian Mixture Model is used to represent each speaker or each pair of speakers.

In the first part of experiments using self-generated datasets, the total number of users that are tested in this paper is 5. 25 voice recordings, where 5 of them belongs to each speaker are used as the input to the system for single speaker identification. For two simultaneous speaker identifications, 65 voice recordings where 40 of them are artificially mixed are used as the input to the system for two simultaneous speaker identifications.

In the second part of experiments using LibriSpeech datasets, the total number of users that are tested in this paper is 20 including me and speakers from LibriSpeech dataset. There are a total of 20 single speaker models. Not only that, 5 single word speaker models are trained to detect each short word is spoken by who. Finally, a Universal Background Model is also built for speaker verification. All the models are built using Gaussian Distributions Technique. The experiments that are done in second part of experiments are single speaker identification, non-overlapped multi-speaker identification with speech extraction with known speakers, speaker verification and speaker verification with noise estimation and speech extraction with unknown which is the main part of the project, Voice To Text Conversion With Speaker Recognition.

TABLE OF CONTENTS

LIST OF FIGURES

Figure **Page**

LIST OF TABLES

Table **Page**

LIST OF ABBREVIATIONS

DSP	Digital Signal Processing
SOT	Serialized Output Training
TS-ASR	Target-Speaker Automatic Speech Recognition
MFCCs	Mel Frequency Cepstral Coefficients
LSTM	Long short-term memory
IDE	Integrated Development Environment
GMM	Gaussian Mixture Model
UBM	Universal Background Model
DER	Diarization Error Rate
VTTSR	Voice To Text With Speaker Recognition (The main project app)

CHAPTER 1: INTRODUCTION

1.1 Problem Statement and Motivation

There are already a lot of existing speech recognition available on the market. However, their functionality drops when there are a lot of background noises. The speech conversion will also take in other people's voice in the background, this will cause difficulty for the user that want to use the speech conversion function as the message will become jumbled up due to others speech getting recognized as well.

In order to solve this problem, the speech recognition system needs to be able to identify and remember the user's voice. By recognizing user's voice, speech recognition system will isolate the speaker voice with others voice when it is recording sound. In order to achieve this, digital signal processing (DSP) technique is needed to be apply. It required user's voice recording and also a few different speakers in order to do training and features extraction to create a model that will extract the targeted speaker voice from the speech recognition. With the speaker identification system, the user is able to enrol their voice into the system and register it. Then the system will extract the speech only from the targeted speaker's voice from the overlapped speech and supress other kinds of noise.

Speech recognition nowadays is already pretty advance, however, the ability to recognize by targeted speaker's voice is still a challenge even until now. This project to create a targeted-speaker speech recognition system will be useful in a conversation. For example, when two foreign peoples are talking with each other using a speech recognition system that helps with translation, a targeted-speaker speech recognition system will be able to only recognize and translate the registered user voice. This will fasten the conversation and make the conversation become easier. Not only that, the ability to extract speech based on speaker will also be useful when an organisation wants to transcribe in a meeting.

1.2 Project Scope

The final product that will be delivered at the end of the project is a targeted-speaker speech recognition system. In order for the targeted-speaker speech recognition system to work, the modules as follow are needed to be completed:

- Speech recognition module
  - This module is responsible for recognizing speech.
- Speaker identification model
  - This model is responsible for identifying the difference between the voice of the user and peoples' voice in the background.
- Speaker extraction module
  - This module is responsible to extract only speech from target speaker.
- Noise estimation module
  - This module is responsible to estimate and predict the noise speech wanted to be suppressed.

1.3 Project Objectives

This project aims to research on voice to text conversion with speaker recognition. It will be able to recognize only the targeted-speaker speech from an overlapped speech. This project consists of 3 main objectives which are:

- To develop a model that can recognize speaker voice with audible background noise such as speech.
- To determine the background noise to be suppressed to improve voice recognition using voice feature extraction.
- To identify the speaker voice by utilizing Mel Frequency Cepstral Coefficient (MFCC) features.

1.4 Impact, Significance and Contribution

By proposing this targeted-speaker speech recognition algorithm, a system can be implemented so that the user will be able to use speech recognition system under more condition, and it will work when a conversation is ongoing at the same time. The proposed method for the targeted-speaker speech recognition will be able implement in a real system in the future. After registering a target user voice, it will extract only the target user voice to be transcribed into text or by registering multiple users, so that it will be separating the speech into speech by different speaker.

Furthermore, this project also will help to advance the technique in speech recognizing targeted-speaker even more. The advancement of this technique will benefit a lot of people. It will not only make transcribing a meeting conversation easier, it might can even be applied to entertainment industry by transcribing a stream or a movie in real time where the speakers' subtitles will be differentiated.

1.5 Background Information

As the processing power becomes faster, combined with massive amounts of speech data, speech recognition has reached a state that its capabilities are roughly equal with humans. Figure 1.1 below show that Google's word accuracy rate recently has broken 95% threshold for human accuracy.

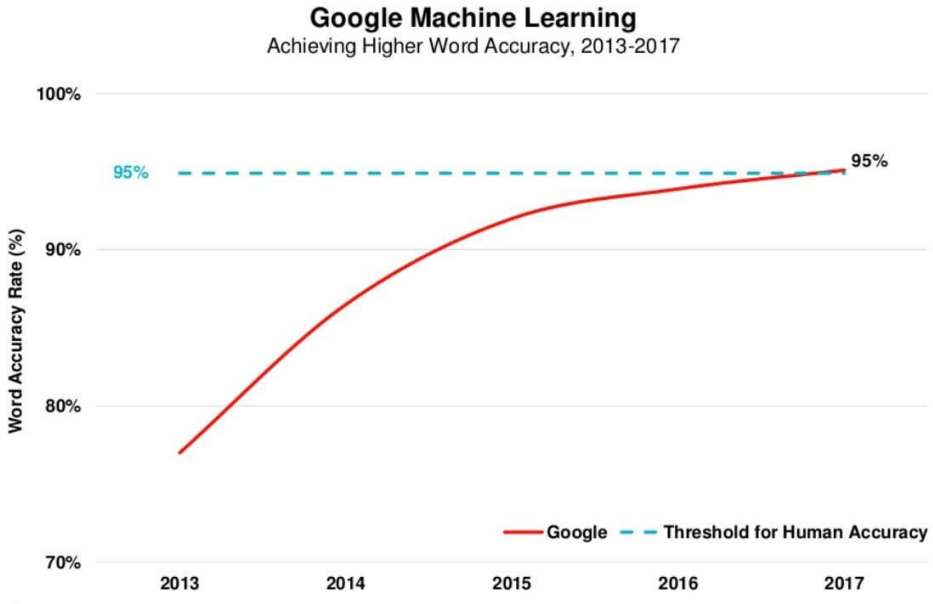


Figure 1.1: Google Machine Learning Accuracy Improvement

CHAPTER 2: LITERATURE REVIEW INTRODUCTION

Speech recognition has come a long way. The first speech recognition systems called "Audrey" system were created in 1952 and it only focused on numbers rather than words. In 1970s, Carnegie Mellon's "Harpy' speech system was created and it was able to understand over 1000 words. In 1980s, the statistical method, "Hidden Markov Model (HMM)" is used to estimate the probability of the sounds being words. In 2000S, google voice search was introduced. In 2010S, Apple's Siri, Amazon's Alexa and Google Home has shown that speech recognition has improved [1]. Speech recognition has improved over the year greatly, however there are still a lot of diverse challenges needed to be overcome. One of the critical problems is that speech recognition only function properly in a controlled environment, where only the user talks and no background noise such as others people voice. Speech recognition system functionality tends to drop when it is used in a noisy environment.

1.6 Report Organization

This report is organized into 5 chapters: Chapter 1 Introduction, Chapter 2 Literature Review, Chapter 3 System Model, Chapter 4 Experiment and Evaluation, Chapter 5 Conclusion. Chapter 1 is introduction which includes problem statement and motivation, project scope, project objectives, impact, significance and contribution, background information and report organization. Chapter 2 is on literature review on past paper related to speech recognition. Chapter 3 is on system model that include tools used, method of implementation and challenges met. Chapter 4 is on experiment and evaluation on the performance. Chapter 5 is about the conclusion and recommendation for future works.

CHAPTER 2: LITERATURE REVIEW

2.1 Speaker-attributed Automatic Speech Recognition Model

There are already tons of researches regarding speaker-attributed automatic speech recognition before this research. There are approaches like applying speech separation before speech recognition and speaker identification. Other than that, there are also past researches that try to improve by combining multiple modules together. However, there is only some researches regarding combining all the modules for speaker-attributed automatic speech recognition.

An end-to-end speaker-attributed automatic speech recognition model that unifies speaker counting, speech recognition, and speaker identification on monaural overlapped speech was proposed [2]. They built their model on serialized output training (SOT) with attention-based encoder-decoder, a recently proposed method for recognizing overlapped speech comprising an arbitrary number of speakers. Not only that, they extended the SOT model by introducing a speaker inventory to become an additional input in order to produce speaker labels and also multi-speaker transcriptions. This model not only can recognize overlapped speech of any number of speakers, it also has the ability to identify the speaker of each utterance among any number of speaker profiles at the same time. The proposed models had achieved significantly better word error rate when comparing with model that consists of separated modules rather than combined modules. However, there is a critical weakness in their model, the authors' model only works if all the speaker profiles are included in the speaker inventory in order for the system to distinguish between speaker. This means that they assumed that there will be no "unknown" speaker. This means that in order for the speech recognition system to function as they want in an environment, the system must have registered all the profile of the speakers that are in that environment. This would make the system unable to function in an unknown environment.

In order to overcome this limitation, using iterative method where estimation of target-speaker embeddings and target-speaker automatic speech recognition based on the estimation are being executed alternately [3] can works with unknown speakers.

2.2 Target-Speaker Automatic Speech Recognition (TS-ASR) model

Kanda et al. has investigated the use of target-speaker automatic speech recognition (TS-ASR) for simultaneous speech recognition and speaker diarization for only a single channel monaural dialogue in [3]. The past problem with TS-ASR is that training sample of targeted speakers must exist prior hand the transcription process. In order to solve this problem, they proposed a repetitive method whereby estimation process for speaker embeddings and TS-ASR process based on the estimation produced take turns.

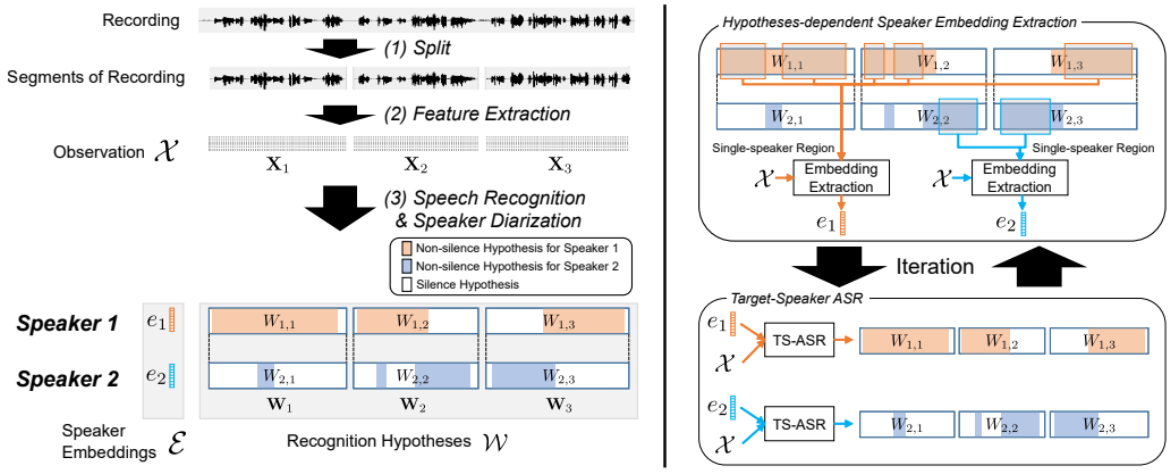


Figure 2.1: (left) Overview of simultaneous speech recognition and speaker diarization, (right) Proposed iterative method. [3]

According to Figure 2.1, first of all the recording undergo the usual step which are splitting and feature extraction. After that, the output X undergo simultaneous speech recognition and speaker diarization to separate the speech. The proposed method is that single-speaker region is used for estimation of characteristic extraction of one speaker, then it is used in TS-ASR process to produce output, then the output is used to estimate again. This process will keep on iterating until the result is out. The strength of this proposed method is that it works even with unknown speakers. There are prove that their model can significantly reduce error in a real dialogue recording where there are over 20% overlap. However, the model was not evaluated using recordings with more than two speakers and used speaker embeddings that are more non discriminative.

In order to overcome this limitation, only the user should be taken into account in speech recognition system. The method should focus more on maintaining only the part of the speech by the user and take in the user data as training sample. The TS-ASR model should also use more discriminative speaker embeddings such as d-vector [4] and x-vector [5].

Other than that, there was also a VoiceFilter-Lite model developed by Wang and others in [6]. In voice filtering, also known as speaker extraction, the identity of the target speaker is known, and the aim is to extract the speech of that targeted speaker. This voice filtering model process the signal frame-by-frame to enhance the features consumes by speech recognizer and it does not reconstruct audio signals from the features.

The target speaker is represented by either a one-hot vector from a closed speaker set or it also represented by a speaker-discriminative voice embedding like i-vector or d-vector. VoiceFilter-Lite model focus more on improving the automatic speech recognition. In this paper, the automatic speech recognition models take stacked log Mel-filterbank energies as input. The automatic speech recognition is integrated with VoiceFilter-Lite model.

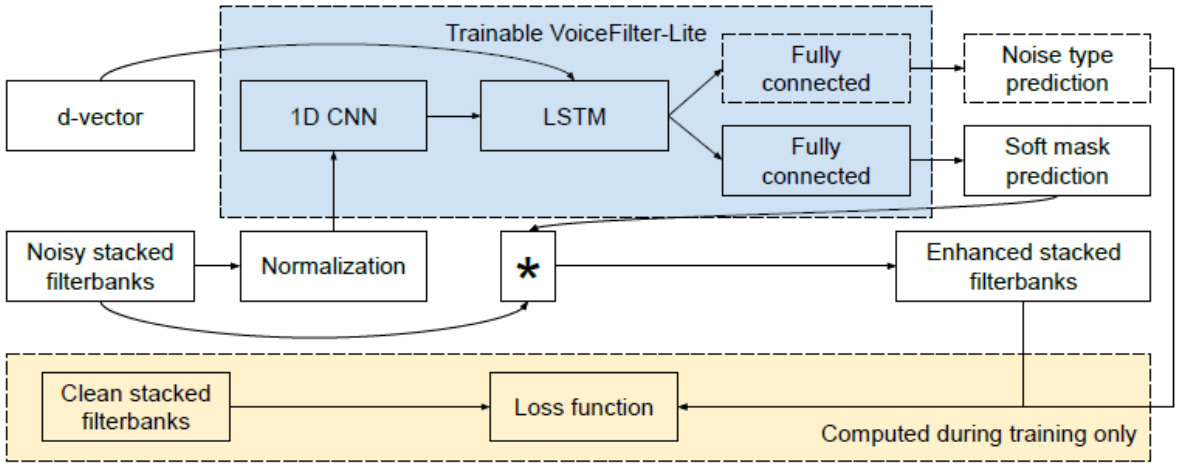


Figure 2.2: Neural network topology of the VoiceFilter-Lite model. [6]

Noisy audio to be enhanced was used and reference audio from the targeted speaker as input for the VoiceFilter-Lite System. A loss function which measures the difference between the clean spectrogram and masked spectrogram are used to train the VoiceFilter-Lite network. Asymmetric loss to overcome over-suppression problem was also proposed in [6]. Asymmetric loss is less tolerant to over-suppression but more tolerant to under-suppression. This proposed method highlight is that it has adaptive suppression strength where the output is not only enhanced spectrogram, it takes weighted average between enhanced spectrogram and original input spectrogram. If there is overlapped speech, the weight should be higher to make the VoiceFilter-Lite more helpful. However, if there is no overlapped speech the weigh should be lower to prevent over-suppression. In order to calculate this weight, noise type prediction is needed as side output of the model.

The strength of VoiceFilter-Lite model is that it uses the target speaker enrolled voice to improve automatic speech recognition on overlapped speech. Not only that, the model is tiny, fast, streaming and part of on-device Automatic Speech Recognition. In order to achieve smaller size, 1D Convolutional Neural Network is used, therefore it is not as powerful as using 2D Convolutional Neural Network.

2.3 Speaker Identification and Verification

There was a speaker identification model done in Matlab in [7]. The purpose of the model is to be combined with speech recognition system and function as an application. Before the speaker identification process, the voice signal must be prepared and processed before extraction Mel Frequency Cepstral Coefficients (MFCCs) characteristics. The voice signal will first undergo signal noise and silence periods elimination. Then, process the voice with framing and window function in order to prevent troubles due to fast changes in signals in extremes of every voice frames. Then finally reach the MFCCs stage.

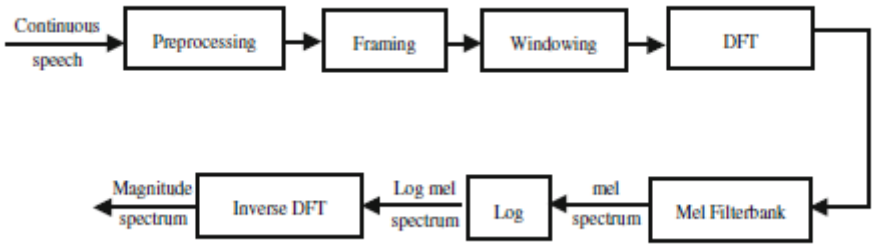


Figure 2.3: MFCCs extraction algorithm. [8]

The strength of this method is that the system is able to identify the registered speaker in automatic speech recognition. However, because of automatic speaker recognition is still currently underdeveloped, the speaker recognition strength might not be enough for the identification to reach lower error rate, in order to improve the speaker recognition quality, expensive products such as KIVOX360, KIVOX Passive Detection, BATVOX, BS3, ASIS and SIFT should be used if want to create a higher quality system.

In Text-Independent Speaker Identification Through Feature Fusion and Deep Neural Network [9], useful features were extracted from training samples to form a master feature vector. The deep neural network architecture uses this master feature vector as input to create the model for the speaker identification. The overview of this method is that at first, multiple voice input samples is used as input signals. Then, it undergoes MFCCs feature extraction and binning in time-domain features to become feature vector to construct the speaker identification model. The testing data which will be identified will also pass through the same pre-processing and feature extraction steps before classifying with the constructed identification model. The speciality of this paper is that, MFCCT features was proposed. MFCCT is time-domain feature extraction from MFCC features. The detailed steps of the extraction steps are (1) MFCCs feature extraction, (2) time-domain feature extraction from MFCCs features and lastly (3) the

target identifier will be appended using the extracted features of each speaker utterance. In (2), firstly binning was performed on extracted MFCC features for every 1500 rows of each column, then 12 different time-domain features were extracted from each bin of extracted MFCC features. This paper also uses hierarchical classification approach. The hierarchical classification is in cascading style. The first-level classify the gender while second-level classify into target identifier. The proposed MFCCT features has high accuracy of about 83.5%-93%.

Furthermore, in Speaker Identification in Multi-Talker Overlapping Speech Using Neural Networks [10], the authors had tried to identify the speaker in an overlapped speech. One of the proposed models is a single-stage overlapping speaker identification system using neural network.

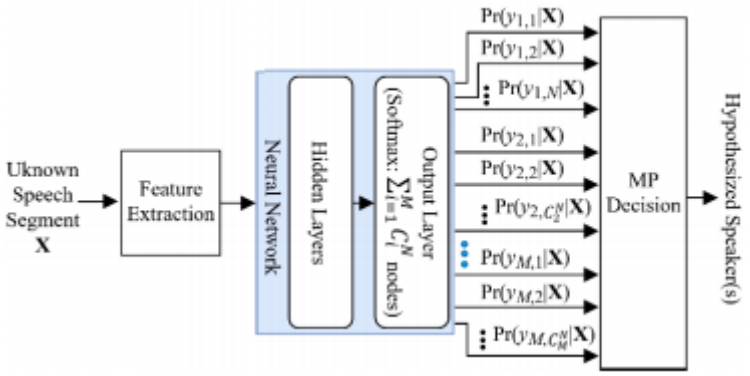


Figure 2.4: Single-stage overlapping speaker identification system. [10]

The authors' model predicts the output based on probability from features extracted from input speech which is the unknown speech segment. The authors had use neural network that have output layer of $\sum_{i=1}^{M} C_i^N$ nodes where it is made up of different combination of speakers, M is the number of simultaneous speakers, and N is the number maximum speakers in the inventory. The training data used is by enrolling each speaker voice, however it is impractical to collect data from all simultaneous speech combination from all the speaker. Therefore, the authors artificially mix the enrolled individual speaker voice to form the overlapping speech for training data. Not only that, the authors also evaluate performance on speech with noise by using this mixing clean audio with noise. The authors have tried the method using 1D-convolutional neural network, multi-layer perceptron and also gaussian mixture models. The authors' main method which is using 1D-convolutional neural network has a flaw because deep learning is very bias toward the training data. Moreover, deep learning is a very heavy process.

In order to overcome this limitation, support vector machine may be a better choice as it is not so heavy compare to deep learning method and it can be optimized for towards real data.

For speaker verification, a Gaussian Mixture Model - Universal Background Model (GMM-UBM) based model can be use as stated in [11]. There are two models built which are speaker model and Universal Background Model. The test audio will undergo feature extraction and will be used to determine the level of match with the speaker model or the universal background model. The similarity can be calculated by taking log-likelihood with speaker model subtracted by log-likelihood with UBM. Then a threshold can be set, to accept or reject the similarity result. For UBM, a high-order Gaussian Mixture Model (usually 512 to 2048 mixtures) need to be trained on a large quantity of speech to represent the unknown speaker.

CHAPTER 3: SYSTEM MODEL

3.1 Development Tools

3.1.1 Python

Figure 3.1: Python

For this project, python is the suitable programming language. This is because:

- Python is simple and consistent.
  - Python code is easier to be readable, therefore it is easier to build model when doing machine learning.
- Python have more selection of libraries and frameworks
  - Libraries and frameworks can reduce difficulty in implementing algorithms, in this project, PyTorch library will be used for deep learning.
- Python is also platform independence
  - This project is to develop a mobile app

3.1.2 Pycharm

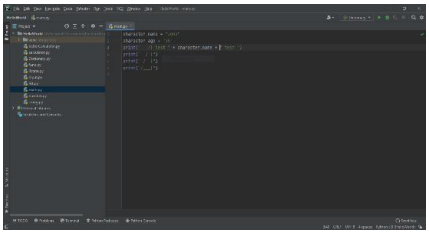


Figure 3.2: Pycharm Interface

PyCharm makes it easier to write program in python. PyCharm free community edition provide smart code completion, code checking, automated code refactoring. All this help in doing large projects more easily.

3.1.3 Google Cloud Speech-To-Text API

Google Cloud Speech-To-Text API is a google cloud service that is used to help in recognizing speech and speech diarization in this project. Even though, this API only give free 60 min usage every month, due to having new google cloud account, 300-dollar free credits were given to spend. Therefore, the quota doesn't really matter with the free credits.

3.1.4 FFmpeg

FFmpeg is a software that is a complete, cross-platform solution to record, convert and stream audio and video. This software does not have graphical user interface and it was accessed with command prompt and such. This software is used to normalize the sampling rate and also file format of the audios.

3.1.5 Python Libraries

Below are all the python libraries that are used to aid in this research project.

- pickle - this library is for serializing and de-serializing a Python object structure.
- pyaudio - this library is for input and output for audio.
- warnings - this library is for ignoring some condition and exception.
- numpy - this library is for array operation.
- sklearn - this library is library for machine learning and metrics to evaluate machine learning.
- scipy - this library is library for reading audio file.
- python_speech_features - this library is for audio feature extraction such as MFCC.
- collections, contextlib, sys, webrtcvad - these libraries are for silence removal.
- pydub - this library is for the audio segment type that will help in slicing the audio and do operation on the audio.
- itertools - this library is for generating combinations of audio.
- speech_recognition - this library is for the pre-trained model for recognizing speech.
- noisereduce - this library is for reducing noise.
- google_cloud - this library is for using the google cloud speech to text API.
- math - this library is for using math operation.

- statistics - this library is for using statistical function.
- simpleder - this library is for evaluating DER.
- Matplotlib.pyplot - this library is to plot graph.

3.1.6 System Specification

Brand	Acer Aspire E14 E5-475G-50N0
CPU	Intel® Core™ i5-7200U 2.5GHz with Turbo Boost up to 3.1GHz
GPU	NVIDIA® GeForce® 940MX with 4GB Dedicated VRAM
RAM	8GB DDR4 Memory
STORAGE	128GB SSD + 1TB HDD
OPERATING SYSTEM	Windows 10 Home

Table 3.1: System Specification of the Experiment Setup

3.2 Methodologies

For this research project, it is separated into two parts. First parts of experiments are done during final year project I while the second parts of experiments are done during final year project II. All modules in the experiments have its own planning, analysis and research phase. In first parts of the experiments, the experiments done are single speaker identification and overlapped two-speaker identification. In the second parts of the experiments, the experiments done are single speaker identification, non-overlapped multi-speaker identification, speech extraction for non-overlapped audio, single speaker identification for a single word and non-overlapped speaker verification that include unknown speakers. Figure 3.3 below show the overall methodologies of this project.

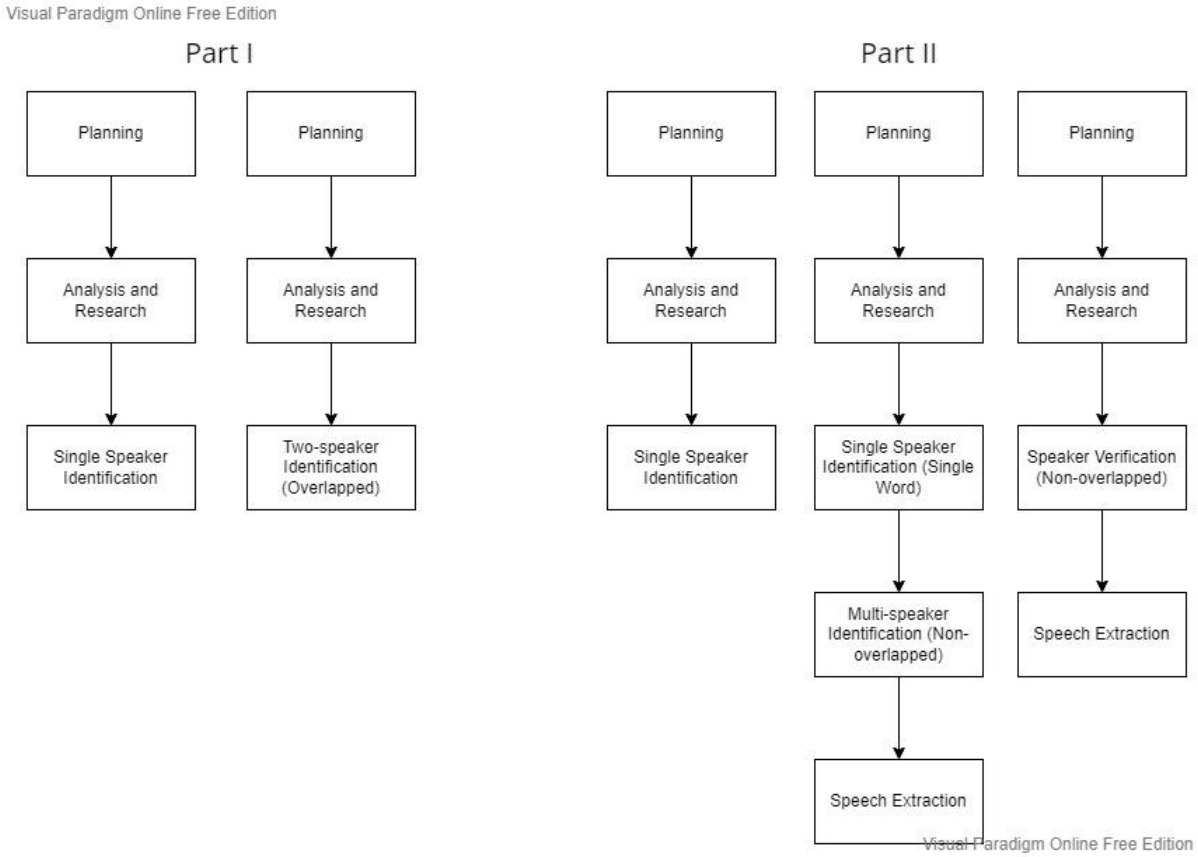


Figure 3.3: Methodologies of this project

3.2.1 Speech Recognition Module

During the final year project I, the speech recognition module is implemented using API and also python library. First of all, by importing PyAudio into the project, microphone input will become available [12]. Next, by importing SpeechRecognition library, the speech recognition engine or API support will be available [12]. There are few engines or APIs such as CMU Sphinx, Google Speech Recognition, Google Cloud Speech API, Wit.ai, Microsoft Bing Voice Recognition, Houndify API, IBM Speech to Text and Snowboy Hotword Detection. Out of all of them, CMU Sphinx and Snowboy Hotword Detection works offline. Google Speech Recognition which is a pretrained python model is used in final year project I.

During final year project II, google cloud speech to text API was used. This is not free. However, due to activating new google cloud platform account, 300-dollar free credits are given for free. These free credits are used to activate the google cloud speech to text API to use in this research project. Google cloud speech to text API performs better than python pretrained google speech recognition model.

3.2.2 Speaker Identification Model

The main model that is used in this project is Gaussian Mixture Model (GMM). GMM is a kind of probabilistic model where all the data points are generated from a mixture of a finite number of Gaussian distributions with unknown parameters [13]. In this project, GMM was implemented using python machine learning library, sklearn.

For all single or two speaker models, the number of mixture components are set to 6. However, for UBM, the number of mixture components is set to 512 to 2048 based on [11]. However, in this project, the number of mixture components is set to 16 for UBM. This is due to the lack of memory and computation power to process. Not only that, the lack of data is also an issue.

The number of expectation-maximization (EM) iterations to perform is set to 200. This is needed because GMM is a kind of unsupervised learning. Therefore, Maximum a Posteriori (MAP) adaptation approach is needed as expectation-maximization iterations will help estimates the parameters in statistical models. This is even more needed in training UBM as the UBM GMM will be adapted using Maximum a Posteriori (MAP) adaptation. The EM algorithm will be started with parameters learnt by UBM [11]. This mean that only mean will be adapted therefore, the GMM can be used for identifying all kind of audios.

The covariance type is set to diagonal type where each component has own diagonal covariance matrix. Diagonal is chosen rather than a full-covariance because it is more efficient and it works better.

The number of initializations is set to 3, where the best result will be kept.

The advantages of GMM are that it is computationally fast with low number of mixture components. Not only that, it is a very strong model for representing speaker. It is only beaten by recent deep learning technique. Deep learning required very strong computational power and large amount of training data.

Table 3.2 below shows the parameters adjusted for sklearn GMM for this project:

n_components	The number of mixture components
max_iter	The number of Expectation-maximization (EM) iterations to perform
covariance_type	A string to describe the type of covariance parameters to use
n_init	The number of initializations to perform

Table 3.2: Main Gaussian Mixture Parameters Adjusted in this Project

3.2.2.1 Single Speaker Identification

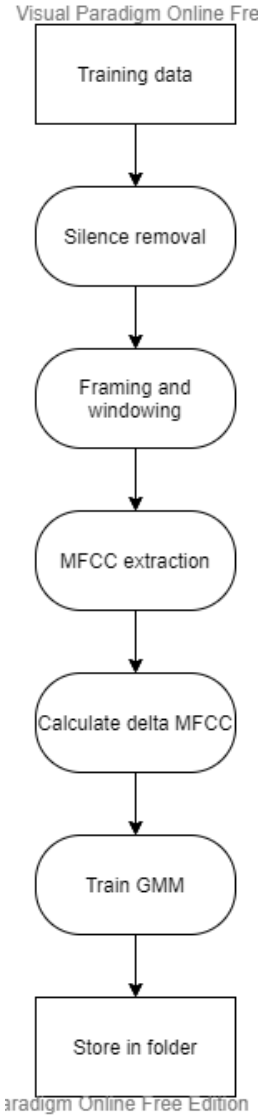


Figure 3.4: Single Speaker Identification

Figure 3.4 show the overall process of single speaker identification. First of all, each user will input their voice into the system with clean non-overlapping speech audio. Then, the silence period will be removed by using webrtcvad, wave and simpleaudio packages [14]. Next, the audio will undergo framing and windowing. Framing is needed as frequency in a signal change over time, doing MFCC extraction across the whole audio signal will not work as the frequency contours of the signal over time will be lost. Therefore, we extract only a short period of time for MFCC extraction. After that, windowing is performed to reduce spectral of voice leakage [15]. The pre-processed audio signal is then extracted using MFCC extraction. The extraction of MFCCs can be done easily with the help of python_speech_features which is an online free open-sources library that provides common speech features for ASR including MFCCs and

filterbank energies. The MFCC extraction algorithm can be used by using the command mfcc() [16]. In this speaker identification, 20 coefficients of the MFCCs will be used. Therefore, a matrix where by 20 coefficients * number of frames will be formed. Next, the delta MFCCs is calculated by finding the difference of each coefficient between each frame. Delta MFCCs is needed because MFCCs describes the instantaneous spectral envelope shape of the speech signal. However, speech signals vary along the time and it is always in a constant flux. Therefore, the sequence of transition between phonemes which is delta MFCCs can describe the acoustical signal more accurately. After that, the MFCCs and also the delta will be used to train a Gaussian Mixture Model and the model will be stored [17].

During testing phase, the Gaussian Mixture Model will be used to calculate similarity score with the test recordings and determine who is the speaker.

It is assumed that all the speakers are known and stored in the databank.

3.2.2.2 Two-Speaker Identification (Overlapped)

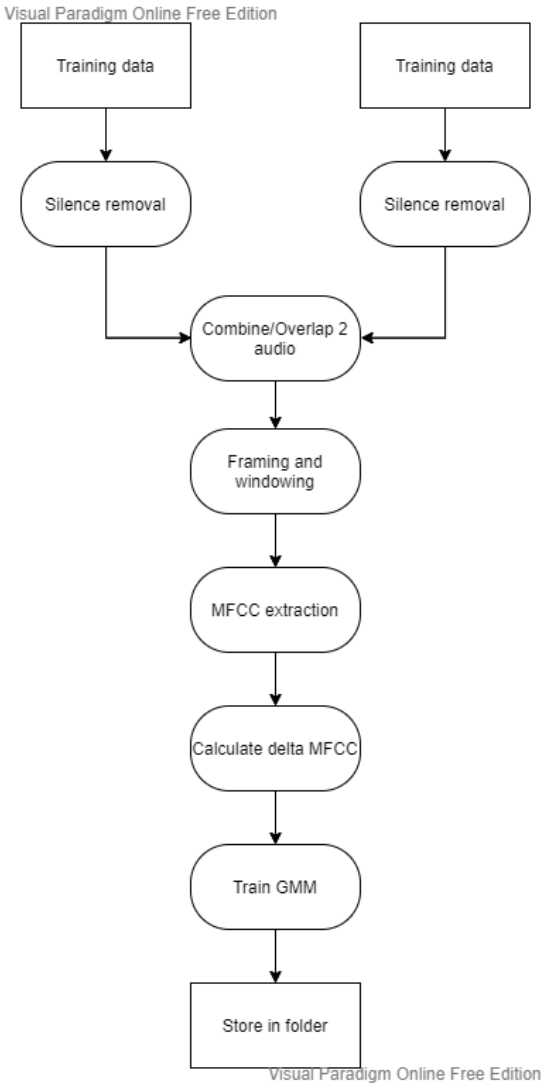


Figure 3.5: Two-Speaker Identification (Overlapped)

According to Figure 3.5, the method is almost the same as single speaker identification, but this time, before the framing, windowing, MFCC extraction and training of model. All the training audios for each user will be combine using combination formula C_2^N, where N = number of speakers registered. The combination process is by overlapping the audio with pydub library. After that, extraction of MFCC, calculation of delta will occur and finally the training of GMM. The number of GMMs stored is equal to $N + C_2^N$, where N = number of individual speakers.

During testing phase, the Gaussian Mixture Model will be used to calculate similarity score with the test recordings and determine who is the two-speaker or single speaker.

All speakers must be known.

3.2.2.3 Single Speaker Identification (Single Word)

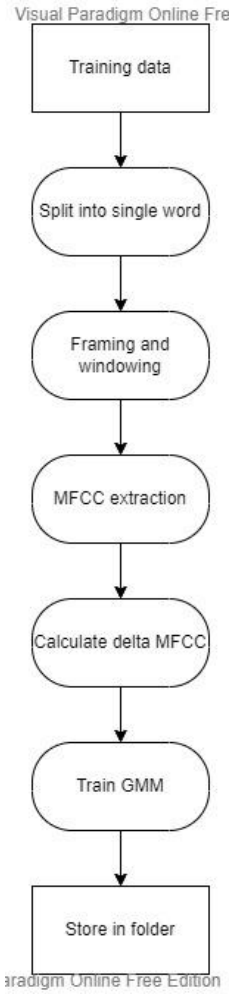


Figure 3.6: Single Speaker Identification (Single Word)

These models are trained by using single word audio of each speaker. It is also trained as GMM. In order to get single speaker models for this single word version, the training audios are split into each word before feeding to the GMM. It undergoes the same process of training single speaker identification model. However, silence removal is not needed as it will be using the training audios that have already been pre-processed. These models will be used to help to identify multi-speaker in the following part by using it to guess how many speakers exist in the audio.

After test audio is split into single word using Google Cloud Speech-To-Text API, each word will use the GMM to calculate and determine the speaker of that word.

All speakers must be known.

3.2.2.4 Multi-Speaker Identification (Non-Overlapped)

This model is aided by google cloud speech to text API. Google cloud speech to text API has speaker diarization function. However, in order for speaker diarization of the google cloud speech to text API to works, number of speakers need to be inputted. Even though setting min speaker to 1 and max speaker to 5, it will always detect as 1 speaker due to it being in BETA version. Therefore, this project proposed a way to help determine how many speakers is in the audio before inputting to help the API by using confident count.

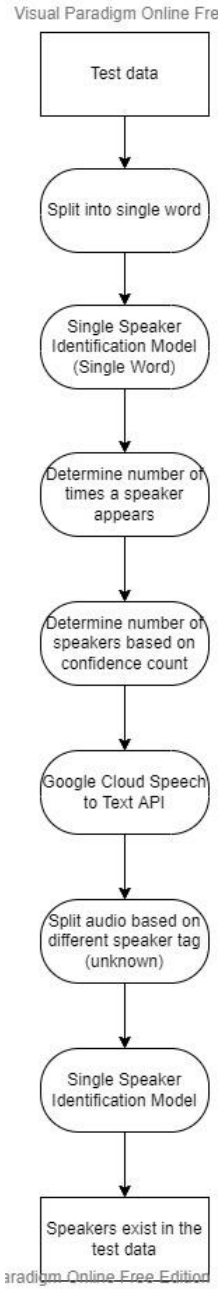


Figure 3.7: Multi-Speaker Identification (Non-Overlapped)

According to Figure 3.7, when a test data that contains multiple speakers where speech is not overlapped is inputted. The test data will be split into each single word. Each word will be tested with the single speaker identification with the single word model. For example, if a word is detected as speaker 19, the count of speaker 19 exist in the data will increase by 1. In order for the system to declare that speaker 19 exist in the test data, the exist count need to cross a confidence value because single speaker identification (single word) model detect wrongly sometimes. After getting the total number of speakers, the number is inputted into speech diarization of google cloud speech to text API. After the result is returned, the speech sequence is labelled with unknown speaker tag 1, 2, 3 etc. However, by splitting the test data based on the sequence and test using single speaker identification model. The known speaker can be determined.

All speakers must be known.

3.2.2.5 Speaker Verification

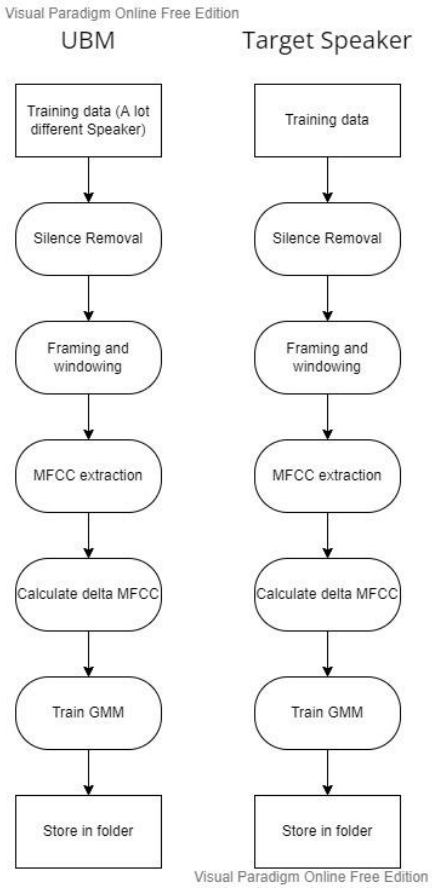


Figure 3.8: Speaker Verification

According to Figure 3.8, two model is trained for speaker verification. One for representing target speaker model and one for representing all others, UBM. The target speaker model is trained in the same way as single speaker identification. However, UBM is trained with training data from a lot of different speakers.

When test data is tested, log-likelihood with both target speaker model and UBM will be calculated.

$$similarity\ score = loglikelihood(target) - loglikelihood(UBM)$$

Threshold can be set manually, but it is usually 0. When similarity score > threshold, it means that the test data is from the target speaker. When similarity score < threshold, it means that the test data is from unknown speaker.

Only target speaker must be known.

3.2.3 Speaker Extraction Module

After model in 3.2.2.4 Multi-Speaker Identification (Non-Overlapped) helps identify the speaker existing in a speech audio without speakers overlapping each other, the speech will be extract to under each target separately with the help of the google cloud speech to text API for speaker diarization and single speaker identification model for speaker recognition.

3.2.4 Noise Estimation Module

This module is for the Voice to Text with Speaker Recognition where only the target speaker speech needs to be extracted. The models that are used to help is the models in 3.2.2.5 Speaker Verification. Target speaker's GMM and UBM-GMM are used.

After google cloud speech to text API help diarize the speech audio into few speakers' tags. Speaker verification will identify whether each speaker tag is the target speaker or the noise (UBM). Target speaker speech will be extracted while speech from unknown voice which is noise speech will be removed.

3.3 General Overview of the Flow between Integrated Models

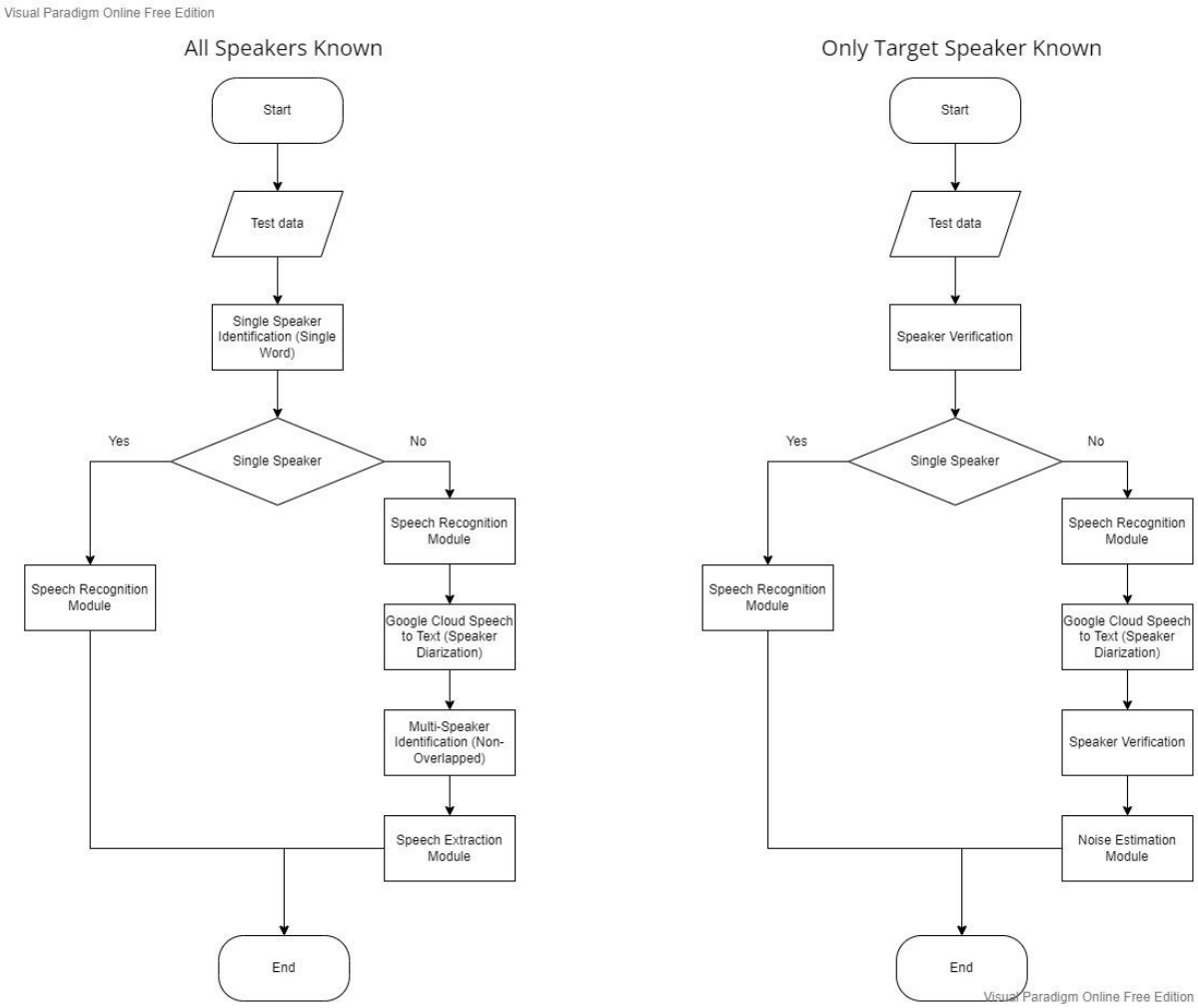


Figure 3.9: General Overview of the Flow between Integrated Models

Figure 3.9 show the general overview of how the integrated model can works if it is being implemented for practical use in the future.

For case where all speakers are known (left), test data with either single or multiple speakers is inputted into the system. Then, by using single word version of single speaker identification, existent of multiple speakers can be confirmed and the number can be guessed. If the test data is only a single speaker, no further action is required and it will be recognized as speech straight. However, if test data consists of multiple speakers, it will be recognized as speech first. Then, speaker diarization is launched before speech extraction and finally every speaker will have their own speech separated under their tag.

For case where only target speaker is known (right), test data with either only the target speaker or target speaker with unknown speakers is inputted into the system. Then, by using speaker verification, it will be classified into single or multiple speakers. If the test data is only the

target speaker, no further action is required and it will be recognized as speech straight. However, if test data consists of unknown voice, it will be recognized into speech first. Then, speaker diarization and speaker verification model will help with the noise estimation module to retain only the target speaker's speech.

3.4 Implementation Issues and Challenges

There are a few expected implementation issues and challenges that will be met during implementation of this voice to text conversion app with speaker recognition. First of all, one of the expected implementation issues is the performance of the system. This is because, the computational power of personal laptop is limited. Therefore, deep learning is hard to be implemented. GMM performs pretty well but for separating overlapped speech, deep learning technique is required. Due to this reason, in this project, the problem solved is for non-overlapped speech using machine learning technique that doesn't touch the territory of data science and deep learning.

Other than that, the limited amount of training data is also one of the implementation issues. Deep learning required large amount of data. Deep learning is hard to be used because of limited amount of training data in this project. Therefore, machine learning technique such as GMM is used in this project because lesser amount of training data is required.

Next, one of the challenges that will be met is the lack of research in this area. There is less research on this speaker recognition area and also lack of open-source codes to aid in this project. Most of the researches done rarely explain the implementation method in details. Therefore, it makes the implementation of the system become a challenge. This research also required high level in mathematics statistical skill in order to understand the wave form, and data generated for wave form. This project was only able to scratch and understand the surface of Digital Signal Processing Technology.

Last but not least, another implementation challenges that will be met is the difficulty of extraction process in speaker extraction. There are a lot of obstacles in order to extract target user voice from an overlapping speech. One of the cases is, it might only work if the louder or dominant voice in the audio is the target. Extraction might not work if the unwanted speech completely covers the voice of the target voice in the overlapping speech.

CHAPTER 4: EXPERIMENT AND EVALUATION

4.1 Experiments with Self-Generated Datasets

```
1.Record audio for training
2.Combine training audio
3.Train Model
4.Record audio for testing
5.Test Model
Input:
```

Figure 4.1: Menu

When the program is started, the menu of the program will appear as Figure 4.1 above. In the experiment, it is assumed that the system works under an environment where all the speakers are known.

4.1.1 Speech Recognition

When option 5 is chosen, the audio recorded from option 4 will be passed into the system to identify the speaker and also be recognized by python pretrained google model after adjusting for the ambient noise using speech recognition library function.

```
================================================================================
ASZ 1.wav
791
20
[-19.69567931 -19.3282809  -22.8983165  -23.42967562 -23.42795245]
    detected as -  Ang Sea Zhe
Ang Sea Zhe: can you saying the teeth come down from the TV with a knife in hand sink
================================================================================
```

```
================================================================================
ASZ 2.wav
812
20
[-21.03014334 -19.79944015 -23.41487302 -23.38348229 -24.23218097]
    detected as -  Ang Sea Zhe
Ang Sea Zhe: virus in modern history perhaps all the time man 1918 Spanish
================================================================================
```

```
==========================================================================================================
ASZ 3.wav
908
20
[-22.79199324 -19.34093877 -25.37142918 -25.21032224 -24.98975894]
    detected as -  Ang Sea Zhe
Ang Sea Zhe: that tall is a non because medical records were not kept in many area the pandemic hit during
==========================================================================================================
==========================================================================================================
ASZ 4.wav
896
20
[-25.50707079 -23.34093851 -27.01846692 -27.01242156 -26.47813364]
    detected as -  Ang Sea Zhe
Ang Sea Zhe: who was Peter to a higher proportion of young adults than most Blue virus the panda mix started
==========================================================================================================
==========================================================================================================
ASZ Loud.wav
869
20
[-26.04904305 -24.00627013 -28.1337011  -27.74365363 -26.4969753 ]
    detected as -  Ang Sea Zhe
Ang Sea Zhe: house near the mountain I have two brothers and one sister and I bought born last my father Turkish
==========================================================================================================
```

Figure 4.2: Speech recognition of speaker - Ang Sea Zhe

Above figure show the recognized speech for the test audio recordings for the speaker Ang Sea Zhe.

In ASZ 1.wav, the actual speech is as below:

> Soon as he finish saying, the thief jump down from the TV with a knife in hand.

The detected speech is as below:

> Can you saying, the teeth come down from the TV with a knife in hand.

In ASZ 2.wav, the actual speech is as below:

> Virus in modern history perhaps all the time “um” 1918 Spanish.

The detected speech is as below:

> Virus in modern history perhaps all the time man 1918 Spanish.

In ASZ 3.wav, the actual speech is as below:

That tall is unknown because medical records were not kept in many area the pandemic hit during.

The detected speech is as below:

That tall is a non because medical records were not kept in many area the pandemic hit during.

In ASZ 4.wav, the actual speech is as below:

That tall is unknown because medical records were not kept in many area the pandemic hit during.

The detected speech is as below:

That tall is a non because medical records were not kept in many area the pandemic hit during.

In ASZ Loud.wav, the actual speech is as below:

House near the mountain I have two brothers and one sister and I was born last my father teach.

The detected speech is as below:

House near the mountain I have two brothers and one sister and I bought born last my father Turkish.

It can be seen that, there is some slight mistake during the speech recognition. The possible reasons that cause these problems are, pronunciation problems. Other than that, it might also be the recordings problems because the recordings for test audio is 10s before silence removal. This means that the recordings might not be a full sentence. In later part of the experiments with LibriSpeech datasets, Google Cloud Speech-To-Text API is used instead of the python speech recognition library in this part.

4.1.2 Single Speaker Identification

For training phase, there are a total of 5 speakers used. Ang Sea Zhe, Ang Kian Beng, Ang See Han, Ang See Chien and Ang See Sin.

Each of the speaker is required to speak for a total of 50s. The training audio recordings for each speaker is separated into 5 wav file.

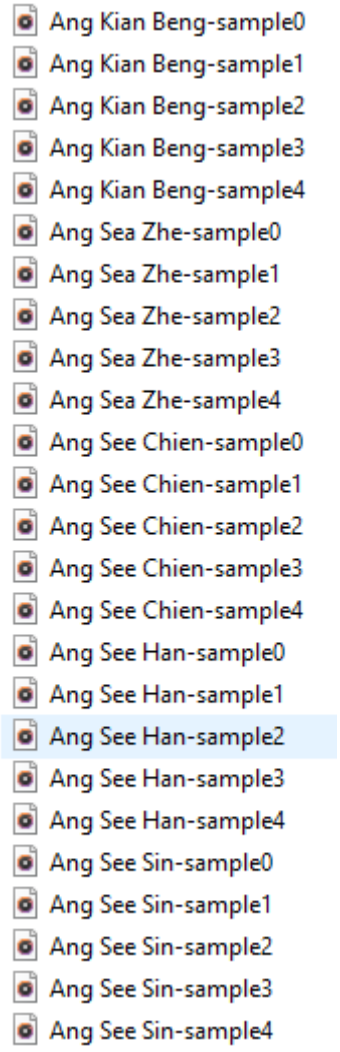


Figure 4.3: The input for the single speaker training phase

However, the total time for each speaker in the end is not equal because right at the time the audio is being recorded, it passes through a silence removal code created using webrtcvad library with an aggressiveness on 0. The aggressive can be ranged from integer between 0-3. The best aggressiveness that works for this project is 0 because as the aggressiveness is raised, there are some speech audios being removed.

After getting the input audio for training phase as Figure 4.3, the model is being trained by selecting option 3. Each single audio file for example Ang Sea Zhe-sample0.wav is being read into the system. Then, MFCC will be extracted and delta will be calculated. The same goes for AngSeaZhe-sample1.wav. However, the extracted features will be vertically stacked to the previous features. This continues until sample4.wav, then a GMM will be trained using the stacked features. The setting of the GMM is as below:

n_components	6
max_iter	200
covariance_type	diag
n_init	3

Table 4.1: Gaussian Mixture Parameters

After that, the GMM is stored as .gmm files as in Figure 4.4.

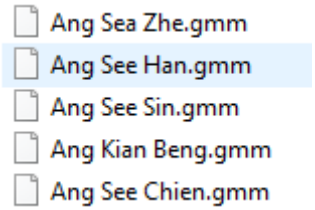


Figure 4.4: The output of the trained model for single speaker identification

The testing audio is recorded using option 4. 25 prerecorded test audio is used to evaluate the model, 5 of them belongs to each speaker. The test recordings undergo silence removal, MFCC extraction and delta calculation also. During testing, it is classified by comparing log likelihood between all the models. The one that has the highest score will be the detected speaker.

5 Test Audios from Each Speaker:

Normal Tone	4
Louder Tone	1

Table 4.2: Test audio types

```
==========================================================================================================================
ASH 1.wav
908
20
[-27.55478431 -28.99130911 -24.20090125 -22.96274308 -26.13390114]
    detected as -  Ang See Han
Ang See Han: in modern City perhaps all the time was the 1918 Spanish do it for about 20 to 50 million people worldwide
==========================================================================================================================
```

Figure 4.5: Example of correctly identified test audio for single speaker identification

Figure 4.5 show that a test recording from Ang See Han is correctly identified as Ang See Han. 24 recordings out of 25 recordings are classified correctly. The misclassified recording is the loud tone type from Ang See Chien. It is misclassified as Ang See Han voice.

It has achieved an accuracy of 96.00%.

4.1.3 Two-Speaker Identification (Overlapped)

For training phase, option 3 from menu is selected. Then it will combine that 5 speakers' training audio. It is combined through combinations function and overlay function. The shorter wav file after silence removal will overlay the longer wav file so that the training data is overlapped most of the time.

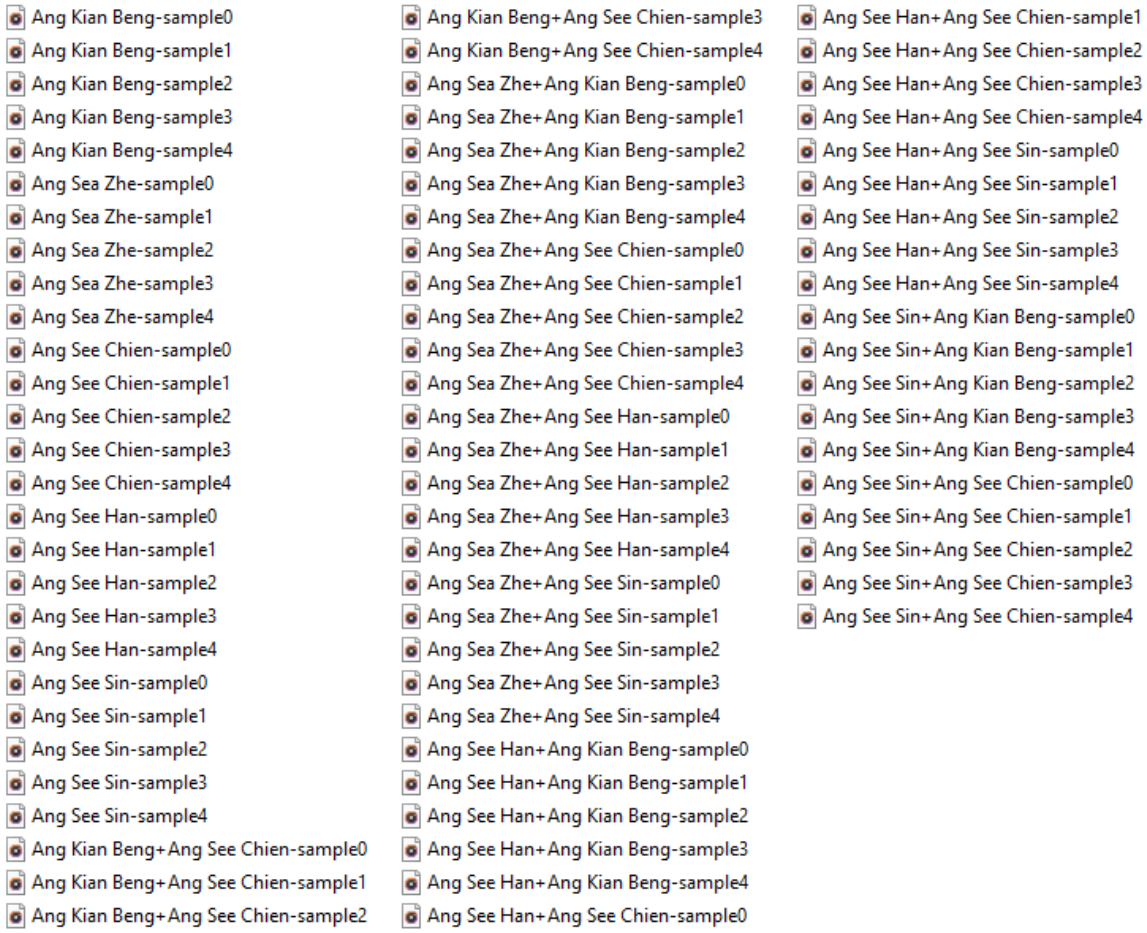


Figure 4.6: The input for two speaker training phase

Figure 4.6 above show all the training wav file after the combination function. After that, the model is trained in the same way as in the single speaker identification. The GMM settings are as below.

n_components	6
max_iter	200
covariance_type	diag
n_init	3

Table 4.3: Gaussian Mixture Parameters

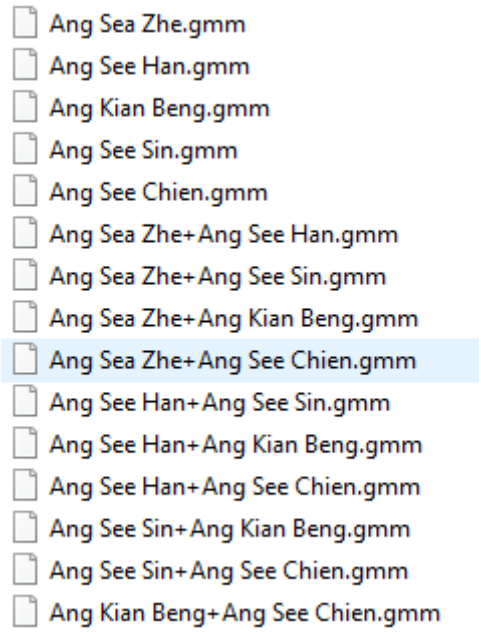


Figure 4.7: The output of trained model for two speaker identification

Figure 4.7 show the output of the trained GMM. Each test audio will be compared to all the GMM in Figure 4.7 and classify it accordingly similar to single speaker identification.

For testing phase, the same 25 recordings in single speaker identification are used. However, in order to test out the pair speaker identification, another 40 test recordings are formed artificially using combinations and overlay function. According to Table 4.2, only the normal tone is use for the combination. Therefore, the total number of outputs of test recordings are $C_2^N \times 4 = 40$ where N = 5 speakers. A total of 65 test recordings are used for testing phase.

```
==========================================================================================
ASS 2+AKB 2.wav
845
20
[-26.1143661  -28.21733007 -26.62241288 -26.8171046  -26.14697116
 -26.38394328 -28.30653375 -29.38042941 -25.87390133 -27.2828306
 -27.1906148  -29.04808624 -25.86745892 -26.87322628 -29.03010311]
    detected as -  Ang See Sin+Ang Kian Beng
Ang See Sin+Ang Kian Beng: in the spring of 1918 was followed by a much more severe
==========================================================================================
```

Figure 4.8: Example of correctly identified test audio for two speaker identification

There are a lot of misclassified audios in this experiment and it only have an accuracy of 72.31%.

4.1.4 Discussion and Concluding Remark

In this part of the experiment, single speaker identification and two-speaker identification (overlapped) was tested with self-generate datasets.

For single speaker identification, it can be seen that GMM performs quite good with an accuracy of 96.00%. This prove that GMM can be used to represents speakers' model as it can even identifies the speaker even when the speaker raises his/her voice. However, the training data and testing data in this experiment is quite limited as it was gathered from family members.

For two-speaker identification (overlapped), it performs worse with only 72.31% accuracy on the test data. It is very skewed to the training data and does not have much practical use. Not only that, this part of the experiment met a great challenge as even after two-speaker is identified in the overlapped speech. It is very hard to split them without deep learning that required a lot of training data and computational power.

4.2 Experiments with LibriSpeech Datasets

4.2.1 Single Speaker Identification

For training phase, there are a total of 20 speakers used. 19 speakers are from the librispeech dataset while a speaker is Ang Sea Zhe. Each speakers have 30 utterances where each utterance is around 10 seconds. Every file is normalized to wav file with sampling rate of 16000Hz.

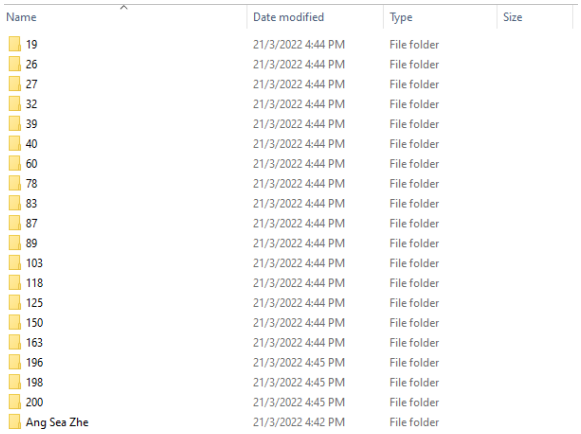


Figure 4.9: Training Data of Single Speaker Identification

By training GMM for each speaker, the following GMMs will be generated. The GMM setting are as Table 4.4 below.

n_components	6
max_iter	200
covariance_type	diag
n_init	3

Table 4.4: Gaussian Mixture Parameters

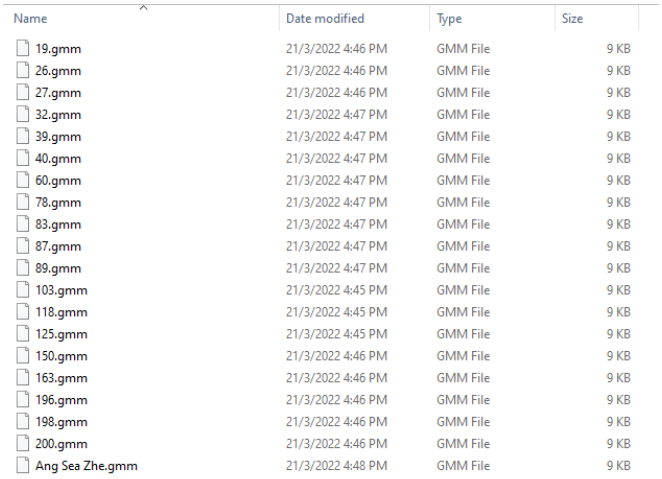


Figure 4.10: The Output of the GMMs for Single Speaker Identification

In this experiment, the classification accuracy is the area of concern. The evaluation tests that are done on the single speaker identification is 100 train data and 200 test data.

For test on the train data, 5 utterances are picked for each speaker. Therefore, 100 train data is chosen to be tested. In this experiment, the evaluation is done on the classification of speaker tag to evaluate the correctness of the speaker identification model.

For test on the test data, 10 utterances are picked for each speaker. Therefore, 200 test data is chosen to be tested.

4.2.1.1 Evaluation on Classification of Speaker Tag

Actual \ Predicted	102	118	125	150	163	19	196	198	200	26	27	32	39	40	60	78	83	87	89	ASZ
102	5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
118	0	5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
125	0	0	5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
150	0	0	0	5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
163	0	0	0	0	5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
19	0	0	0	0	0	5	0	0	0	0	0	0	0	0	0	0	0	0	0	0
196	0	0	0	0	0	0	5	0	0	0	0	0	0	0	0	0	0	0	0	0
198	0	0	0	0	0	0	0	5	0	0	0	0	0	0	0	0	0	0	0	0
200	0	0	0	0	0	0	0	0	5	0	0	0	0	0	0	0	0	0	0	0
26	0	0	0	0	0	0	0	0	0	5	0	0	0	0	0	0	0	0	0	0
27	0	0	0	0	0	0	0	0	0	0	5	0	0	0	0	0	0	0	0	0
32	0	0	0	0	0	0	0	0	0	0	0	5	0	0	0	0	0	0	0	0
39	0	0	0	0	0	0	0	0	0	0	0	0	5	0	0	0	0	0	0	0
40	0	0	0	0	0	0	0	0	0	0	0	0	0	5	0	0	0	0	0	0
60	0	0	0	0	0	0	0	0	0	0	0	0	0	0	5	0	0	0	0	0
78	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	5	0	0	0	0
83	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	5	0	0	0
87	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	5	0	0
89	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	5	0
ASZ	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	5

Table 4.5: The Confusion Matrix on Train Data for Single Speaker Identification

According to Table 4.5, it can be seen that it performs quite well on the train data and the classification is correct for all the train data tested for every speaker.

	Precision	Recall	F1-score	Support
102	1.00	1.00	1.00	5
118	1.00	1.00	1.00	5
125	1.00	1.00	1.00	5
150	1.00	1.00	1.00	5
163	1.00	1.00	1.00	5
19	1.00	1.00	1.00	5
196	1.00	1.00	1.00	5
198	1.00	1.00	1.00	5
200	1.00	1.00	1.00	5
26	1.00	1.00	1.00	5
27	1.00	1.00	1.00	5
32	1.00	1.00	1.00	5
39	1.00	1.00	1.00	5
40	1.00	1.00	1.00	5
60	1.00	1.00	1.00	5
78	1.00	1.00	1.00	5
83	1.00	1.00	1.00	5
87	1.00	1.00	1.00	5
89	1.00	1.00	1.00	5
ASZ	1.00	1.00	1.00	5
Accuracy			1.00	100
Macro Average	1.00	1.00	1.00	100
Weighted Average	1.00	1.00	1.00	100

Table 4.6: The Classification Report on Train Data for Single Speaker Identification

It can be seen that it performs very well on the train data. There is no misclassification for any speaker that is enrolled into the system. It has an accuracy of 100.00%.

Actual \ Predicted	102	118	125	150	163	19	196	198	200	26	27	32	39	40	60	78	83	87	89	ASZ
102	10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
118	0	10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
125	0	0	10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
150	0	0	0	10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
163	0	0	0	0	10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
19	0	0	0	0	0	10	0	0	0	0	0	0	0	0	0	0	0	0	0	0
196	0	0	0	0	0	0	10	0	0	0	0	0	0	0	0	0	0	0	0	0
198	0	0	0	0	0	0	0	10	0	0	0	0	0	0	0	0	0	0	0	0
200	0	0	0	0	0	0	0	0	10	0	0	0	0	0	0	0	0	0	0	0
26	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0	0	0	0	0	0
27	0	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0	0	0	0	0
32	0	0	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0	0	0	0
39	0	0	0	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0	0	0
40	0	0	0	0	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0	0
60	0	0	0	0	0	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
78	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	10	0	0	0	0
83	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	10	0	0	0
87	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	10	0	0
89	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	10	0
ASZ	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	10

Table 4.7: The Confusion Matrix on Test Data for Single Speaker Identification

The confusion matrix shows that every classification of speaker has given a correct result. Every speaker has a true positive of 10 out of 10 utterances.

	Precision	Recall	F1-score	Support
102	1.00	1.00	1.00	10
118	1.00	1.00	1.00	10
125	1.00	1.00	1.00	10
150	1.00	1.00	1.00	10
163	1.00	1.00	1.00	10
19	1.00	1.00	1.00	10
196	1.00	1.00	1.00	10
198	1.00	1.00	1.00	10
200	1.00	1.00	1.00	10
26	1.00	1.00	1.00	10
27	1.00	1.00	1.00	10
32	1.00	1.00	1.00	10
39	1.00	1.00	1.00	10
40	1.00	1.00	1.00	10
60	1.00	1.00	1.00	10
78	1.00	1.00	1.00	10
83	1.00	1.00	1.00	10
87	1.00	1.00	1.00	10
89	1.00	1.00	1.00	10
ASZ	1.00	1.00	1.00	10
Accuracy			1.00	200
Macro Average	1.00	1.00	1.00	200
Weighted Average	1.00	1.00	1.00	200

Table 4.8: The Classification Report on Test Data for Single Speaker Identification

The classification report on test data show that indeed GMM is good at representing speaker model in single speaker identification. It has a result of 100.00% accuracy even on the test data.

4.2.2 Multi-Speaker Identification (Non-Overlapped) with Training Audio Longer Than or Equal to 0.5s

In this experiment, 10 speakers from LibriSpeech dataset are used. In order to determine the number of speakers in a non-overlapped audio, extra single speaker models are built using a lot of single word utterances. By using google cloud speech to text API, all the training data of the selected 10 speakers are split into single word. In this experiment, only audio with longer than or equal to 0.5 seconds are take into account. Even in the test data, only words longer than 0.5 seconds are tested. This is tested to determine whether does the length of each training audio chosen affect the result due to feature extraction from different size of audio.

Below is example of single word audio of speaker 19.

Figure 4.11: Example of Single Word Audio of Speaker 19

They are used to build GMM using the following settings.

n_components	6
max_iter	200
covariance_type	diag
n_init	3

Table 4.9: Gaussian Mixture Parameters

There are two kinds of model used in this experiment. One of them is the models mentioned above, another one is the models built in 4.2.1 Single Speaker Identification. After the model is done building, the next part is on making the test data.

Only single speaker datasets can be found on the internet. Therefore, artificial test data has to be made from the available dataset, which is librispeech in this experiment. In this experiment, 100 test data are created. 25 for 1-speaker, 25 for 2-speaker, 25 for 3-speaker and 25 for 4-speaker. The number of speakers is ranging from 1 to 4 speaker and all the speakers are known in this experiment.

Figure 4.12: Way of Combining to make Test Data

According to Figure 4.12, the audios are combined in this way. Therefore, there is no overlapping in speech audio. During the combination of data, a python list will be generated for diarization evaluation reference later. Table below show how the list looks like.

[(	"19",	0.0,	11.4	),
(	"26",	11.4,	25.6	)]

Table 4.10: Diarization Evaluation Reference

The list above means that, speaker 19 speaks from 0.0 seconds to 11.4 seconds and then speaker 26 speaks from 11.4 seconds to 25.6 seconds. After generating the list, the list is saved to binary stream data using pickle library to be used as reference later.

The evaluation of this experiments can be separate into 2 major parts. The first part is on evaluating the classification of number of speakers. The second part is on evaluating the Diarization Error Rate (DER). DER evaluation is only done on 2-speaker, 3-speaker and 4-speaker data. DER is an error rate used to evaluate diarization technique. In this project, a lightweight library called simpleDER is used to evaluate DER. The strict formula of DER is as below

$$DER = \frac{False\ Alarm + Miss + Overlap + Confusion}{Reference\ Length}$$

Reference length is the total length of the reference data (the actual data). False Alarm is length of the segment that is considered as speech in the predicted but not in the reference data. Miss is the length of segments that are considered speech in reference data but not the predicted. Overlap is the length of segments considered as overlapped speech in predicted but not in reference data. Overlap is not available in this library, but the testing in this experiment also doesn't include overlap. Confusion is the length of segments which are assigned to different speaker in predicted and reference data. Confusion is the major concern in this experiment.

4.2.2.1 Evaluation on Classification of Number of Speakers

		Predicted			
		1-speaker	2-speaker	3-speaker	4-speaker
Actual	1-speaker	23	2	0	0
	2-speaker	1	24	0	0
	3-speaker	1	6	18	0
	4-speaker	0	5	12	7

Table 4.11: The Confusion Matrix on Test Data for Multi Speaker Identification 1

From the confusion matrix in Table 4.11, it can be observed that for 1-speaker test data, 23 out of 25 are classified correctly and 2 out of 25 are classified wrongly as 2-speaker.

For 2-speaker test data, 24 out of 25 are classified correctly and 1 out of 25 are classified wrongly as 1-speaker.

For 3-speaker test data, 18 out of 25 are classified correctly, 6 out of 25 are classified wrongly as 2-speaker, 1 out of 25 are classified wrongly as 1-speaker.

For 4-speaker test data, 7 out of 25 are classified correctly, 12 out of 25 are classified wrongly as 3-speaker, 5 out of 25 are classified wrongly as 2-speaker.

	Precision	Recall	F1-score	Support
1-speaker	0.92	0.92	0.92	25
2-speaker	0.65	0.96	0.77	25
3-speaker	0.60	0.72	0.65	25
4-speaker	1.00	0.28	0.44	25
Accuracy			0.72	100
Macro Average	0.79	0.72	0.70	100
Weighted Average	0.79	0.72	0.70	100

Table 4.12: The Classification Report on Test Data for Multi Speaker Identification 1

Based on the result above, it can be deduced that, as the number of speakers increase, the model will make more mistake. 4-speaker has high precision score because there are no false positive, none of other class data is classified as 4-speaker. The model in this experiment has an average accuracy around 72.0%.

It can be concluded that, the model in this experiment can work up to 3 speakers with optimal result.

4.2.2.2 Evaluation on Diarization Error Rate

The average of DER across all test data including 2-speaker, 3-speaker and 4-speaker is 0.100. This mean that the diarization model has scored 90% with 10% DER. However, this is the based on the result of all class. Below table show the DER of all class separately.

	DER
2-speaker	0.038
3-speaker	0.110
4-speaker	0.153

Table 4.13: DER of each class for Multi Speaker Identification 1

According to Table 4.13, it can be seen that the diarization perform very well on 2-speaker with only 3.8% of error. It performs slightly worse on 3-speaker with 11% error and 15.3% error on 4-speaker. The DER is usually 10% to 20%, therefore, the model in this experiment can be said to perform quite okay.

4.2.3 Multi-Speaker Identification (Non-Overlapped) with Training Audio of All Lengths

In this experiment, 10 speakers from librispeech dataset are used. Almost all the settings are the same as the previous experiment 4.2.2 Multi-Speaker Identification (Non-Overlapped) with Training Audio Longer Than or Equal to 0.5s. However, this time all single word audio file is used for training and all the word in test data is tested.

The evaluation plan is the same as the previous experiment. First being evaluation on classification of number of speakers and second being evaluation on DER.

4.2.3.1 Evaluation on Classification of Number of Speakers

		Predicted			
		1-speaker	2-speaker	3-speaker	4-speaker
Actual	1-speaker	21	4	0	0
	2-speaker	1	22	2	0
	3-speaker	0	1	23	1
	4-speaker	0	1	7	17

Table 4.14: The Confusion Matrix on Test Data for Multi Speaker Identification 2

From the confusion matrix in Table 4.14, it can be observed that for 1-speaker test data, 21 out of 25 are classified correctly and 4 out of 25 are classified wrongly as 2-speaker.

For 2-speaker test data, 22 out of 25 are classified correctly, 1 out of 25 are classified wrongly as 1-speaker and 2 out of 25 are classified wrongly as 3-speaker.

For 3-speaker test data, 23 out of 25 are classified correctly, 1 out of 25 are classified wrongly as 2-speaker, 1 out of 25 are classified wrongly as 4-speaker.

For 4-speaker test data, 17 out of 25 are classified correctly, 1 out of 25 are classified wrongly as 2-speaker, 7 out of 25 are classified wrongly as 3-speaker.

	Precision	Recall	F1-score	Support
1-speaker	0.95	0.84	0.89	25
2-speaker	0.79	0.88	0.83	25
3-speaker	0.72	0.92	0.81	25
4-speaker	0.94	0.68	0.79	25
Accuracy			0.83	100
Macro Average	0.85	0.83	0.83	100
Weighted Average	0.85	0.83	0.83	100

Table 4.15: The Classification Report on Test Data for Multi Speaker Identification 2

Based on the result above, it can be deduced that, as the number of speakers increase, the model will make more mistake which was the same as the experiment before. 4-speaker has high precision score because there are no false positive, none of other class data is classified as 4-speaker. It can be seen that the recall score of 4-speaker is very low, because there is less true positive compare to all the positive data for 4-speaker. The model in this experiment has an average accuracy around 83.0%.

It can be concluded that, the model in this experiment can work up to 4 speakers with optimal result.

Comparing to experiment in 4.2.2 Multi-Speaker Identification (Non-Overlapped) with Training Audio Longer Than or Equal to 0.5s, it is proven that the hypothesis of using longer word only for identification is wrong as the model using every single word is better than the model using longer word only.

4.2.3.2 Evaluation on Diarization Error Rate

The average of DER across all test data including 2-speaker, 3-speaker and 4-speaker is 0.077. This mean that the diarization model has scored 92.7% with 7.7% DER. However, this is the based on the result of all class. Below table show the DER of all class separately.

	DER
2-speaker	0.082
3-speaker	0.049
4-speaker	0.099

Table 4.16: DER of each class for Multi Speaker Identification 2

According to Table 4.16, it can be seen that the diarization perform very well on 3-speaker with only 4.9% of error. It performs slightly worse on 2-speaker with 8.2% error and 9.9% error on 4-speaker. The DER is usually 10% to 20%, therefore, the model in this experiment can be said to perform quite okay and better than model in 4.2.2 Multi-Speaker Identification (Non-Overlapped) with Training Audio Longer Than or Equal to 0.5s for overall result.

4.2.4 Speaker Verification

In this experiment, target speaker is set as speaker 19 from librispeech to simulate the experiment.

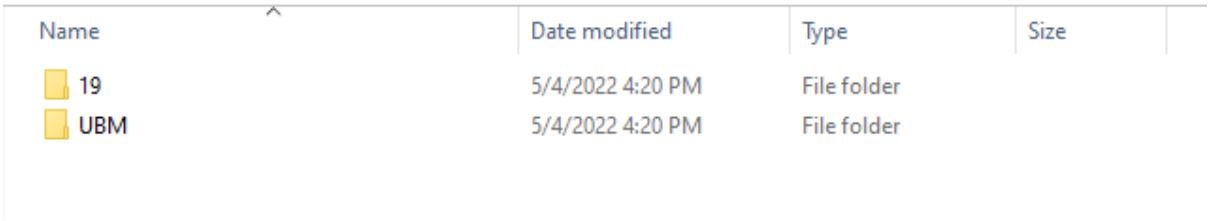


Figure 4.13: Training Data for Speaker Verification

According to Figure 4.13, these 2 sets will be used to train 2 GMMs. Folder 19 contains 30 utterances of speaker 19 while UBM which will represent the unknown consists of 30 utterances each from 50 speakers including speaker 19 (This is to prove that it is not only based on single speaker identification method in 4.2.1 Single Speaker Identification).

The GMM settings for speaker 19 are as below.

n_components	6
max_iter	200
covariance_type	diag
n_init	3

Table 4.17: Gaussian Mixture Parameters

However, for UBM, the GMM settings is in higher order with a greater number of mixture components.

n_components	16
max_iter	200
covariance_type	diag
n_init	3

Table 4.18: Gaussian Mixture Parameters

For the testing set, there are a total of 162 utterances to be tested. 81 utterances from speaker 19 that are not from training set and 81 utterances from unknown speakers from librispeech. Unknown speakers in this experiment are not in the training set of UBM. Meaning they are completely unknown to the system.

For the evaluation plan, the correction of the speaker verification will be tested with threshold = 0. Other than that, Detection Error Tradeoff will be evaluated too. It is a kind of graph to plot error rates of binary classification systems, by plotting false negative rate against false positive rate. This can be obtained by adjusting the threshold to accept or reject the similarity score mentioned in 3.2.2.5 Speaker Verification.

4.2.4.1 Evaluation on Verification of Speaker with Threshold of 0

		Predicted	
		19	Unknown
Actual	19	77	4
	Unknown	0	81

Table 4.19: The Confusion Matrix on Test Data for Speaker Verification

For the testing data, it can be observed that only 4 utterances from speaker 19 is misclassified as unknown. It can be said that this model performs in speaker verification pretty well.

	Precision	Recall	F1-score	Support
19	1.00	0.95	0.97	81
Unknown	0.95	1.00	0.98	81
Accuracy			0.98	162
Macro Average	0.98	0.98	0.98	162
Weighted Average	0.98	0.98	0.98	162

Table 4.20: The Classification Report on Test Data for Speaker Verification

From the classification report in Table 4.20, it can be observed that the model performs well on speaker 19 with overall accuracy of 0.98 which is 98%.

4.2.4.2 Evaluation on Detection Error Tradeoff

By adjusting the acceptance threshold using numpy linspace, the following detection error tradeoff graph is generated.

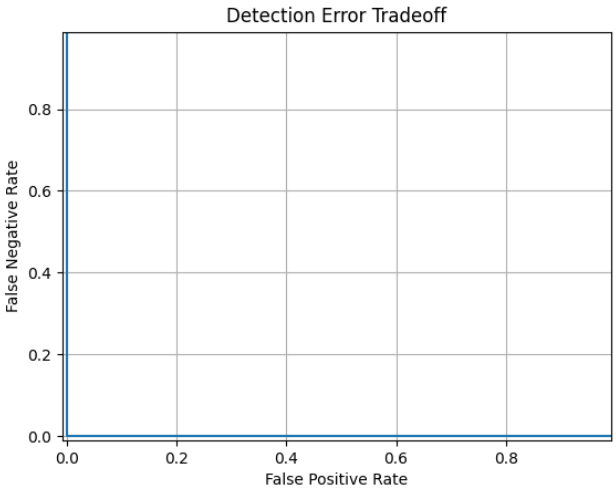


Figure 4.14: The Detection Error Tradeoff on Test Data for Speaker Verification

On the above graph, no curve is observed, it is a sharp L shape, this means that perfect no tradeoff result can be obtained on certain threshold. Based on the experiment, the perfect threshold that will give perfect result is around -1.475 ~ -0.667 for speaker 19. However, this is actually not realistic. The possible reason that causes this might be, the training size for the target speaker is too large, or the testing set is not large enough with more variety. However, the lack of data makes it impossible to increase the testing set.

4.2.5 Voice to Text Conversion App with Speaker Recognition

In this experiment, target speaker chosen is speaker 19 from LibriSpeech datasets. In this experiment, the aim is to extract only the target speaker speech from an audio with target speaker and unknown speaker. Figure below show how the test data is combined.

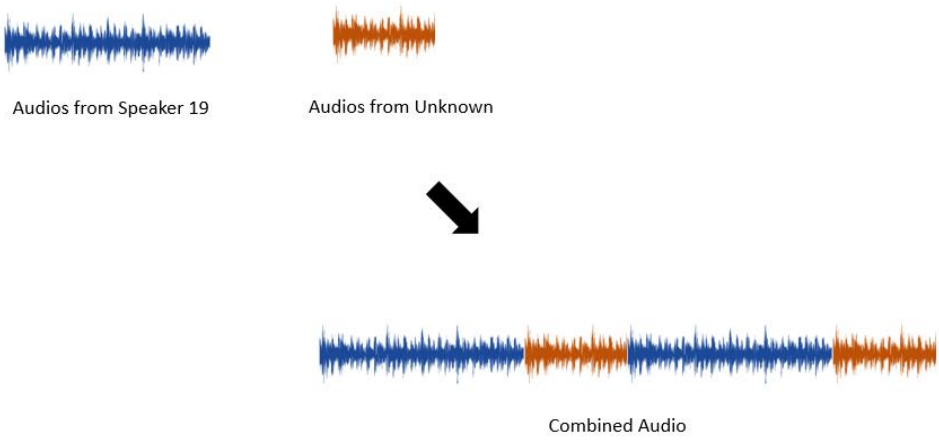


Figure 4.15: Way of Combining to make Test Data for Voice to Text Conversion with Speaker Recognition Experiment

After the combined audio pass through speaker diarization of google cloud speech to text API, the speech can be cut into sequence with different speaker tag, however, the speaker tag is unknown. The speech sequence will then pass into the speaker verification model used in the previous experiment - 4.2.4 Speaker Verification in order to extract only the target speaker speech.

The evaluation plan on this experiment is to look at the word error after the extraction of target speaker speech.

4.2.5.1 Evaluation on Word Error

In the evaluation, I will show 5 results. Each result will have 3 tables, first table showing the actual speech based on the dataset reference, second table showing the whole speech translated by google cloud speech to text API without diarization and last table showing the result after extraction of target speech. Green in colour is the actual target speech transcript while yellow colour is the actual unknown speech transcript

Test Data 1:

19	Chapter thirty Catherine's disposition was not naturally sedentary nor had her habits been ever very industrious but whatever might hitherto have been her defects of that sort
Unknown	And then spreading my cloak I lay on the ground and sank into sleep it was morning when I awoke and my first care was to visit the fire
19	Her mother could not but perceive them now to be greatly increased she could neither sit still nor employ herself for ten minutes together walking round the garden and orchard again and again as if nothing but motion was voluntary
Unknown	I observed this also

Table 4.21: Actual Speech Test Data 1

Mixed Transcript	Chapter 30. Catherines disposition was not naturally sedentary nor had her habits been ever very industrious. But whatever might Heather to have been her defects of that sort. And then spreading my cloak, I lay on the ground and sank and asleep. It was morning when I awoke and my first care was to visit the fire. Her mother could not but perceive them now to be greatly increased. She could neither sit still nor employ herself for 10 minutes together walking around the garden and Orchard again and again as if nothing but motion was voluntary service. Also.

Table 4.22: Mixed Transcript Test Data 1

From Table 4.22, it can be seen that the google cloud speech to text API works quite well except for the last part “I observed this also” become “service. Also”.

19	-
Unknown	Chapter 30. Catherines disposition was not naturally sedentary nor had her habits been ever very industrious. But whatever might Heather to have been her defects of that sort. And then spreading my cloak, I lay on the ground and sank and asleep. It was morning when I awoke and my first care was to visit the fire. Her mother could not but perceive them now to be greatly increased. She could neither sit still nor employ herself for 10 minutes together walking around the garden and Orchard again and again as if nothing but motion was voluntary service. Also.

Table 4.23: Result of VTTSR Test Data 1

From Table 4.23, it can be seen that VTTSR makes a terrible error here and classifying all speech sequence by unknown.

Test Data 2:

19	And it seemed as if she could even walk about the house rather than remain fixed for any time in the parlour her loss of spirits was a yet greater alteration in her rambling and her idleness
Unknown	Teaching said the master ten the master said
19	But when a third night's rest had neither restored her cheerfulness improved her in useful activity nor given her a greater inclination for needlework she could no longer refrain from the gentle reproof of
Unknown	Much could be done in three years

Table 4.24: Actual Speech Test Data 2

Mixed Transcript	And it seemed as if she could even walk about the house rather than remain fixed for any time in the Parlor. Her loss of spirits was a yet greater alteration in her rambling and her idleness teaching master. 10. The master said, but when a third nights arrest had neither restored. Her cheerfulness improved her and useful activity, nor given her a greater inclination for needlework. She could no longer refrain from the gentle reproof of

Table 4.25: Mixed Transcript Test Data 2

It can be noticed that, google cloud speech to text API made some error around middle part and end part.

19	And it seemed as if she could even walk about the house rather than remain fixed for any time in the Parlor. Her loss of spirits was a yet greater alteration in her rambling and her idleness but when a third nights arrest had neither restored. Her cheerfulness improved her and useful activity, nor given her a greater inclination for needlework. She could no longer refrain from the gentle reproof of
Unknown	teaching master. 10. The master said,

Table 4.26: Result of VTTSR Test Data 2

From Table 4.26, it can be observed that VTTSR performed perfectly by filtering out all the unknown speech from target speaker. Although, there are loss in speech in unknown speaker, it is not that much of a concern.

Test Data 3:

19	My dear Catherine I am afraid you are growing quite a fine lady I do not know when poor Richard's cravats would be done if he had no friend but you your head runs too much upon bath
Unknown	But this was all knavery and collusion
19	Catherine took up her work directly saying in a dejected voice that her head did not run upon bath much then you are fretting about general Tilney and that is very simple of you
Unknown	Yet there was fourteen of the spotted fever as well as fourteen of the plague

Table 4.27: Actual Speech Test Data 3

Mixed Transcript	My dear Katherine, I am afraid. You are growing quite a fine lady. I do not know. When Poor Richard's cravats, would be done if you had no friends, but you Your head runs too much upon bath, every in collusion with her. Head did not run upon bath much. Then you are fretting about General tilney. And that is very simple. Love you as well as 14 of the plague.

Table 4.28: Mixed Transcript Test Data 3

19	My dear Katherine, I am afraid. You are growing quite a fine lady. I do not know. When Poor Richard's cravats, would be done if you had no friends, but you Your head runs too much upon bath, with her. Head did not run upon bath much. Then you are fretting about General tilney. And that is very simple. Love you
Unknown	every in collusion as well as 14 of the plague.

Table 4.29: Result of VTTSR Test Data 3

From Table 4.29, it can be observed that VTTSR did classify speech sequence correctly, but the transcription is not that correct since the beginning cause the word error rate to be high.

Test Data 4:

19	For ten to one whether you ever see him again you should never fret about trifles after a short silence I hope my Catherine you are not getting out of humour with home because it is not so grand as Northanger
Unknown	Your presence in the private staircase was the last straw you will forgive us Carlos

19	That would be turning your visit into an evil indeed wherever you are you should always be contented but especially at home because there you must spend the most of your time I did not quite like at breakfast
Unknown	He smoked with a vast contentment that's better nothing to forgive bobby let us call it a misunderstanding graham moved closer

Table 4.30: Actual Speech Test Data 4

Mixed Transcript	41021 why. Don't you ever see him again? You should never fret about Trifles. After a short silence, I hope my Kathryn. You are not getting out of humour with home because it is not so Grand, as Northanger. You won't forgive us. Carlos, that would be wherever you are. You should always be contented, but especially at home because there you must spend most of your time. I did not quite like at breakfast. That's better. Nothing to forgive Bobby. Let us call it, a misunderstanding and move closer.

Table 4.31: Mixed Transcript Test Data 4

19	41021 why. Don't you ever see him again? You should never fret about Trifles. After a short silence, I hope my Kathryn. You are not getting out of humour with home because it is not so Grand, as Northanger. that would be wherever you are. You should always be contented, but especially at home because there you must spend most of your time. I did not quite like at breakfast.
Unknown	You won't forgive us. Carlos, That's better. Nothing to forgive Bobby. Let us call it, a misunderstanding and move closer.

Table 4.32: Result of VTTSR Test Data 4

The VTTSR remove unknown speech pretty well, but the speech at the unknown part has quite high word error rate.

Test Data 5:

19	To hear you talk so much about the French bread at Northanger I am sure I do not care about the bread it is all the same to me what I eat there is a very clever essay in one of the books upstairs upon much such a subject
Unknown	Was ascending the stairs leading to debray's apartments
19	About young girls that have been spoilt for home by great acquaintance the mirror I think I will look it out for you some day or other because I am sure it will do you good Catherine said no more and with an endeavour to do right

Unknown	And the conversation which had become interesting during the discussion of such serious affairs

Table 4.33: Actual Speech Test Data 5

Mixed Transcript	To hear you talk so much about the French bread at Northanger. I am sure, I do not care about the bread. It is all the same to me. What I eat. There is a very clever essay in one of the books upstairs upon much such a subject sending the stairs leading to The Breeze Apartments. I think I will look it out for you someday or other because I am sure it will do you good? Catherine said no more and with an Endeavor to do right in the conversation which had become interesting during the discussion of such serious affairs.

Table 4.34: Mixed Transcript Test Data 5

19	To hear you talk so much about the French bread at Northanger. I am sure, I do not care about the bread. It is all the same to me. What I eat. There is a very clever essay in one of the books upstairs upon much such a subject Apartments. I think I will look it out for you someday or other because I am sure it will do you good? Catherine said no more and with an Endeavor to do right
Unknown	sending the stairs leading to The Breeze in the conversation which had become interesting during the discussion of such serious affairs.

Table 4.35: Result of VTTSR Test Data 5

From Table 4.35, it can be seen when VTTSR including google cloud speech to text API, made a mistake by putting “Apartments” speech under speaker 19 which is wrong. However, slight mistake is still acceptable.

From the test result above, it can be said the performance of VTTSR is quite acceptable.

4.2.6 Discussion and Concluding Remark

In second part of the experiment with LibriSpeech datasets, three main experiments are done. First is single speaker identification, followed by multi-speaker identification in non-overlapped speech with speech extraction and speaker verification in non-overlapped speech. After these experiments, speaker verification is integrated with noise estimation module in order to form a simple Voice to Text Conversion App with Speaker Recognition.

Single Speaker Identification

For single speaker identification, it can be concluded that it performs very well on the classifying part as it scores perfect score of 100.0% accuracy for all 20 speakers registered. This has proven that GMM is a strong model for speaker identification. However, for the result of single speaker identification of single speaker identification in 4.1.2 Single Speaker Identification, there was a slight mistake in classification with accuracy of 97.00%. There are few possible reasons that causes this misclassification. First of all, the training size used is very small, with only 5 utterances per speaker while in 4.2.1 Single Speaker Identification, 30 utterances per speaker is used for training. Other than that, experiment in 4.1.2 Single Speaker Identification was done using audio recorded with a cheap quality mic of earphone while experiment in 4.2.1 Single Speaker Identification was done using audio from Librispeech datasets or phone mic recording.

As a conclusion for the single speaker identification experiments, GMM with MFCC works pretty well for single speaker identification task.

Multi-Speaker Identification (Non-Overlapped)

For multi-speaker identification (non-overlapped), two sub experiments are done. First sub experiment is that, the single word models mentioned in 3.2.2.3 Single Speaker Identification (Single Word) are built with word that are longer than 0.5s only, even when the test data is pass into the system and split into each word, only word longer than 0.5s is tested to determine the speaker before increasing the count. In the second sub experiment, words of every length are considered. These 2 was tested because a hypothesis is made that. The hypothesis is that word that are too short might not be able to describe the feature of speaker. However, this hypothesis is rejected as the sub experiment with all words give better result in both classification and diarization. First sub experiment gives accuracy of 73.0% in classification and 10.0% DER while second sub experiment gives accuracy of 83.0% in classification and 7.7% DER.

From these experiments, it can be said that the multi-speaker identification result is acceptable. However, the hypothesis testing experiment mentioned is tested due to trying to improve the single word version of single speaker identification works not as well as normal length single speaker identification. During the counting of word by distinct speaker, error is made, rather than scoring perfect score like normal single speaker identification. Therefore, confident count mentioned in 3.2.2.4 Multi-Speaker Identification (Non-Overlapped) is introduce to tackle this slight error, but the confident count theory is holding down the improvement of the result. From this, it can be deduced that GMM with MFCC doesn't work perfectly for short word. If there is a way to perfect the speaker identification of short word, the algorithm will be improved. However, there was also another problem. Even if the model counted correct number of speakers, sometimes Google Cloud Speech-To-Text API return wrong result by reducing the speaker tag by 1.

Speaker Verification

For speaker verification, using the GMM-UBM concept, it can be found that the speaker verification performs good with accuracy of 98.0% and have a perfect DET curve that can avoid tradeoff by adjusting threshold to -1.475 ~ -0.667. However, this is not practical, this may be the result due to the lack of testing data or the training data use for target speaker might be too large, therefore, the GMM describe the target speaker too well.

Voice to Text Conversion App with Speaker Recognition

Voice to Text Conversion App with Speaker Recognition is the final experiment of this project. In this experiment, target speaker is registered. And UBM-GMM built in speaker verification experiment is used. After each word in the speech is marked with unknown speaker tag by Google Cloud Speech-To-Text API speaker diarization, the audio is chopped into sequence based on the tag before classifying it as target speaker speech or noise speech from the unknown speaker.

In conclusion, Voice to Text Conversion App with Speaker Recognition works to a certain extend with acceptable result. It can be applied into the real world by implementing it in speech transcription app. Although it won't be help much if the overlapping between target speaker speech and unknown noise is a lot, it can still improve the result of transcription if there was a bit of overlapping only. This is useful already, as in real world, overlapping can't be occurring over the whole speech duration.

4.3 Objectives Evaluation

In this project there are 3 main objectives.

- To develop a model that can recognize speaker voice with audible background noise such as speech.

This is achieved in 4.2.2 Multi-Speaker Identification (Non-Overlapped) with Training Audio Longer Than or Equal to 0.5s and 4.2.3 Multi-Speaker Identification (Non-Overlapped) with Training Audio of All Lengths. The models used in these experiments have proven that the models can recognize speaker voice of multiple speakers in a speech sequence. In order to mark others as background noise, just set one speaker as the target speaker.

- To determine the background noise to be suppressed to improve voice recognition using voice feature extraction.

In 4.2.5 Voice to Text Conversion App with Speaker Recognition, the end result has reached the state where it can extract the target speech from mixed transcript with noise speech. This will be able to improve voice recognition.

- To identify the speaker voice by utilizing Mel Frequency Cepstral Coefficient (MFCC) features.

This is the primary milestone of this project. This is the first objective achieved in order to continue in the research of this project. This is achieved in the experiments in 4.1.2 Single Speaker Identification and 4.2.1 Single Speaker Identification with a pretty high overall accuracy and it can be concluded that MFCC features is good in describing single speaker features.

CHAPTER 5: CONCLUSION

5.1 Summarization of Finding

In this project, it has been found out that MFCC and delta MFCC can describe the vocal traits of a person. By training GMM with MFCC and delta MFCC, the speaker can be identified. Not only that, in order to identify two simultaneous speakers in overlapped speech, training GMM using overlapped speech can work to a certain extend. Not only that, GMM with MFCC and delta MFCC performs very well for single speaker. However, training set need to be larger and the tested data must be longer. GMM does not performs well for test data that are too short, for example test data below 1 seconds. Lastly, UBM can be made with large datasets for speaker verification purpose.

5.2 Novelties of Work

In this project, the novelty of the work is that silence removal is applied prior to extraction vocal traits such as MFCC and delta MFCC. This can further improve the system as silence period will be recorded as the vocal traits if not removed. Furthermore, combination has also been applied to artificially mix 2 speakers training data together in order to train GMM to identify 2 simultaneous speakers. This is more practical than collection real overlapped speech from the registered speakers. Other than that, GMM that are trained using single word utterances is also used to determine the number of different enrolled speakers in a test data. Lastly, GMM and UBM are used together for speaker verification.

5.3 Concluding with Supportive Remark

In this project, the objective is to identify the speaker and transcribe the voice into text. A solution is proposed based on python speech recognition library and GMM. First of all, silence period will be removed from the audio. Then, the audio is windowed and hammed before extracting MFCC and delta MFCC. The features are then use to train GMM for single speaker identification. For two simultaneous speaker identifications, the audio is combined and undergo the same process before training GMM. Lastly, the audio will be converted into text. The single speaker identification on self-generated datasets has achieved an accuracy of 96.00% while the simultaneous speaker identifications on self-generated datasets only achieved an accuracy of 72.31%.

Next up, Google Cloud Speech-To-Text API is also use for speech recognition purpose. Google Cloud Speech-To-Text API has better overall performance than python speech recognition

library. For single speaker identification on 19 speakers from LibriSpeech datasets and myself, a perfect 100% accuracy is achieved. This has proven that GMM is good for single speaker identification with suitable amount of training data with good quality mic. For multi-speaker identification on the non-overlapped speech, there are two experiments done, the experiment that give better result gives an accuracy of 83% and DER of 7.7% only. Lastly, for speaker verification, it performs with 98% accuracy and are able to avoid tradeoff when adjusting threshold. Avoiding tradeoff is usually impossible, the possible reason to cause this may be the lack of testing data.

Last but not least, testing of Voice to Text Conversion App with Speaker Recognition is created finally. With speaker 19 from LibriSpeech datasets as the target, the non-overlapped speech from other unknown speaker is able to be removed with optimal result.

5.4 Recommendation

In this project, there are few things that are not achieved. One of them is speech extraction for overlapped speech. By overlapping main target speech with noise speech of a lower volume, few non-machine learning methods are used to try and remove the noise speech. However, all the methods perform worse than recognizing the speech directly using speech recognizer such as Google Cloud Speech-To-Text API. First method is by lowering the test audio volume and remove silence period. Not only that, method like amplifying and low-pass filter are also used. Low-pass filter is a technique to filter out frequency below threshold, but all of them ended in a failure, it can be seen that filtering frequency can’t suppress non-dominant voice in overlapped speech. Methods that should be tested in the future are technique using recurrent neural network, LSTM and more on deep learning methods. It was a regret as in this project, due to the limitation of computation power and datasets, deep learning method can’t be tested that much. Not only that, a better method should also be used to identify the number of speakers in multi-speaker speech. LSTM method in diarization should be used for speaker diarization rather than GMM as GMM does not perform well for audio below 1s. Each word is usually less than 1 seconds, therefore identifying the speaker of each word using GMM is not that accurate.

References

[1] Sonix, "A Short History of Speech Recognition," Sonix. [Online]. Available: https://sonix.ai/history-of-speech-recognition#:~:text=The%20first%20speech%20recognition%20systems,to%2016%22words%20in%20English.

[2] N. Kanda et al., "Joint Speaker Counting, Speech Recognition, and Speaker Identification for Overlapped Speech of Any Number of Speakers," *INTERSPEECH 2020*, pp. 36-40, 2020.

[3] N. Kanda et al., "Simultaneous Speech Recognition and Speaker Diarization for Monaural Dialogue Recordings with Target-Speaker Acoustic Models," *Conference: 2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, 2019.

[4] Q. Wang, C. Downey, L. Wan, P. A. Mansfield and I. L. Moreno, "Speaker Diarization with LSTM" *Proc. ICASSP*, pp. 5239–5243, 2018.

[5] D. Garcia-Romero, D. Snyder, G. Sell, D. Povey and A. McCree, "Speaker Diarization using Deep Neural Network Embeddings," *Proc. ICASSP*, pp. 4930–4934, 2017.

[6] Q. Wang et al., "VoiceFilter-Lite: Streaming Targeted Voice Separation for On-Device Speech Recognition," *INTERSPEECH 2020*, pp. 2677-2681, 2020.

[7] D. Guffanti, D. Martinez, J. Paladines and A. Sarmiento, "Continuous Speech Recognition and Identification of the Speaker System," *Proceedings of the International Conference on Information Technology & Systems (ICITS 2018)*, pp. 767-776, 2019.

[8] O. Khalifa, R. Islam, S. Khan, M. Faizal and D. Dol, "Text Independent Automatic Speaker Recognition," *3rd International Conference on Electrical and Computer Engineering*, Dhaka, Bangladesh, pp. 561-564, 2004.

[9] R. Jahangir et al., "Text-Independent Speaker Identification Through Feature Fusion and Deep Neural Network," *IEEE Access Volume 8*, pp. 32187-32202, 2020.

[10] V.T. Tran and W.H. Tsai, "Speaker Identification in Multi-Talker Overlapping Speech Using Neural Networks," *IEEE Access Volume 8*, pp. 134868-134879, 2020.

[11] M. Fabien, "Speaker Verification using Gaussian Mixture Model (GMM-UBM)," MF, 2019. [Online]. Available: https://maelfabien.github.io/machinelearning/Speech1/#limits-of-gmm-ubm

[12] PyPI, “SpeechRecognition 3.8.1,” PyPI. [Online]. Available: https://pypi.org/project/SpeechRecognition/

[13] Scikit Learn, “2.1. Gaussian mixture models,” Scikit Learn. [Online]. Available: https://scikit-learn.org/stable/modules/mixture.html#:~:text=A%20Gaussian%20mixture%20model%20is,Gaussian%20distributions%20with%20unknown%20parameters.

[14] B. Murugan, “Audio Processing and Remove Silence using Python,” 2020. [Online]. Available: https://ngbala6.medium.com/audio-processing-and-remove-silence-using-python-a7fe1552007a

[15] A. Chapagain, “Speech Signal Processing using python,” 2020. [Online]. Available: https://aadityachapagain.com/2020/08/asr-mfcc-filterbanks/

[16] J. Lyons et al., “python speech feature,” 2020. [Online]. Available: https://github.com/jameslyons/python_speech_features

[17] A. Bhapkar, “Speaker Identification Using Machine Learning,” 2019. [Online]. Available: https://medium.com/analytics-vidhya/speaker-identification-using-machine-learning-3080ee202920

Weekly Log

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: Y3S3	**Study week no.: 5**
Student Name & ID: Ang Sea Zhe 18ACB01470	
Supervisor: Mr. Tou Jing Yi	
Project Title: Voice to Text Conversion App with Speaker Recognition	

1. WORK DONE

[Please write the details of the work done in the last fortnight.]

- Tried to implement determination model.
- Tried to implement speech extraction function.

2. WORK TO BE DONE

- Optimize the speech extraction function.
- Update report.

3. PROBLEMS ENCOUNTERED

- Speech recognition perform worse after speech extraction

4. SELF EVALUATION OF THE PROGRESS

- Lack of digital signal processing (dsp) and deep learning knowledge is halting progress.

Supervisor's signature

Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: Y3S3	**Study week no.: 6**
Student Name & ID: Ang Sea Zhe 18ACB01470	
Supervisor: Mr. Tou Jing Yi	
Project Title: Voice to Text Conversion App with Speaker Recognition	

1. WORK DONE

[Please write the details of the work done in the last fortnight.]

- Continue to research on dsp to improve speech extraction module.
- Researched a bit on training speech recognition model from scratch.

2. WORK TO BE DONE

- Confirm direction with supervisor.
- Continue improve speech extraction module.

3. PROBLEMS ENCOUNTERED

- Limited data and computation power to train speech recognition model.

4. SELF EVALUATION OF THE PROGRESS

- Progressing steadily.

Supervisor's signature

Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: Y3S3	**Study week no.: 7**
Student Name & ID: Ang Sea Zhe 18ACB01470	
Supervisor: Mr. Tou Jing Yi	
Project Title: Voice to Text Conversion App with Speaker Recognition	

1. WORK DONE

[Please write the details of the work done in the last fortnight.]

- Discussed with supervisor and not to train speech recognition model from scratch.
- Started free google cloud account with 300-dollar credit.
- Started to explore on google cloud speech to text api.

2. WORK TO BE DONE

- Use google cloud speech to text api for diarization.
- Determine number of speakers.

3. PROBLEMS ENCOUNTERED

- Unsure the limit of the speaker diarization of google cloud speech to text api.

4. SELF EVALUATION OF THE PROGRESS

- Progressing steadily on implementation but not the report.

Supervisor's signature

Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: Y3S3	**Study week no.: 8**
Student Name & ID: Ang Sea Zhe 18ACB01470	
Supervisor: Mr. Tou Jing Yi	
Project Title: Voice to Text Conversion App with Speaker Recognition	

1. WORK DONE

[Please write the details of the work done in the last fortnight.]

- Implemented single word speaker identification model to determine number of speakers.
- Implemented speaker diarization using google cloud speech to text api

2. WORK TO BE DONE

- Start to work more on the report.

3. PROBLEMS ENCOUNTERED

- Unsure the limit of the number of speaker I should go for.

4. SELF EVALUATION OF THE PROGRESS

- Progressing steadily on implementation but not the report.

Supervisor's signature

Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: Y3S3	**Study week no.: 9**
Student Name & ID: Ang Sea Zhe 18ACB01470	
Supervisor: Mr. Tou Jing Yi	
Project Title: Voice to Text Conversion App with Speaker Recognition	

1. WORK DONE

[Please write the details of the work done in the last fortnight.]

- Researched on Speaker Verification technique.
- Started to edit chapter 3 of the report.

2. WORK TO BE DONE

- Start to work more on the report.
- Implement Speaker Verification.

3. PROBLEMS ENCOUNTERED

- Unknown about some report flow.

4. SELF EVALUATION OF THE PROGRESS

- Progressing steadily on implementation but not the report.

Supervisor's signature

Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: Y3S3	**Study week no.: 10**
Student Name & ID: Ang Sea Zhe 18ACB01470	
Supervisor: Mr. Tou Jing Yi	
Project Title: Voice to Text Conversion App with Speaker Recognition	

1. WORK DONE

[Please write the details of the work done in the last fortnight.]

- Written Chapter 3 and a bit on chapter 4
- Implemented Speaker Verification

2. WORK TO BE DONE

- Plan out Experiment for chapter 4 and evaluate them.

3. PROBLEMS ENCOUNTERED

- Need to research more on way of evaluating experiment.

4. SELF EVALUATION OF THE PROGRESS

- Progressing steadily.

Supervisor's signature

Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: Y3S3	**Study week no.: 11**
Student Name & ID: Ang Sea Zhe 18ACB01470	
Supervisor: Mr. Tou Jing Yi	
Project Title: Voice to Text Conversion App with Speaker Recognition	

1. WORK DONE

[Please write the details of the work done in the last fortnight.]

- Done evaluating most of the experiment and written down the report

2. WORK TO BE DONE

- Wrapping up the report and fixing the report flow.

3. PROBLEMS ENCOUNTERED

- Don't know should the failed experiment should be kept in the report.

4. SELF EVALUATION OF THE PROGRESS

- Progressing steadily.

Supervisor's signature

Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: Y3S3	**Study week no.: 12**
Student Name & ID: Ang Sea Zhe 18ACB01470	
Supervisor: Mr. Tou Jing Yi	
Project Title: Voice to Text Conversion App with Speaker Recognition	

1. WORK DONE

[Please write the details of the work done in the last fortnight.]

- Fixing the report

2. WORK TO BE DONE

- Finishing up the report.
- Record Presentation.

3. PROBLEMS ENCOUNTERED

- None.

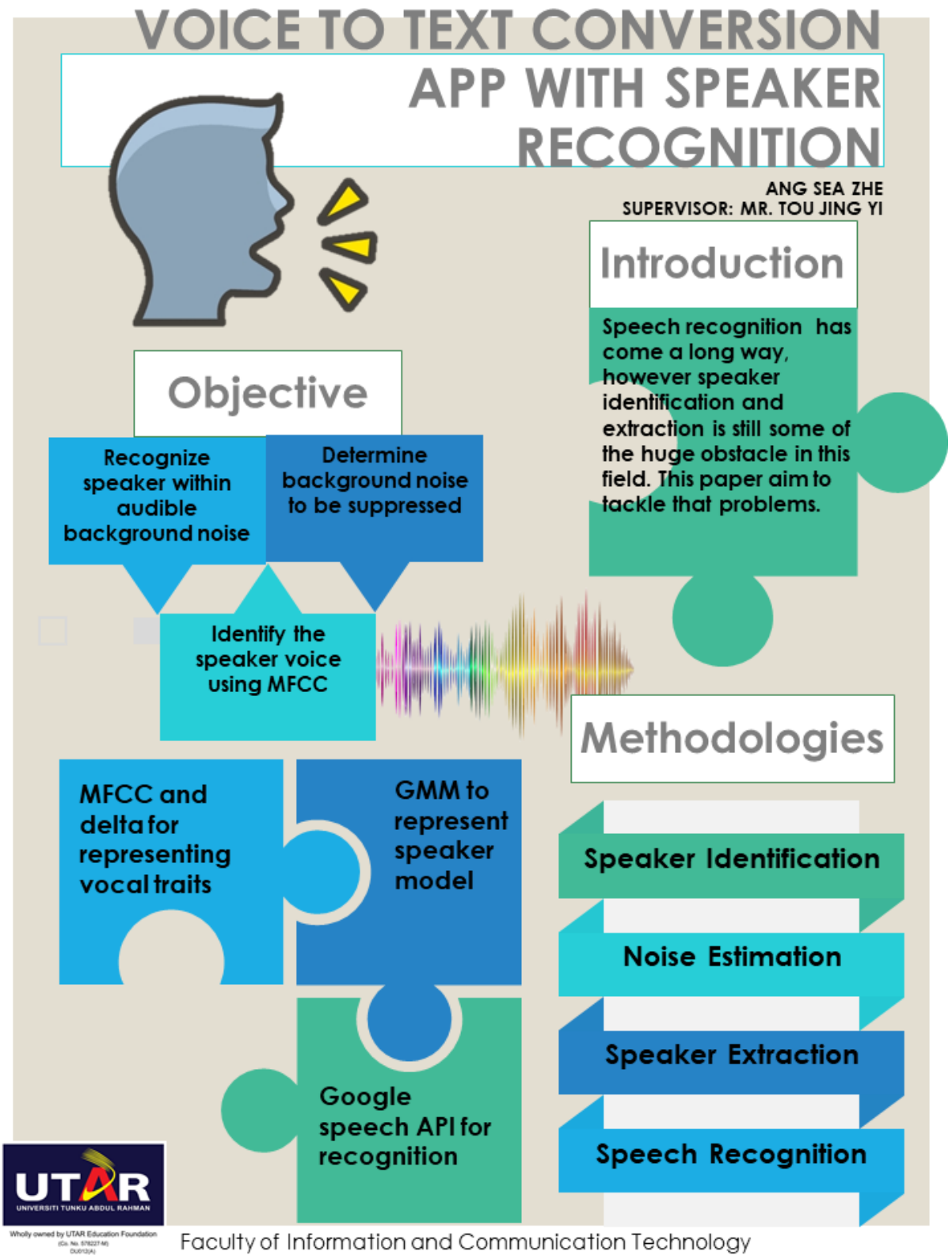
4. SELF EVALUATION OF THE PROGRESS

- Wrapping up.

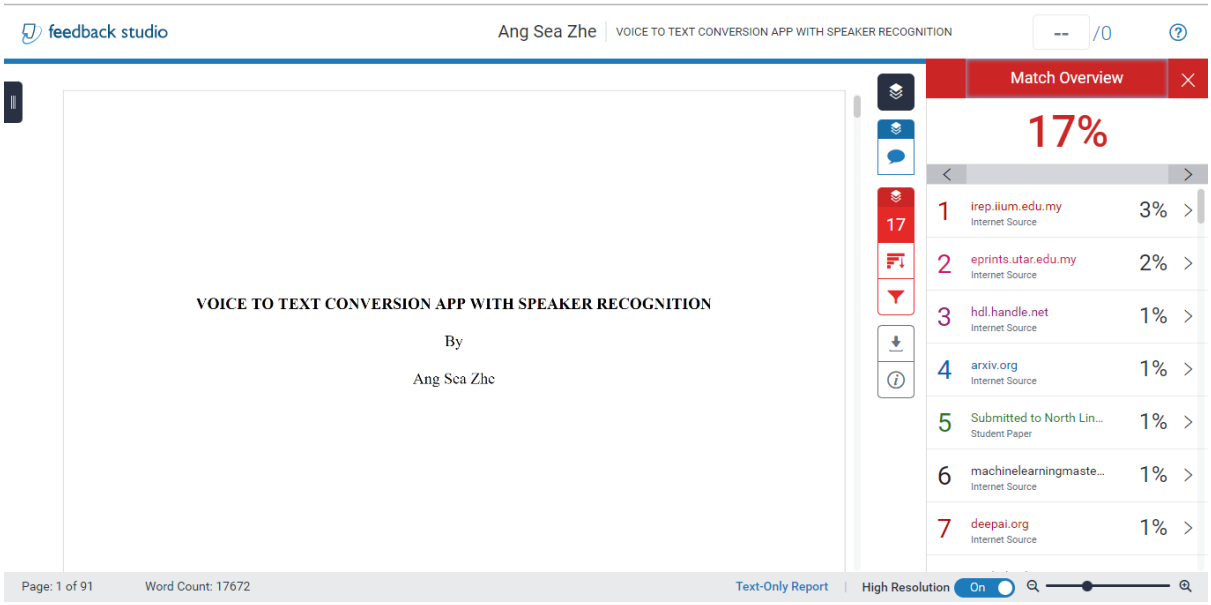
Supervisor's signature

Student's signature

Poster


VOICE TO TEXT CONVERSION APP WITH SPEAKER RECOGNITION
ANG SEA ZHE
SUPERVISOR: MR. TOU JING YI
Introduction
Speech recognition has come a long way, however speaker identification and extraction is still some of the huge obstacle in this field. This paper aim to tackle that problems.
Objective
Recognize speaker within audible background noise
Determine background noise to be suppressed
Identify the speaker voice using MFCC
Methodologies
MFCC and delta for representing vocal traits
GMM to represent speaker model
Google speech API for recognition
Speaker Identification
Noise Estimation
Speaker Extraction
Speech Recognition
UTAR
UNIVERSITI TUNKU ABDUL RAHMAN
Faculty of Information and Communication Technology

Plagiarism check result



Universiti Tunku Abdul Rahman			
Form Title : Supervisor's Comments on Originality Report Generated by Turnitin for Submission of Final Year Project Report (for Undergraduate Programmes)			
Form Number: FM-IAD-005	Rev No.: 0	Effective Date: 01/10/2013	Page No.: 1of 1

FACULTY OF INFORMATION AND COMMUNICATION TECHNOLOGY

Full Name(s) of Candidate(s)	Ang Sea Zhe
ID Number(s)	18ACB01470
Programme / Course	Bachelor of Computer Science
Title of Final Year Project	Voice to Text Conversion App with Speaker Recognition

Similarity	**Supervisor's Comments (Compulsory if parameters of originality exceeds the limits approved by UTAR)**
Overall similarity index: 17 **%** **Similarity by source** Internet Sources: 12 % Publications: 8 % Student Papers: 5 %	
Number of individual sources listed of more than 3% similarity: 0	
Parameters of originality required and limits approved by UTAR are as Follows: **(i) Overall similarity index is 20% and below, and** **(ii) Matching of individual sources listed must be less than 3% each, and** **(iii) Matching texts in continuous block must not exceed 8 words** ***Note: Parameters (i) – (ii) shall exclude quotes, bibliography and text matches which are less than 8 words.***	

Note Supervisor/Candidate(s) is/are required to provide softcopy of full set of the originality report to Faculty/Institute

Based on the above results, I hereby declare that I am satisfied with the originality of the Final Year Project Report submitted by my student(s) as named above.

Signature of Supervisor

Name: Tou Jing Yi

Date: 22/4/2022

Signature of Co-Supervisor

Name: _____

Date: _____

FYP 2 Checklist

UNIVERSITI TUNKU ABDUL RAHMAN

FACULTY OF INFORMATION & COMMUNICATION TECHNOLOGY
(KAMPAR CAMPUS)

CHECKLIST FOR FYP2 THESIS SUBMISSION

Student Id	18ACB01470
Student Name	Ang Sea Zhe
Supervisor Name	Mr. Tou Jing Yi

TICK (√)	DOCUMENT ITEMS Your report must include all the items below. Put a tick on the left column after you have checked your report with respect to the corresponding item.
	Front Plastic Cover (for hardcopy)
√	Title Page
√	Signed Report Status Declaration Form
√	Signed FYP Thesis Submission Form
√	Signed form of the Declaration of Originality
√	Acknowledgement
√	Abstract
√	Table of Contents
√	List of Figures (if applicable)
√	List of Tables (if applicable)
	List of Symbols (if applicable)
√	List of Abbreviations (if applicable)
√	Chapters / Content
√	Bibliography (or References)
√	All references in bibliography are cited in the thesis, especially in the chapter of literature review
	Appendices (if applicable)
√	Weekly Log
√	Poster
√	Signed Turnitin Report (Plagiarism Check Result - Form Number: FM-IAD-005)
√	I agree 5 marks will be deducted due to incorrect format, declare wrongly the ticked of these items, and/or any dispute happening for these items in this report.

*Include this form (checklist) in the thesis (Bind together as the last page)

I, the author, have checked and confirmed all the items listed in the table are included in my report.

_____
(Signature of Student)
Date: 20/4/2022