MANIPAL
ACADEMY of HIGHER EDUCATION
(Deemed to be University under Section 3 of the UGC Act, 1956)

MANIPAL SCHOOL OF INFORMATION SCIENCES
(A Constituent unit of MAHE, Manipal)

Development of template matching technique for speech to text conversion

Project Status 2

submitted to

Manipal School of Information Sciences, MAHE, Manipal

Reg. Number	Name	Branch
201046001	GIRISH KUMAR	BDA
201046016	ANKIT KUMAR PATHAK	BDA

Under the guidance of

Raghudathesh G P

Assistant Professor on Contract,

Manipal School of Information Sciences,

MAHE, MANIPAL

26/12/2020

MANIPAL SCHOOL OF INFORMATION SCIENCES
MANIPAL
(A constituent unit of MAHE, Manipal)

DECLARATION

We declare that this mini project, submitted for the evaluation of course work of Mini Project to Manipal School of Information Sciences, is an existing idea/concept/code available at (**https://jonathan-hui.medium.com/speech-recognition-acoustic-lexicon-language-model-aacac0462639**, **https://medium.com/@kangeugine/hidden-markov-model-7681c22f5b9**)), conducted under the supervision of my guide **Prof. Raghudathesh G P** and panel members, **Prof. SamarendranathBhattacharya**, **Dr. Harishchandra Hebbar** References, help and material obtained from other sources have been duly acknowledged.

ABSTRACT

The fundamental of speech dates back in time wherein humans began to communicate with one other. Over the time it was realized that it is an effective medium of commutation. Later on, it acquired various forms and thus different languages came into existence. On close analysis it has been found that there are over hundreds of different languages; English being the most widely used.

Accordingly, most the formal talks are in English. Most of the communication devices these days like security devices, home appliances, mobile phones, ATM machines, computers and hotels use speech processing. Our project focuses on establishing an interface between these two. The human computer interface has been developed in order to communicate and interact with ones who are suffering from different types of disabilities. Speech-to-Text Conversion (STT) system is advantageous for deaf and dumb people. It is also used in our day to day lives. The main aim of our system is to convert input speech signals into text as output. This project extract features of the speech signal by Mel-Frequency Cepstral Coefficients(MFCC) for multiple isolated words. Gaussian Mixture Model (GMM) and Hidden Markov Model (HMM) are used to train the audio files to get the spoken word recognized.

Keywords

Mel-Frequency Cepstral Coefficients(MFCC), Gaussian Mixture Model (GMM), Hidden Markov Model (HMM)

Contents

LIST OF FIGURES

ABBREVIATIONS

MFCC	Mel-Frequency Cepstral Coefficients
GMM	Gaussian Mixture Model
HMM	Hidden Markov Model

1. Objective

1.1 Introduction

Speech is the most basic of the means of human communication. In parallel to communication systems improvements, computer science developments considerably change the ways of interaction between people. Research in speech makes significant progress in speech synthesizers, speech transmission systems and automatic speech recognition (ASR). Yet speech technology can be greatly improved and speech recognition is still far from the desired goal of a machine able to understand spoken discourse on any subject by all speakers in all environments.

Speech recognition depends on a lot of different context variations and environmental conditions. Ideally speech recognition should be speaker independent, accept natural language and unrestricted lexicon. This perfect scenario must be replaced in a real scenario context considering that human hearing is much more complex than the signal processing techniques currently implemented. Some issues to be taken into account are :

- Background noise: fans, computers, machinery running.

- Speech interference: TV, radio background conversation.

- Sound reflections due to the room geometry.

- Non stationary events: door slams, irregular road noise, car horns.

- Signal degradation : microphone and transmission system distortions.

- Unknown words: improper English grammar, unfamiliar accent, out-of-vocabulary words.

- Unusual circumstances: stressed speaker.

- Speaker sound artifacts: speaker lip smacks, heavy breathing, mouth clicks and pops.

Background noise and speech interference as well as acoustic reflection can be reduced by choosing a proper microphone set close to the speaker and pointed to the desired source, while non stationary noise is difficult to handle and will probably lead to a bad trial that must be repeated. The signal degradation due to the material is neglected if the same microphone and system is used for the training and for the testing. This condition is necessary to obtain reasonable performances and also helps taking into account the previously cited acoustic interference. The latter points are more connected to the speaker, and their influence depends on the application and on the techniques used to handle them.

For example grammatical rules can be used to improve the system robustness by penalizing incorrect succession of words. However the use of grammatical rules assumes that the speaker knows the rules and uses them correctly. More difficult is the accent of the speaker and the way to handle stressed voices and artifacts. This is why the best results come from speaker dependent systems, where these characteristics are taken into account. Even some more restrictions of speech input on the size of the

vocabulary for example can improve the recognition capabilities of the system.

1.2 Problem definition

The goal of this synopsis is to build a real time speech recognizer demonstrator using a template matching approach The application would generate a visual feedback for the user after the speech recognition is performed, simply by displaying the recognized word and thus providing an interface to show and test the recognition capabilities.

1.3 Automatic Speech Recognition

Nowadays speech signal is commonly represented as a sequence of vectors in an acoustic space. It is acquired that two sequences of the same succession of pronounced words have some similarities. The speech recognition problem becomes “simply” the measure of the dissimilarity between prerecorded and processed sequences and the speech pronounced. A decision rule will be used to decide with a certain confidence which words have been pronounced. The major work in speech recognition consists in enhancing the algorithm used for this method.

In ASR, the main task is to derive a sequence of words from a stream of acoustic information. One further processing step for an application is speech understanding. It includes a language analyzer and an expert system to select the correct/desired output, then to replace it in the context and finally to derive an appropriate command representing the user expectation. Sometimes ASR can also be seen as a process which includes speech understanding. But most of the time ASR and speech understanding are strongly linked and difficult to clearly distinguish, as the processing of each one depends closely on the other. In this work we will focus on the speech recognition process. The structure of any speech recognition system follows the three basic steps represented in Figure 1, namely feature extraction, pattern comparison and decision rule.

2. Project design status

2.1 Architecture

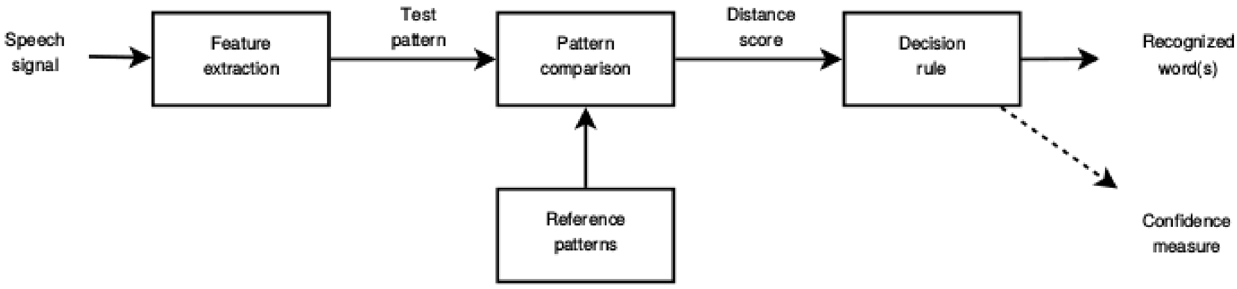


Figure 1: Structure of ASR systems

2.2 Implementation

2.2.1 Acquisition of speech recordings from user

Using the audacity tool speech samples will be collected.

2.2.2 Segmentation of speech recordings at phoneme level

i) Feature Extraction

The most easiest and prevalent method to extract spectral features is calculating the Mel-Frequency Cepstral Coefficients (MFCC) from human voice. It is one of the most popular methods of feature extraction used in speech recognition systems. It is based on frequency domain using the Mel scale which is based on the human ear scale. Time domain features are less accurate than the frequency domain features. The main aim of feature extraction is to reduce the size of the speech signal before the recognition of the signal. Steps involved in feature extraction are pre-emphasis, framing, windowing, fast fourier transform, Mel-frequency filtering, Logarithmic function and Discrete Cosine Transform etc.

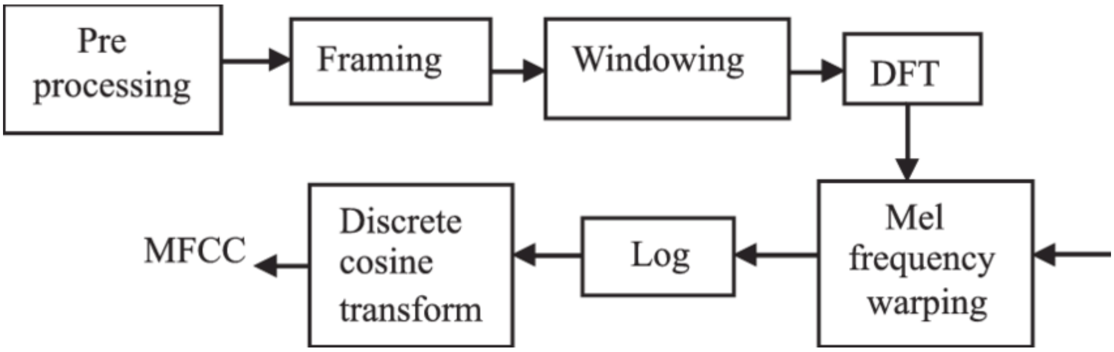


Figure 2: MFCC Feature Extraction

The first step in MFCC is pre-emphasis which is used to boost the high frequencies of a speech signal which are lost during speech production. Pre-emphasis is needed because high frequency components of the speech signal have small amplitude with respect to low frequency components. Therefore higher frequencies are artificially boosted in order to increase the signal-to- noise ratio. Next, is framing which is used to block the frames obtained by analog to digital conversion (ADC) of speech signal . The number of samples in each frame is chosen as 256 and the number of samples overlapping between adjacent frames is 128. Overlapping frames are used to acquire the information from the boundaries of the frame. Due to discontinuities at the start and the end of the frame causes undesirable effects in the frequency response, so windowing is used to eliminate the discontinuities at the edges. Hamming window is used which introduces least amount of distortion. Generalized hamming window equation is

$$w(n)=\alpha-\beta\cos\left(\frac{2\Pi n}{N-1}\right)$$

After windowing, Fast Fourier Transform(FFT) is measured for every frame to extract the frequency components of the signal in time domain. Speech signal does not follow linear frequency scale used in FFT. Hence Mel-scale is used for feature extraction which is directly proportional to the logarithm of linear frequency. Equation is used to convert linear scale frequency into Mel-scale frequency.

$$Mel(f)=2595\log_{10}\left(1+\frac{f}{700}\right)$$

Triangular bandpass filters are used to extract the spectral envelope. 20 filters are used. Log is applied to the absolute magnitude of the coefficients of which is obtained after Mel-scale conversion. Discrete cosine transform (DCT) converts the Mel-frequency domain into time domain

$$C(k)=\sum_{m=0}^{M-1} S(m)\cos(\Pi k(m+1/2)/M),0\leq k<K$$

The value of K ranges between 8 and 13. We choose K as 13 and hence we obtain 13 coefficients for each frame.

ii) Pattern Classification

Pattern classification mainly consists of the development or training of a system (given a feature vector) which will divide a large number of individual examples into groups called classes. As the source of the speech is often due to a large amount of many causes, the available speech signal results of the combination of the audio channel, noise, additive noise, etc. A classical assumption is that these different sources are not correlated to the message being communicated.

iii) Parametric vs Non-Parametric Classification

Parametric models summarize information on parameters, making some assumptions about the data distribution. They provide a flexible mathematical framework and since a lot of training data are available to estimate the parameters, they are widely used.

Non-parametric models do not make any assumption and use real data for decoding the new sequence but they also require even more data than parametric models.

Unfortunately for some applications, for example when the lexicon is specific or defined by the customer, not enough data are available to train a statistical framework. It is even worse for non-parametric models. An alternative approach with less limiting factors, without prior parameters estimation, which supplies a more realistic model and represents speech more faithfully has to be used.

iv) HMM

HMM take into account the inherent statistical variations in speaking rate and pronunciation. It uses a temporal sequence without requiring any explicit segmentation in terms of speech units (phones or phonemes), but the optimum parameters are estimated by a training procedure and require a large amount of data. Furthermore the system must be retrained to decode properly an unknown sequence.

Additionally in order to use the HMM representation two strong assumptions have to be made :

• The features extracted within a phonetic segment should be uncorrelated with one another.

• Each speech segment is piecewise stationary.

A Markov chain contains all the possible states of a system and the probability of transiting from one state to another. The probability of observing an observable given an internal state is called the emission probability. The probability of transiting from one internal state to another is called the transition probability.

GMM models the observed probability distribution of the feature vector given a phone. It provides a principled method to measure “distance” between a phone and our observed audio frame.

On the other hand, HMM produces a principled model on how states are transited and observed. As the probability of the observations can be modeled with HMM as

$$P(x)=\sum_{h} P(h)P(x \vee h) \qquad \text{(HMM)}$$

where h is the hidden state (phone). The likelihood of features given phone can be modeled with GMM.

$$P(x \vee h)=\sum_{j} p_i N(\mu_j,\Sigma_j) \qquad \text{(GMM)}$$

2.3 Template matching of segmented phone for each recording

When a pronunciation dictionary is not available and there are only a few samples per word, template matching (TM) seems to be the most suitable approach. A template is a collection of vectors of features 1 repeating a particular pronunciation and can also be seen as the succession of frames. TM stores during training one or more reference templates 2 per word. During testing phase each new frame is compared with all of the reference frames and identifies the new utterance as being the word associated with the template with the smallest distance to the new sequence. In TM all information contained in the templates is kept and used to recognize the pronounced word, no a priori assumptions are made and a word can be identified by only a few samples. However, its generalization capabilities are weak and its performances are not as competitive as HMM-based approaches.

The TM approach is subject to the following drawbacks:

1. Separate template for each word brings dependency with the lexicon size in opposition to smaller units like phones or phonemes.

2. Nonlinear time alignment is crucial (inevitable different speaking rates, even for a same speaker, we have different speaking rates for the same word).

3. Reliability determine the word boundaries.

2.4 Decision Rule

The Gaussian Mixture Model(GMM) is a parametric probability density function which is represented as a weighted sum of Gaussian component densities. It is used as a parametric model of probability distribution of measuring features in biometric systems. Gaussian Mixture Model(GMM) is used as a classifier to compare the features extracted from the MFCC with the stored templates. Gaussian Mixture Model is represented by its Gaussian distribution and each Gaussian distribution is calculated by its mean, variance and weight of the Gaussian distribution. Gaussian Mixture density is weighted sum of M component densities and can be expressed:

$$p(\bar{x} \vee \lambda) = \sum_{i=1}^{M} p_i b_i(\bar{x})$$

bi (x) – component densities, that can be written:

$$b_i\left(\bar{x}\right)=\frac{1}{2\Pi^{D/2}\left|\sum_i\right|^{1/2}}e^{-1/2\left(x^2-\mu_i\right)'\sum_i^{-1}\left(x^2-\mu_i\right)}$$

where = mean vector Gaussian Mixture Model is described by the mean vectors, co-variance matrices and mixture weights from all component densities. These parameters are represented by the notation:

λ = {p$_i$ μ_i i=1, 2, 3

Every speaker is shown by his GMM and is referred to his model . Plenty of techniques are available for calculating the parameters of GMM. One of the most popular methods is Maximum Likelihood(ML) estimation. It finds out the model parameters which maximize the likelihood of GMM. For T training vectors {x1,. . . . . .,xT} the GMM likelihood is given as:

$$P\left(X\vee t\right)=\prod_{t=1}^{T}P\left(x_t\vee\lambda\right)$$

The above equation is a non-linear function of so the iterative Expectation-Minimization(EM) algorithm is used for training and matching. Thus the following parameters are calculated in the iterations:

Mixture weight :

$$\bar{w}_i=\frac{1}{T}\sum_{t=1}^{T}Pr\left(i\vee x_t,\lambda\right)$$

Mean :

$$\bar{\mu}_i=\frac{\sum_{t=1}^{T}Pr\left(i\vee x_t,\lambda\right)x_t}{\sum_{t=1}^{T}Pr\left(i\vee x_t,\lambda\right)}$$

Variance :

$$\bar{\sigma}^2 = \frac{\sum_{t=1}^{T} Pr\left(i \vee x_t, \lambda\right) x_t^2}{\sum_{t=1}^{T} Pr\left(i \vee x_t, \lambda\right)} - \bar{\mu}_i^2$$

The a posteriori probability for component i is given as follows:

$$P\left(x_t \vee \lambda\right) = \frac{w_i g\left(x_t \vee \mu_i, \Sigma_i\right)}{\sum_{k=1}^{M} w_k g\left(x_t \vee \mu_k, \Sigma_k\right)}$$

These iterative steps are carried out for matching purposes in real-time and the Euclidean distance is found out between various database, hence a correct match is found.

2.5 Confidence Measure

As the decision rule for the recognition is based on a distance measure it is possible to introduce a confidence measure to provide a measure of certainty of the detection. In this work confidence measure is used to implement a rejection task. The aim is to accept the correct decision of the recognizer and to reject the false detection.

2.6 Displaying of uttered/spoken word

Spoken word will be compared with the template phoneme datasets and text will be displayed as the output.

3. Test environment creation

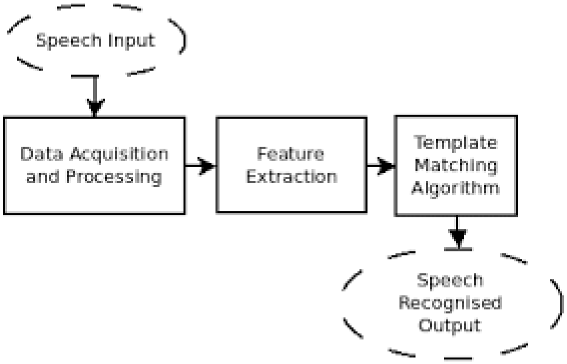


Figure 3: Speech Recognition

Four words ‘yes’, ‘no, ‘left’, ‘right’ of different variations in the speech are recorded. 20 variations per word saved in the folder labelled with the respective words. These recorded words are fed into the speech recognition program to train and accuracy and confusion matrix are obtained.

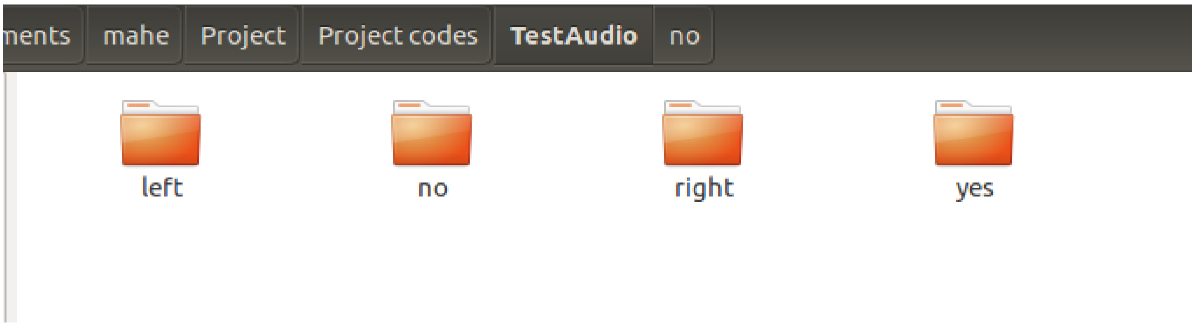


Figure 4: Folder containing all the audio samples

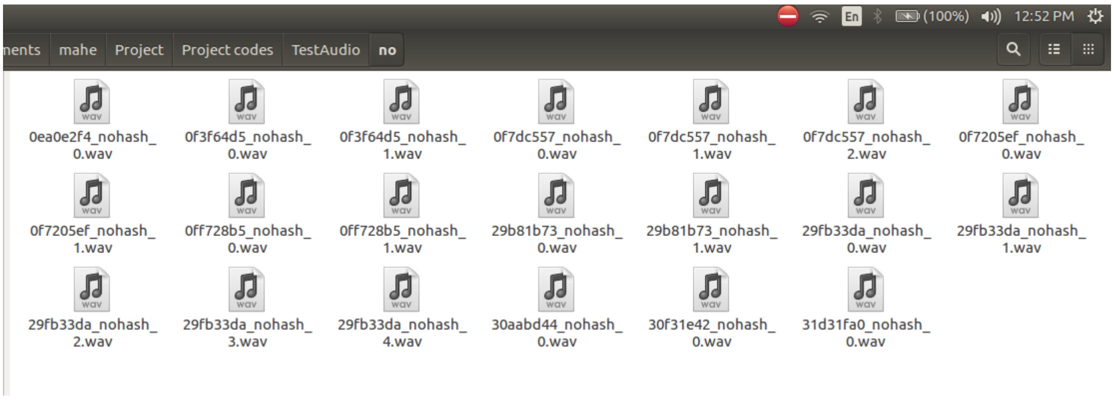
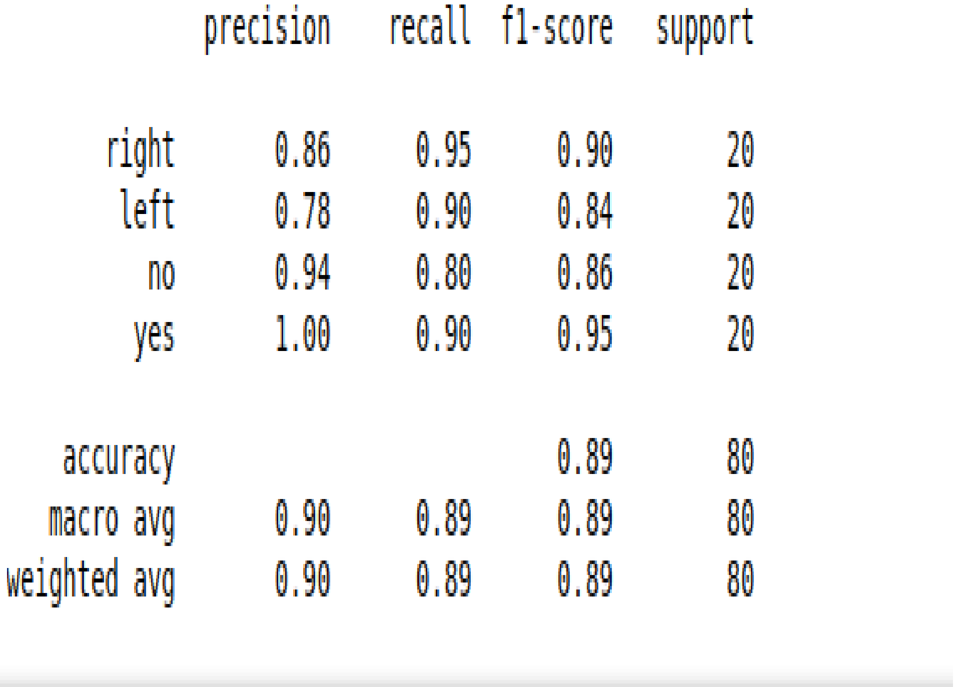


Figure 5: Different variation Audio samples for the word ‘no’

4. Result obtained

Accuracy and confusion matrix calculated and is shown below



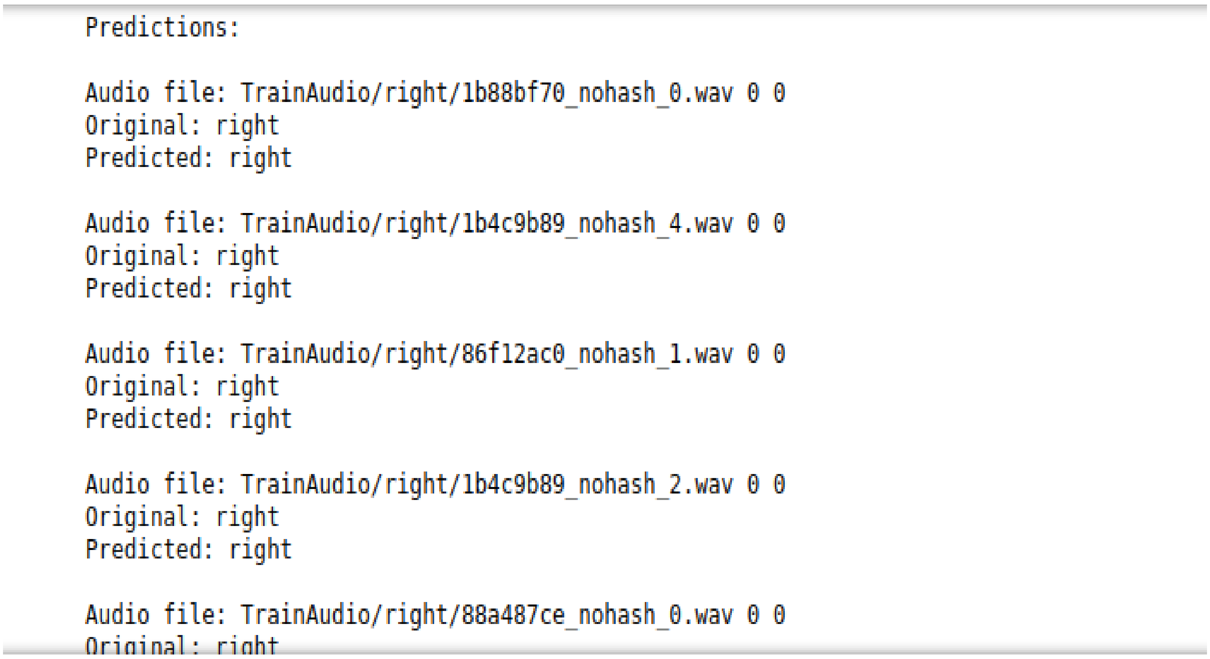
```
              precision    recall  f1-score   support

       right       0.86      0.95      0.90        20
        left       0.78      0.90      0.84        20
          no       0.94      0.80      0.86        20
         yes       1.00      0.90      0.95        20

    accuracy                           0.89        80
   macro avg       0.90      0.89      0.89        80
weighted avg       0.90      0.89      0.89        80
```

Figure 6: Results with Confusion Matrix

Speech recognised for the test input audio ‘left’ is below. ‘left ’ has 50% accuracy of correctly recognised as per confusion matrix.



```
Predictions:

Audio file: TrainAudio/right/1b88bf70_nohash_0.wav 0 0
Original: right
Predicted: right

Audio file: TrainAudio/right/1b4c9b89_nohash_4.wav 0 0
Original: right
Predicted: right

Audio file: TrainAudio/right/86f12ac0_nohash_1.wav 0 0
Original: right
Predicted: right

Audio file: TrainAudio/right/1b4c9b89_nohash_2.wav 0 0
Original: right
Predicted: right

Audio file: TrainAudio/right/88a487ce_nohash_0.wav 0 0
Original: right
```

Figure 7: Test Output for the audio sample ‘left’

5. Challenges faced

At first there was confusion related to the template models to implement the ASR and we chose the popular HMM model due to major implementation of HMM models for the accuracy.

We had the trouble in mfcc feature extraction and implemented the peak detection method to learn about the HMM model prediction and utilized the knowledge gained in the peak detection method for the mfcc feature extraction method and implemented the ASR using the MFCC feature and HMM model.

6. Lessons learnt

- We used the peak detection as the speech feature that resulted in the poor accuracy.
- Also, Peak detection works for single person speech.
- During the peak detection method, we learnt about the HMM model and using the speech features in HMM to predict the speech.
- We learned about using the libraries related to speech and its implementation methodologies.

7. References

Dataset:
https://www.kaggle.com/c/tensorflow-speech-recognition-challenge/data

Program Codes:
https://github.com/UNREALre/SpeechRecognitionHMM_Basics
https://www.kaggle.com/ilyamich/mfcc-implementation-and-tutorial

Reports:
1. **SPEECH TO TEXT CONVERTER USING GAUSSIAN MIXTURE MODEL(GMM)**, Virendra Chauhan 1 , Shobhana Dwivedi 2 , Pooja Karale 3 , Prof. S.M. Potdar 4
1,2,3 B.E Student
4 Assitant Professor of Electronics and Telecommunication Engineering
1,2,3,4 Sinhgad academy of Engineering, Pune 443001

2. **Speech Recognition based on Template Matching and Phone Posterior Probabilities**
a Cédric Gaudard
b Guillermo Aradilla
b Hervé Bourlard
IDIAP–Com 07-02
a EPFL student, MSc. internship performed at IDIAP
b IDIAP Research Institute

*3. **Gaussian Mixture Models***
Douglas Reynolds
MIT Lincoln Laboratory, 244 Wood St., Lexington, MA 02140, USA

Online references:
1. https://medium.com/@kangeugine/hidden-markov-model-7681c22f5b9
2. https://jonathan-hui.medium.com/speech-recognition-gmm-hmm-8bb5eff8b196
3. https://jonathan-hui.medium.com/speech-recognition-feature-extraction-mfcc-plp-5455f5a69dd9
4. https://jonathan-hui.medium.com/speech-recognition-acoustic-lexicon-language-model-aacac0462639