Telugu Speech Sentence Modeling For controlling smart devices

A Project Report Submitted
in Partial Fulfillment of the Requirements for the award of the Degree

Bachelor of Technology

in

Computer Science and Engineering

by

V Lasya	19761A05I7
P Dhyeya	19761A05H4
M Anusha	20765A0515

under the guidance of

Dr. P. Bhagath M. Tech (IITG), Ph. D (IITG)

Associate Professor

Department of Computer Science and Engineering

Lakireddy Bali Reddy College of Engineering (Autonomous)

Approved by AICTE, New Delhi & Affiliated to JNTUK, Kakinada
Accredited by NAAC with 'A' Grade & NBA(Under Tier-1) and ISO 9001:2015 Certified Institution
LBReddy Nagar, Mylavaram, NTR District, Andhra Pradesh, India-521230

2019 – 2023

Lakireddy BaliReddy College of Engineering (Autonomous)

Approved by AICTE, New Delhi & Affiliated to JNTUK, Kakinada
Accredited by NAAC with 'A' Grade & NBA(Under Tier-1) and ISO 9001:2015 Certified Institution
LBReddy Nagar, Mylavaram, NTR District, Andhra Pradesh, India-521230
2019 – 2023

Department of Computer Science and Engineering

CERTIFICATE

This is to certify that the project entitled "**Telugu Speech Sentence Modeling For Controlling Smart Devices**" is a bonafide work of

V Lasya	19761A05I7
P Dhyeya	19761A05H4
M Anusha	20765A0515

in partial fulfillment of the requirements for the award of degree of **B.Tech.** in **Computer Science Engineering** from **Jawaharlal Nehru Technological University Kakinada** is a record of bonafide work carried out by them at **Lakireddy Bali Reddy College of Engineering(A).**

Supervisor
Dr. P. Bhagath

Head of the Department
Dr. D. Veeraiah

EXTERNAL EXAMINER

Declaration

We are here by declaring that the project entitled "**Telugu Speech Sentence Modeling For Controlling Smart Devices**" work done by us. We declare that the work contained in the report is original and has been done by me under the guidance of supervisor. The work has not been submitted to any other institute in preparing for any degree or diploma. We have followed the guidelines provided by the institute in preparing the report. We have conformed to the norms and guidelines given in the Ethical Code of Conduct of the institute. Whenever we have used materials (data, theoretical analysis, figures and text) from other sources, we have given due credit to them by citing them in the text of the report and giving their details in the references. Further, we have taken permission from the copyrights owner of the sources, whenever necessary.

Name of the Student	Signature of the student
V Lasya	19761A05I7
P Dhyeya	19761A05H4
M Anusha	20765A0515

Acknowledgements

We take great pleasure to express our deep sense of gratitude to our project guide **Dr. P. Bhagath**, Associate Professor, for his valuable guidance during the course of our project work. We would like to thank **Dr. D. Veeraiah**, Professor & Head of the Department of Computer Science & Engineering for his encouragement. We would like to express our heart-felt thanks to **Dr. K. Appa Rao**, Principal, Lakireddy Bali Reddy College of Engineering for providing all the facilities for our project. Our utmost thanks to all the faculty members and non teaching Staff of the Department of Computer Science & Engineering for their support throughout our project work. Our Family Members and Friends receive our deepest gratitude and love for their support throughout our academic year.

V Lasya	19761A05I7
P Dhyeya	19761A05H4
M Anusha	20765A0515

Abstract

Speech recognition is fast growing and an active research field in which spoken words or sentences are processed to understand the meaning. It continues to this day as a major area of research having commercial importance as with a number of applications. One of the significant applications is designing spoken dialog systems whose task is to interpret the message in the speech signal. Home appliance management is an interactive environment where there is a scope to develop speech interfaces for the operation of the appliances. Smart home control systems help disabled people to control their home electrical devices such as television(TV), lights, and fans using only voice commands without moving to turn on or turn off electrical equipment. Even though commercial systems are available, they are to be customized to suit the needs of a specific language. In this project the problem is handled in the context of the Telugu language. Since the dataset is not readily avialable, we prepared a sentence dataset for Telugu language for controling the IoT devices. Hidden Markov Modeling(HMM) with the use of different feature extraction techniques such as LPC, LPCC, and MFCC. In the study we find that MFCC features, work well comparitively for sentence modeling .The complete procedure of dataset collection, modeling, and evaluation are discussed in the report.

Contents

ListofFigures

ListofTables

Chapter 1

Introduction

Nowadays, appliances and other items are updated gradually along with technology. Home appliances like fans, bulbs, and lights that turn on or off automatically when a person enters a room were first introduced to us. Although this is a great tool, it can be difficult for children, the elderly people, and physically challenged persons to move form one place to other in home. Automatic voice recognition can assist them avoid this difficulty by allowing them to utilise gadgets simply by speaking. We use Hidden Markov Model (HMM) to classify this audio clip and output the specific sentence that was spoken. In this project we use Telugu Language as it is a low resource language and the development on local language instead of other language telugu which is very low resource language. so in this project we use telugu language instead of other regional languages. Telugu is spoken by 81 million people and it is the third most spoken Indian language by the native speakers. Telugu ranks fifteenth on the list of most-spoken languages worldwide and there are many websites and blogs in Telugu [1].

Automatic speech recognition(ASR) is a technology that converts the phrases or words spoken by humans into text. Although the technology has been developed for years, the accuracy of ASR is still a major issue that prohibits it form being used as the primary input method for the users. The accuracy of ASR depends heavily on the core technology adopted by the recognition engine, which can be classified into three types based on the dependence of the speaker, with different backend Hidden Markov models (HMM). Some speech recognition systems require ”training” (also called ”enrollment”) where an individual speaker reads text or isolated vocabulary into the system. The system analyzes the person's specific voice and uses it to fine-tune the recognition of that person's speech, resulting in increased accuracy. Systems that do not use training are called ”Speaker-independent” systems. Systems that use training are called ”speaker-dependent”.

The proposed system provides speech recognition in telugu language spoken by isolated and Impaired people in different ways [2] . The input is taken by an ASR that is internally used as a speech recognizer to identify the command. The project is mainly used for the elderly and

physically challenged people to operate household items without having trouble [3] . This device is very much useful who are in need. The main use of such a module is to provide an easy operation to switch on/off of home appliances. The speech recognition system is developed as an embedded module.Therefore, the internet is not required. In the device, we have Automatic speech recognition and both are connected. Speech recognition system is done in the Telugu language. Speech recognition is the process by which a computer or any other device identifies the spoken words. It means talking to your computer and having it correctly recognize what you are saying. They are several ways to do it but the basic principle is to extract certain key features from the uttered speech and then treat those features as the key to recognizing the word when it is uttered again. The processing has many facets, for example, distinguishing different utterances, speaker identification, etc. Different characteristics of speech can be used to identify the spoken words, the gender of the speaker, and the identity of the speaker. Two important features of speech are pitch contaminant frequencies. Pitch is a significant distinguishing factor between make and female speakers. The frequency of vibration of vocal folds determines the pitch, for example, 300 times per second oscillation of the folds results in a 300 Hz pitch. Harmonics (integer multiples of a fundamental frequency) are also created while the air passes through the folds. Age also affects the pitch. Just before puberty, the pitch is around 250Hz. For adult males, the average pitch is 60 to 120 Hz, and for females, it is 120 to 200 Hz [4] .

The vocal tract, consisting of the oral cavity, velum, epiglottis, and tongue, modulates the pitch harmonics created by the pitch generator. The modulations depend on the diameter and length of the cavities. These reverberations are called contaminant frequencies(or resonances). The harmonics closer to the contaminant frequencies get amplified, while others are attenuated [5] .

While humans are speaking, the contaminants vary depending on the positions of the tongue, jaw, velum, and other parts of the vocal tract[6]. Two related key factors are the bandwidth of each contaminant, and contaminant membership in a known bandwidth. The vowels for all human beings tend to be similar. Each vowel uttered by a person generates different formats. So we can say that the vocal tract is a variable filter whose inputs are the pitch, and the pitch harmonics. The output of the filter is the gain or the harmonics falling in different contaminant frequencies. The filter is called the variable filter model. The transfer function for the filters determined by the contaminant frequencies. Although the pitch can be easily varied by a speaker, the pitch ca be easily filtered for any electrical noise, by using a low pass filter. Contaminants on the other hand are unique for different speakers and are useful in individual speaker identification. In general, a combination of pitch and contaminant can be used in a speech recognition system. The voice of individual changes over time due to environmental, contextual, and variability in the speaker's voice. The limited features-based training approaches may not provide the desired level of performance under various conditions. It may lead to the scenario of false rejection even with a minor variation in the speech utterances by the speaker. The accuracy of the system is also dependent on

several other factors like the control over recording equipment, recording conditions, cooperation form the speaker, etc. In this work, a speaker verification model is presented addressing the long-term speaker variability issue. A combination of time, frequency, and spectral-based components are used for feature extraction from the sample voices [7] .

Chapter 2

Review of Prior Works

2.1 Speech Recognition in Telugu Language

Speech processing is a growing area of research that involves understanding the meaning of spoken words or sentences. Designing spoken-dialog systems whose job it is to decipher speech signals is one of the important applications. In the Internet of Things, spoken communication with connected devices can be useful [8] . The development of speech interfaces for appliance operation is possible in the interactive environment of home appliance management. Given how difficult it is to develop an automated system, spoken interfaces for controlling home appliances must be specifically tailored to a given language. Commercial systems are readily available, but they must be modified to meet the requirements of a particular language. In this paper, a Telugu-language spoken dialogue system for controlling home appliances is suggested. A mobile device' speech processing functionality is built in to make it useful in unreliable network environments. The paper discusses the design and implementation specifics.

Speech analysis is an active area of study where various feature extraction methods are investigated to address various problems. Through an understanding of the necessary cues to select the features, such studies assist in reducing the time complexity of solutions. A crucial step in feature engineering is selecting the most important features by eliminating irrelevant data. Perceptual Linear Predictive (PLP) modelling focuses on the features perceived at the listener end in order to understand the speech signals. Many speech processing applications have used them successfully. The recognition task heavily relies on the choice of the PLP coefficient order for effective classification of spoken units. To create an automatic speech recognition system, a traditional speech processing system must undergo a lengthy training process. Such systems are efficient for the languages that have enough resources i. e . data. But, low-resource languages especially Asian languages haven't been developed to provide the data sufficient for such tasks. In this context, alternative methods and techniques are encouraged to enhance or optimize the development process with less amount of data. This paper proposes a pre-clustering technique to improve the classification rate with low resources.

Automatic Speech Recognition (ASR) is the process of identifying spoken words or sentences

using an algorithm or automated programme. ASR has a wide range of uses in fields like mobile speech recognition, the internet of things, human-machine interaction, etc. [9] For more than 50 years, researchers have been focusing on various issues in these fields. Mobile speech recognition is a dominant field that supports a wide range of applications beneficial for people with physical disabilities, elderly people, and inexperienced mobile users. Despite the domain's considerable practicality, there are difficulties because there is a dearth of information in the intended language.

Although it is not significant for highly developed languages like English, it has a significant impact on underdeveloped and low-resourced languages. Hindi is the most common language spoken in India, which has a population of 1,391.99 million people. Every region of the country has its own native language. However, there are a number of reasons why research that advances ASR is dubious, and the lack of data is just one of them. Due to this, an open-source speech digit data set for Telugu, the sixth most widely spoken language in the nation, has been created and will be made available through this paper.

Home automation is a fascinating field of study that focuses on creating new processes and systems that make technology accessible to the general public.

Physically disabled people are becoming more and more dependent on environments with hands-free operation. This specialised field, where speech interfaces are developed for using various devices, is particularly benefiting from the speech processing domain. The development of such systems poses new difficulties when they are intended to support low-resource languages [10] . Despite the fact that these languages are actually needed, they have made relatively few advances. India is a multicultural nation where many languages are spoken. Making spoken dialogue systems for Indian languages is extremely difficult because there aren't many speech corpora available. In this paper, we propose a speech recognition system and implement it for a home automation system for two spoken languages.

Speech processing is an active area of study that involves understanding the meaning of spoken words or sentences. Designing spoken dialogue systems whose job it is to decipher the message contained in the speech signal is one of the important applications. In the Internet of Things, spoken dialogue can be useful for interacting with connected devices. The development of speech interfaces for controlling appliances is possible in the interactive environment of home appliance management. Because it is difficult to create a generic system, the spoken interfaces for controlling home appliances must be tailored to a specific language. Commercial systems are readily available, but they must be modified to meet the requirements of a particular language. In this paper, a Telugu-language spoken dialogue system for controlling home appliances is suggested. A mobile device's embedded speech processing functionality makes it usable in unreliable network environments. The paper discusses the design and implementation specifics.

The voice recognition method involves the system receiving a speech signal from the user or client through a microphone, analysing the signal, and extracting usable information that is then transformed into text. With a raspberry pi at its core, a speech recognition system that is coupled with the internet of things is designed and put into use to operate home furnishings and electrical

equipment [11] . We use digital signal processing techniques to create this speech recognition system. Hidden Markov Models and digital processing methods are employed. These methods are fully taken into account for the system's processing, extraction, and high predicted accuracy. Using the Google application programming interface was used as a cloud server to store command and give the access to the internet. This research has used the system to analyse 150 speech samples, and they have achieved an accuracy level of 80

Neural network-based speech recognition is used for the majority of intelligent Internet of Things (IoT) goods at the moment. The typical interface for human-machine interaction. The typical speech recognition frameworks for IoT smart devices, however, always collect and transmit voice data in plaintext, which may result in the disclosure of user privacy. The widespread use of voice features for biometric authentication makes privacy leaks extremely damaging to both individual property and personal freedom [12] . Therefore, we suggest an edge computing and long short-term memory (LSTM) neural network-based outsourced privacy-preserving speech recognition framework (OPSR) for smart IoT devices in this research. The system aims to deliver lightweight outsourced compute through a set of additive secret sharing-based interaction protocols between two edge servers. Additionally, we develop the LSTM neural network training procedure for intelligent IoT device voice control based on the protocols. Finally, we theoretically demonstrate the accuracy and security of our framework using the universal composability theory in combination with the outcomes of our experiments – Zhuo Ma et al.

Some research initiatives have already concentrated on developing a network of effective voice recognition for the growth of edge computing, driven by the vision of the Internet of Things. Other studies (like tpool2) do not fully exploit the spatial and temporal information contained in the speech acoustic properties. In this paper, we introduce EdgeRNN, an edge computing-focused compact voice recognition network using spatiotemporal features. As an alternative, EdgeRNN processes the overall spatial information of each frequency domain of the acoustic characteristics using a 1-Dimensional Convolutional Neural Network (1-D CNN). The temporal data of each frequency domain of the acoustic characteristics is processed by a recurrent neural network (RNN) [13] . Additionally, we suggest a condensed attention strategy to improve the network component. EdgeRNN's overall effectiveness has been confirmed in terms of speech emotion and keyword recognition. Speech emotion recognition uses the IEMOCAP dataset, and the unweighted average recall (UAR) is 63.98 percent With a weighted average recall (WAR) of 96.82 percent, Google's Speech Commands Datasets V1 are used for speech keyword recognition. The accuracy of EdgeRNN has increased for both speech emotion and keyword detection when compared to experimental results of similar efficient networks on Raspberry Pi 3B+ - S Yang et al.

Speech recognition technology is being used in a variety of fields, including robotics, healthcare, vehicle control, and unmanned aerial vehicle systems. Many voice recognition systems have been created in recent years to address a variety of real-world application problems Utilizing windowing and framing techniques along with an improved mel frequency cepstral coefficient, we have suggested a revolutionary speech recognition system. The input speech signal's Gaussian white

noise is eliminated using the windowing and framing method. The nonnegative matrix factorization approach is successfully used by the de-noising block to factorise the Mel-magnitude spectra of the noisy input audio signal. Additionally, the mel-frequency cepstral coefficients (MFCC) are employed to identify the speech signal's more crucial aspects. In order to recognise the audio signals, the Laplace smoothing approach is used as the language model. The proposed Mel frequency cepstral coefficient with Windowing and Framing based voice recognition system is demonstrated using MATLAB software. The suggested voice recognition system has been put up against artificial neural network- and wavelet-based feature extraction techniques for speech recognition. The results of the experiments demonstrated the proposed Mel frequency cepstral coefficient's successful use with a windowing and framing-based speech recognition system – S Lokesh et al.

Recent research in speech recognition technology has focused on how robust speech recognition systems are against linguistic variance. The first step in creating proper speech recognition technology for everyone is to create a system that can converse with humans in any language like any other human. With a population of more than a billion people, India has a very diverse linguistic community. Consequently, it offers a reliable field of study for language-specific speech recognition technology. Due to its capacity to represent the temporal features of speech and encode them as a series of spectral vectors, the hidden Markov model (HMM) methodology has dominated the field of technology. HMM technique is also used extensively in work done in Indian languages. But in the last 10 to 15 years, as neuro computing has gained acceptance as a viable alternative to HMM, artificial neural network (ANN)-based approaches have begun to draw interest for use in voice recognition. This is a global trend, and several researches have reported a few works as a part of it – Mousmita Sharma et al.

The Automatic Speech Recognition (ASR) systems for Southeast Asian languages are presented in this study as a whole. As there hasn't been much research done on these languages, there are a few issues that need to be resolved before developing the systems: limited speech and text resources, lack of linguistic expertise, etc. This study uses Bahasa Indonesia and Thai as examples to show how to gather the various materials needed to develop ASR systems – Lei Wang et al.

A particularly active and difficult study area is natural language and human-machine interaction. However, the major goal is to develop a system that can effectively connect with humans, regardless of operating setting. The Automatic Speech Recognition (ASR) for tonal languages spoken worldwide is systematically surveyed in this research. Tonal languages of the American and Austral-Asian continents are not reviewed, only those of the Asian, Indo-European, and African continents are. The most significant portion of this paper is the presentation of previous research on the ASR of Indo-European continent tonal languages like Punjabi, Lithuanian, Swedish, Croatian, and Yoruba, as well as African continent tonal languages like Hausa and Yoruba. These languages include Chinese, Thai, Vietnamese, Mandarin, Mizo, and Bodo. On the basis of the findings, the synthesis analysis is then investigated. Tonal languages are examined along with several problems and difficulties. It has been noted that while much work has been done for the tonal languages of the Asian continent, such as Chinese, Thai, Vietnamese, and Mandarin, little has been reported for

the tonal languages of the Mizo, Bodo, and Indo-European families, such as Punjabi, Latvian, and Lithuanian, as well as for the tonal languages of the African continent, such as Hausa – Jaspreet Kaur et al.

A speech recognition-based smart classroom management system has been designed and developed to raise the amount of information in smart classrooms in light of the expansion of teaching scale and the quick growth of educational material. The application of multimedia equipment control is covered in this study. For speech recognition, the system uses a local speech database and the advanced campus network in the cloud. The system's implementation demonstrates that it has more benefits than the conventional multimedia classroom management system and is infinitely expandable. The smart classroom management system is based on speech recognition in cloud architecture [14] . It makes unified management of the school easier, boosts administrators' efficacy, conserves a lot of human and financial resources, and significantly fosters the growth of school information construction. A smart learning scenario for enhancing the learning and teaching scenario, the smart classroom represents a new direction in the application of information technology in the field of teaching and learning. This study focuses on the use of smart classroom as an additional teaching tool to enhance Chinese language teaching and learning, and it provides examples of how smart classroom can optimise the learning environment for vocabulary development and listening and speaking practise - Lijun Hu et al.

The development of a speaker-independent LVCSR engine for Mandarin Chinese using our multilingual database Global Phone is presented in this study. We outline a two-pass method in which recognition first creates Pinyin hypotheses, which are then converted into Chinese character hypotheses. We demonstrate how this strategy can simplify things while enhancing flexibility. We assess and contrast several systems, including various voice recognition base units such as phoneme units as opposed to syllables. We also examine the impact of tone information. Our current best system achieves a 15.0 percent character mistake rate, which is quite encouraging – J. Reichert et al.

The goal is to develop a voice and speech recognition system for smartphones that can recognise voices, record Tamil speech, and then store and transform that text into Tamil. The captured message is conveyed to the receiver in Tamil while using voice calling or sending SMS by speaking the message aloud. The use of Tamil in voice and speech recognition on smartphones has not received much attention. Tamil voice recognition and speech would give native users more options in their smartphone experience [15] . Additionally, Tamil-only speakers would find it simpler to use the speech recognition feature on their smartphones if it were made available. For local users, using phones won't be any more challenging, and learning how to use a smartphone is not necessary. The success of automatic speech recognition (ASR) systems in various applications is very high. The most widely used of these include Machine Learning techniques like Neural Networks (NN), Support Vector Machine (SVM), and Decision Trees (DT), as well as Template Matching techniques like Dynamic Time Warping (DTW), Statistical Pattern Matching techniques like Hidden Markov Model (HMM), and Gaussian Mixture Models (GMM). The HMM approach allows for the highest

word recognition accuracy in this system. During the training process, it delivered 100 percent accuracy, and during the testing process, it offered about 98 percent - Kiran R et al.

A technology called automatic voice recognition is used to convert spoken language into written language. It is utilised in a number of contexts, voice commands, customer service, and more, for instance. In the process of digitising daily life, it has grown to become a crucial instrument. It is well known to play a crucial role in significantly easing the lives of elderly and disabled individuals. In this study, we present an automatic speech recognition model that was refined from the Facebook XLSR Wav2Vec2 model, which was trained on the Common Voice Dataset, utilising three pretrained models. The word mistake rate of the data is used to choose the optimal voice recognition model for Tamil. This paper discusses SSNCSE NLP's submission to the joint task run by LT-EDI at ACL 2022. The achieved word error rate is 39.4512 – Dhanya Srinivasan et al.

This study examines the use of a machine-learned model for word recognition in the Urdu language. For modelling, speech snippets from a variety of speakers were used. The discrete Fourier transformation is applied to the initial time-domain samples after normalisation and pre-processing in order to extract the speech features. For the same words uttered by various speakers, a high degree of correlation in the frequency domain was discovered. High recognition accuracy models were created as a result of this. This paper includes information on model realisation in MATLAB. Utilizing linear predictive coding for effective hardware implementation, current work is being extended – Azam Beg et al.

This study uses the CMU Sphinx Open Source Toolkit for speech recognition to develop acoustic and linguistic models for reliable Urdu speech recognition. Up to two speakers' worth of speech data from each pass have been added incrementally to three models: one using data from 40 female speakers only, one from 41 male speakers only, and one using data from both male and female speakers (81 speakers). The current recognition results are presented in this paper along with methods for raising these recognition rates - Huda Sarfraz et al.

The task of speech recognition for the Urdu language is intriguing and underdeveloped. This is mainly because Urdu does not have access to linguistic resources like rich corpora. However, there have been few attempts to create Urdu speech recognition frameworks using conventional methods like Hidden Markov Models and Neural Networks. In this work, we examine the application of three classification techniques for the task of recognising Urdu speech. In order to perform Urdu speech recognition, the classifiers are trained using delta and delta-delta features from the speech data. We demonstrate the results of training the Support Vector Machine (SVM) classifier, the Random Forest (RF) classifier, and the Linear Discriminant Analysis (LDA) classifier in order to compare their performance to that of the SVM. As a result, the experimental findings demonstrate that SVM performs better than RF and LDA classifiers on this particular task – Wageesha Manamperi et al.

At the IBM Italy Scientific Center in Rome, a system for automatic speech recognition of the Italian language has been created. It can recognise natural language sentences that are composed of words from a 6500 items dictionary and are spoken aloud by a speaker with brief pauses in between.

The system is speaker-dependent; in order to use it, the speaker must complete a training phase that lasts 15-20 minutes and involves reading a predefined text. It is powered by an architecture made up of a mainframe IBM 3090 and a workstation PC/AT equipped with signal processing tools – Paolo D'Orto et al.

Arabic automatic speech recognition is a very difficult task. Despite the fact that Arabic speech recognition can be successfully performed using all of the conventional Automatic Speech Recognition (ASR) techniques, it is crucial to take into account the unique characteristics of the language. Speech recognition for Modern Standard Arabic (MSA) is the main topic of this article. We discuss the difficulties that the Arabic language faces, including its difficult morphology and the lack of short vowels in written text. As a result, each grapheme has a number of potential vowelizations that are frequently incompatible. Using the Kaldi toolkit, we create an ASR system for MSA. Many linguistic and acoustic models are trained – Mohamed amine menacer et al.

The main initiatives in large vocabulary, continuous speech recognition (LVCSR) for European languages are presented in a basic overview. We discuss issues with comparative evaluation, lexical representation, language modelling, and acoustic modelling for a number of European languages. By resourcing the lattice and rewriting the output with the correct grammar, we achieve a Word Error Rate (WER) of 14.42 for the baseline system and 12.2 relative improvement.

Automatic Speech Recognition is a method for identifying human speech patterns, leading to the production of pertinent text [16] . There are many interactive software options on the market, but because not all languages are supported, their use is constrained by vernacular languages. We have examined the Gujarati Speech Recognition System in this paper using a variety of speech recognition techniques that will meet the needs of Gujarati users and aid in their technological advancement – Deepang Raval et al.

Speech recognition for Punjabi is urgently needed because it is one of the languages used in media and communication the most. As a result, a survey on Punjabi speech recognition has been conducted. Here, work has been done on everything from boundary detection for isolated word recognition to present-day scenarios. This has only dealt with limitations and presumptions, though. This essay discusses current efforts and upcoming difficulties – Muskan Garg et al.

The Punjabi language is widely spoken there as well as in some parts of Rajasthan and Haryana. Many speech recognition systems exist for the Punjabi language, but they all have issues that must be resolved for better performance. The purpose of this essay is to discuss various other speech recognition techniques that have been created for single words, groups of words, and in other languages. Additionally, it goes over the procedures used and the accuracy attained – Ravinder Singh et al.

Speech recognition, also known as speech to text processing, is the process by which a computer recognises human speech and converts it into text. To create transcripts for speech recognition, recordings of transcriptions of spoken word in text and as audio. Research is ongoing in the field of speech-based applications that use Natural Language Processing (NLP) methods. Such applications accept natural language input and produce natural language results. Three approaches, namely

the Acoustic phonetic approach, the Pattern recognition approach, and the Artificial intelligence approach, are primarily used in speech recognition. It takes a sizable database of speech and training algorithms to create an acoustic model. An ASR system's output is the recognition and conversion of spoken language into text by computers and other electronic devices. ASR is now widely used for tasks like voice dialling and others that call for human-machine interfaces. Our primary contribution to this paper is the creation of Marathi language corpora and the investigation of the application of the Sphinx engine for automatic speech recognition – Aman Ankit et al.

Speech is the primary form of human communication. The development of automatic speech recognition of Marathi numerals, from Shunya (Zero) to Nau, is discussed in this paper (Nine). The technique of feature extraction and feature matching is crucial for speech recognition. We used the dynamic time wrapping (DTW) technique for feature matching and the Mel Frequency cepstral coefficients (MFCC) technique for feature extraction. The extracted feature's data are minimised using vector quantization. With the aid of high-performance Headsets and PRAAT software for data recording, the total database collection is approximately 100 speakers. Data is separated into speakers who are male and female. It was recorded in a noisy setting. Then feature extraction and feature matching techniques are used after the noise removal technique to address the DTW technique is employed due to the varied speaking rate – Ratnadeep R. Deshmukh et al.

Speech has long been the most instinctive form of human communication. As a result, since the development of computer science and signal processing, the field of automatic speech recognition has undergone tremendous advancement (ASR). In actuality, it has been a focus of active research for more than 50 years. Regardless of the semantics of the words or sentences, ASR converts audio speech signals to corresponding text transcriptions. This method is thought to revolutionise both human-human and human-machine communications – Ilhem Isra Mekki .

The description of a speech recognition system for continuously spoken Japanese simple sentences. The acoustic analyzer can represent the speech sound by a phoneme string in an expanded sense that contains acoustic features like buzz and silence in addition to regular phonemes. This method is based on a psychological assumption for phoneme identification. The reference phoneme string and the reference characteristic phoneme string required for matching the input phoneme sequences are taken from the word dictionary using a translating routine. Each entry in the word dictionary is written in Roman letters using the Hepburn system. When analysing syntax, inflexions of verbs, adjectives, and some key auxiliary verbs are taken into consideration. Utilizing a network that deals with state transitions between speech parts, the syntax analyzer predicts subsequent words and outputs their syntactic interpretation of the input phoneme string. The semantic knowledge system creates a semantic network by addressing the semantic definition of each verb, the semantic nature of each word, and the sentence structure. The semantic analyzer checks the recognised sentence's semantic validity to see if each word complies with the verb's definition or those of other words. The syntax analyzer uses a network that deals with state changes between parts of speech in order to forecast subsequent words and output their syntactic interpretation of the input phoneme string. The semantic knowledge system builds a semantic network by taking

into account the structure of the sentence, the semantics of each word, and the semantics of each verb. The semantic analyzer determines whether each word in the recognised sentence satisfies the definition of the recognised verb or those of other words by evaluating the semantic validity of the sentence – Minoru Shigenag et al.

In addition to introducing three research projects carried out at NTT's Human Interface Laboratories and ATR Interpreting Telephony Research Laboratories, this paper provides an overview of the speech recognition problems for the Japanese language. The first topic is stochastic language models for Japanese character sequences that can be used in an unrestricted vocabulary Japanese dictation system. The second topic is an accurate and effective HMM-LR-based continuous speech recognition algorithm for very large vocabulary. This algorithm was used to recognise spontaneous speech with a vocabulary size of roughly 80,OOO in a telephone directory assistance system. The third topic is a continuous speech recognition system built on methods of (allophonic) modelling and parsing that depend on phoneme context – S Furui.

In order to facilitate natural cross-language communication, ATR is currently developing a next-generation speech translation system. Further research efforts should focus on the robustness for large vocabulary, speaking variations frequently found in fast spontaneous speech, and speaker variances to cope with the various requirements to speech recognition technology for the new system. These are important issues that need to be resolved for both speech translation and the general application of speech recognition in realistic settings. This paper reports the current state of data collection and designs three sizable speech databases to address these speech recognition issues – A. Nakamura et al.

The research area of Russian speech recognition is significant. This article introduces a fundamental HMM recognition system and examines the use of HMM in this system, starting with the fundamentals of speech recognition. It highlights the importance of Russian text and speech corpora as a guarantee for high recognition rates. We discover some methods for further improvement through comparison and experimentation, with the optimization of the recognition algorithm and its integration with Russian being the main areas of future study – Yanzhou Ma et al.

The paper looks at the practical difficulties in creating a deep neural network-based speech-to-text system. The creation of a Deep Speech-based speech recognition system for the Russian language is described. The Deep Speech English opensource implementation developed by the Mozilla company served as the basis for further development. Using Docker technology, the system was trained in a containerized environment. It made it possible to describe every step of component assembly from scratch, including various CPU and GPU optimization methods. Additionally, Docker makes it simple to replicate computation optimization tests on different infrastructures. We investigated the use of TensorFlow XLA technology, which speeds up computations involving linear algebra during neural network training. Based on the word error rate (WER) obtained on a test data set and taking GPU memory constraints into consideration, the number of nodes in the internal layers of the neural network were optimised. We investigated the application of probabilistic language models with different maximum word lengths and chose the model that exhibits the best

WER. A Russian-language acoustic model was trained as a result of our study using a data set that included audio and subtitles from YouTube video clips. The popular Russian-language articles from Wikipedia and the texts of their subtitles served as the foundation for the language model's construction. The resulting system was tested on a dataset of audiobooks of Russian literature from voxforge.com; the best WER displayed by the system was on this dataset – Oleg Lakushkin et al.

We provide a review of the most recent advancements in Russian speech recognition research in this paper. Despite the Although the underlying speech recognition technology is largely language-neutral, differences in language structure and grammar have a significant impact on the system's performance. The word formation system in Russian is intricate and is characterised by a high level of inflection and flexibility in word order. Due to this, the predictive power of the traditional language models is drastically reduced, which raises the error rate. A significant amount of speech and text data are needed for the current statistical approach to speech recognition. Numerous Russian speech databases exist, and this paper provides descriptions of them. In addition, we discuss and contrast a number of speech recognition systems created in both Russia and other nations. Finally, we make some encouraging recommendations for future studies in Russian speech technology – Daria Vazhenina et al.

A data-driven technology, speech processing depends on public corpora and related resources. Brazilian Portuguese has few resources compared to languages like English (BP). This work describes initiatives to close this gap and presents speech recognition systems in BP that use the public corpora Spoltech and OGI-22. ATK and HTK scripts, a pronunciation dictionary, language and acoustic models, and other materials are made available. The baseline outcomes attained with these resources are discussed in the work – Patrick Silva et al.

While there are numerous publicly available resources for some languages (such as English and Japanese), the resources for Brazilian Portuguese (BP) are still scarce. An automatic speech recognition system has modules that depend on the language. This article discusses the creation of free resources and tools for BP speech recognition, including language and acoustic models, phonetic dictionaries, grapheme-to-phone converters, and text and audio corpora. All of them are freely accessible, and they have been combined with an envisioned application programming interface to create a number of new applications, including a speech module for the OpenOffice software. Performance tests that contrast the developed BP system with a piece of commercial software are presented. In addition, a programme using synthesis, speech recognition, and a natural language processing module specifically designed for statistical machine translation are described in the paper. With the help of this application, spoken conversations can be translated from BP to English and vice versa. The resources facilitate the industry's and other academic institutions' adoption of BP speech technologies – Nelson Neto et al.

A multilingual environment is necessary for the Verbmobil speech-to-speech translation system. This consists of language identification components that can switch between these recognizers and recognition engines for the three languages of German, English, and Japanese that run in a single

shared framework [17]. The difficulties of multilingual speech recognition are discussed, along with various solutions to the issue of the automatic language identification task. The aforementioned elements come together to create a flexible and approachable multilingual spoken dialogue system - Alex Waibel et al.

2.2 Related Work

2.2.1 Hidden Markov Models

Understanding the key properties of voice utterances represented as waveforms is the first step. LPCC (Linear Predictive Cepstral Coefficients) and MFCC (Mel-Frequency Cepstral Coefficients) are two examples of characteristics. The cepstral components are recorded at various levels by MFCC . The most successful features are MFCCs . Each speech utility is extracted as a feature in the feature extraction procedure. Each speech utterance is broken into a number of smaller pieces called frames with a fixed number of samples during the feature extraction procedure. The frame size for these frames is usually set to 20 milliseconds. The rationale for the frame size has been investigated in many ways by researchers , and it has been picked as the best option for short-term window processing. The window has a defined size. The discrete cosine transform is then applied to the fixed-sized window to get the final MFCC features. For each 20 ms window, the processed features set has 13 features [1].

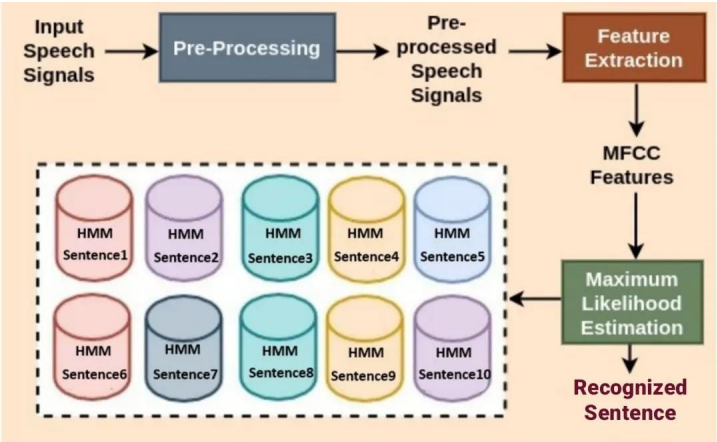


Figure 2.1 HMM training

In most of the cases, real-world processes result in observable outputs which are referred to as signals. The signals can be discrete (e.g., characters from a finite alphabet, quantized vectors from a codebook, etc.) or continuous (e.g., characters from a finite alphabet, etc). (e.g., speech samples, temperature measurements, music, etc.). The signal source can be stationary (that is, its static qualities do not change over time) .

The HMM framework is used to model the features, which involves describing the phonetic units of a word as states in a stochastic process. A set of parameters can be used to represent an HMM

system. Both deterministic and stochastic signal models have performed well in the applications of interest, especially speech processing. We will concentrate on one sort of stochastic signal model in this study, namely the hidden Markov model (HMM) . In the communications literature, these models are referred to as Markov sources or probabilistic functions of Markov chains.

For tasks like clustering, feature reduction, and so on, the HMM framework employs a variety of techniques. In general, an HMM system will have n states, with 5 being the most common choice across various systems. Once the models are complete, they are utilized to recognize the input words using matching algorithms.

Even though it is effective for handling large numbers of words, it is critical for small-scale devices like smartphones. The language model must be reduced in size to improve the system's performance. Keyword-based, grammar-based, language-based, and phonetic-based language models can all be created with Sphinx. Because of its ability to handle command processing, the grammar-based language model has been used in the current system [18].

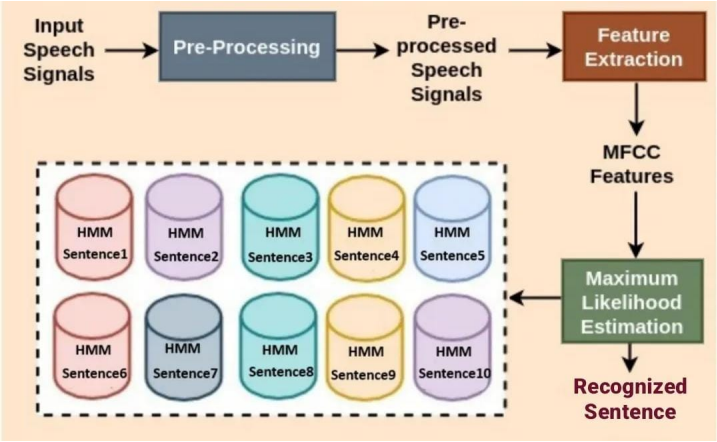


Figure 2.2 HMM testing

2.2.2 Limitations of HMM

Although the use of HMM technology has greatly aided recent advances in speech recognition, this type of statistical model for speech has some inherent limitations . The assumption that subsequent observations (frames of speech) are independent, and thus the probability of a series of observations may be represented as a product of probabilities of individual observations, is a significant constraint. Another drawback is the assumption that individual observation parameter distributions can be well represented as a mixture of Gaussian densities. Finally, the Markov assumption, which states that the probability of being in a given state at time t only depends on the state at time t - 1, is clearly inapplicable to speech sounds, where dependencies frequently span multiple states [19]. Despite these drawbacks, this type of statistical model has proven to be extremely effective for certain types of speech recognition issues.

Pre-emphasis

Filtering that highlights the higher frequencies is referred to as pre-emphasis. Its goal is to balance the frequency range of vocal sounds with a sharp high-frequency roll-off. The glottal source has a slope of about 12 dB/octave for voiced sounds . Pre-emphasis removes some of the glottal effects from the vocal tract parameters.

Frame Blocking and Windowing

The speech signal is a slowly time-varying signal. Speech must be analyzed for a sufficiently short time to have stable acoustic features. As a result, speech analysis should always be performed on short segments when the speech signal is presumed to be steady. Short-term spectral measurements are often made in 20ms windows, with 10ms intervals . The temporal properties of individual speech sounds can be followed by advancing the time frame every 10ms, and the 20ms analysis window is usually enough to offer good results. Sound's spectrum resolution, but also being short enough to resolve noteworthy temporal qualities.

Mel-Spectrum

Mel spectrum is calculated by passing the Fourier converted signal through a Mel-filter bank of band-pass filters. A Mel is a unit of measurement based on the frequency perceived by the human ear. Because the human auditory system does not appear to perceive pitch linearly, it does not relate linearly to the actual frequency of the tone. The Mel scale is approximately a linear frequency spacing below 1 kHz and a logarithmic spacing above 1 kHz. The formula used to caluculate the mels for any frequency is :

$$mel(f) = 2595x \log_{10}(1 + \frac{f}{100})$$

where mel(f) is the frequency (mels) and f is the frequency (Hz).

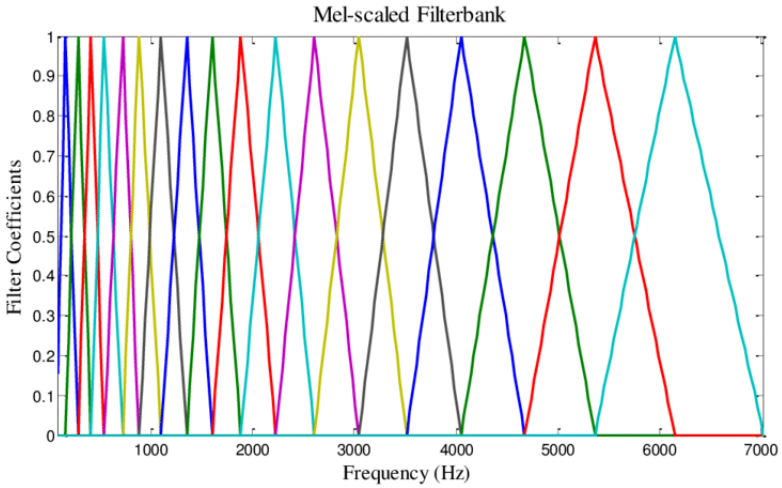


Figure 2.3 Mel-scaled Filterbank

Discrete Cosine Transformation

The energy levels in the vicinity bands tend to be connected since the vocal tract is smooth. When the DCT is applied to the transformed Mel frequency coefficients, cepstral coefficients are produced. The Mel spectrum is commonly shown on a log scale before DCT is computed. The Mel spectrum is commonly shown on a log scale before DCT is computed. This yields a cepstral domain signal with a que-frequency peak corresponding to the signal's pitch and a number of formants indicating low que-frequency peaks.

The MFCCs are caluculated using this equation

$$\hat{C}_n = \Sigma_{n=1}^{k} (\log \hat{S}_k) \cos[n(k - \frac{1}{2})\frac{\pi}{k}]$$

where k is the number of mel cepstrum coefficients, $\hat{S}_k$ is the output of filterbank and $\hat{C}_n$ is the final mfcc coefficients. So far, we discussed on the techniques and available in the literature to process speech signals. The methodology used in the present study is elaborated in the next chapter.

Chapter 3

Methodology

The system's modules are covered in this chapter. We must first gather the data and divide the dataset into training and assessing groups. 30 percent of the data is used for evaluation, with the remaining 70 percent going towards training. The model is built using the training data. [20]. and testing information is employed to assess the model. The training and evaluation process is shown in the block diagrams below [21] .

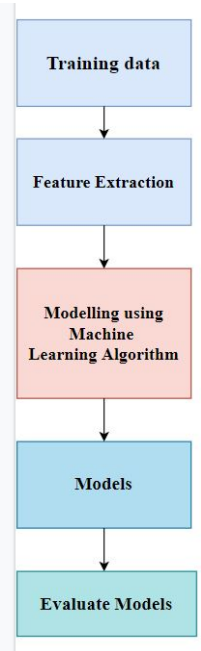


Figure 3.1 Training

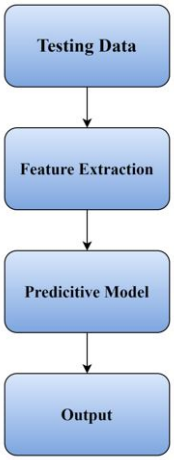


Figure 3.2 Evaluation

3.1 Data set collection

Data set collection involves acquiring the data. The data contains spoken data of Telugu sentences. The recordings were done with a sampling frequency 16kHz and a single channel using the Audio

Lab recordings application.

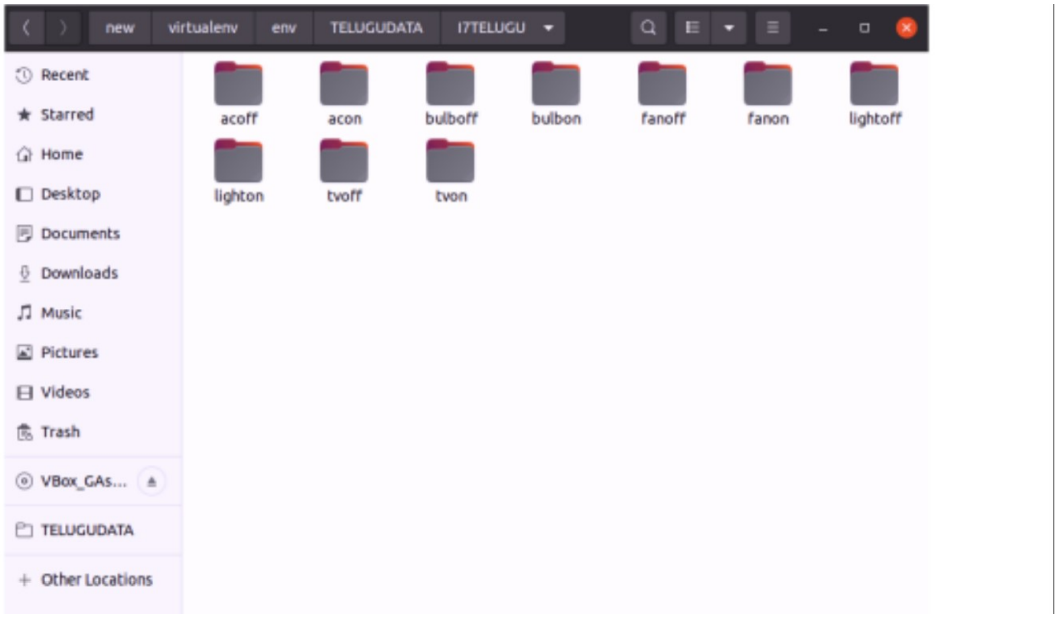


Figure 3.3 Recording files

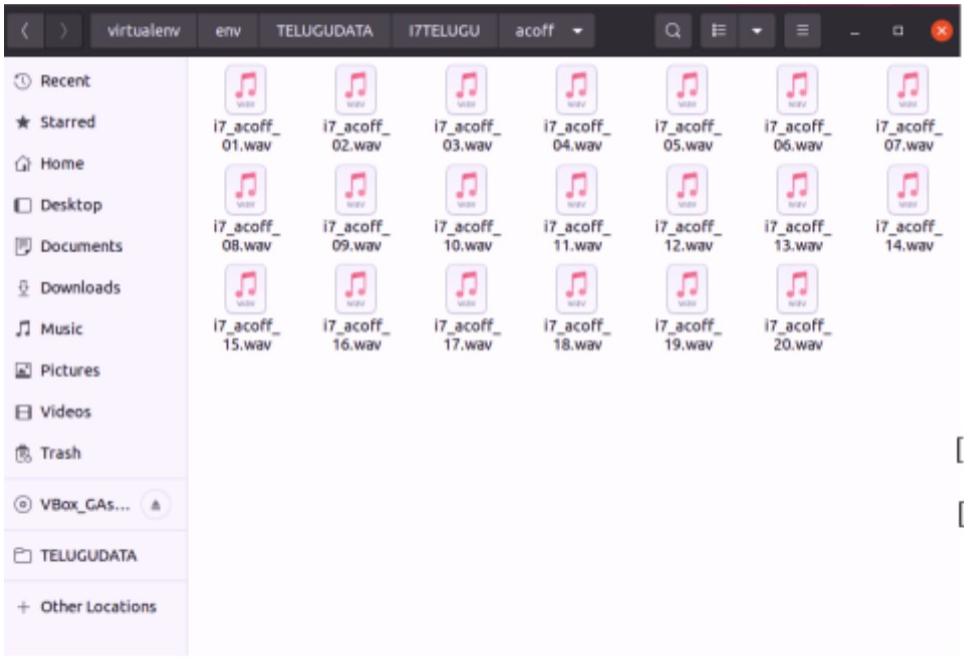


Figure 3.4 20 samples

3.2 Feature Extraction

Feature extraction refers to the process of transforming raw data into numerical features that can be processed while preserving the information in the original data set [16] .We can't take the raw audio signal as input to our model because there will be a lot of noise in the audio signal. It is observed that extracting features from the audio signal and using it as input to the base model will produce much better performance than directly considering raw audio signal as input.Here the actual data is very large in size so to reduce the complexity the actual data will be summarized which it can be transformed into reduced set of features in this step.And this data will be used for analysis and classification.The features that are generated will be used to build the models through modelling . Feature extraction can be accomplished manually or automatically:

1. Manual feature extraction requires identifying and describing the features that are relevant for a given problem and implementing a way to extract those features. In many situations, having a good understanding of the background or domain can help make informed decisions as to which features could be useful.

2. Automated feature extraction uses specialized algorithms or deep networks to extract features automatically from signals or images without the need for human intervention. This technique can be very useful when you want to move quickly from raw data to developing machine learning algorithms

The feature extraction techniques that are used in this project are:

1. Linear prediction coefficients (LPC)

   Linear prediction method is applied to obtain the filter coefficients equivalent to the vocal tract by reducing the mean square error in between the input speech and estimated speech. Linear prediction analysis of speech signal forecasts any given speech sample at a specific period as a linear weighted aggregation of preceding samples [20]. Subsequently, each frame of the windowed signal is autocorrelated, while the highest autocor- relation value is the order of the linear prediction analysis. This is followed by the LPC analysis,where each frame of the autocorrelations is converted into LPC parameters set which consists ofthe LPC coefficients. A summary of the procedure for obtaining the LPC is as seen in Figure.

Figure 3.5 LPC block diagram

2. Linear prediction cepstral coefficients (LPCC)

   In speech processing, LPCC analogous to LPC, are computed from sample points of a speech waveform, the horizontal axis is the time axis, while the vertical axis is the amplitude axis. The LPCC processor is as seen in Figure. It pictorially explains the process of obtaining LPCC. The cepstral coefficients derived from either linear prediction (LP) analysis or a filter bank approach are almost treated as standard front end features. Speech systems developed based on these features have achieved a very high level of accuracy, for speech recorded in a clean environment. Basically, spectral features represent phonetic information, as they are derived directly from spectra. The features extracted from spectra, using the energy values of linearly arranged filter banks, equally emphasize the contribution of all frequency components of a speech signal . In this context, LPCCs are used to capture emotion-specific information manifested through vocal tract features. In this work, the 10th order LP analysis has been performed, on the speech signal, to obtain 13 LPCCs per speech frame of 20 ms using a frame shift of 10 ms. The human way of emotion recognition depends equally on two factors, namely: its expression by the speaker as well as its perception by a listener. The purpose of using LPCCs is to consider vocal tract characteristics of the speaker, while performing automatic emotion recognition. Cepstrum may be obtained using linear prediction analysis of a speech signal. The basic idea behind linear predictive analysis is that the nth speech sample can be estimated by a linear combination of its previous p samples

Figure 3.6 LPCC block diagram

3. Mel frequency cepstral coefficients (MFCC)

   The MFCC feature extraction includes Windowing the signal, applying the DFT, taking the log of the magnitude, and then warping the frequencies on a Mel scale, followed by using the inverse DCT is all part of the MFCC feature extraction approach.

A sampling rate of 16000 samples per second is used. Each voice signal is separated into 16 ms windows, resulting in 256 samples.After applying feature extraction techniques, you'll have a matrix of feature vectors extracted from all of the frames. The rows in this output matrix correspond to the frame numbers, while the columns correspond to the feature vector coefficients . This output matrix is then used in the categorization procedure.

3.3 Modelling

It is the statistical represenation of the sounds which make up each word and it is used by speech recognition system to recognize the speech and it identifies the properties of language and predict the next word in speech sequence. Here in our project we used Hidden Markov model for modelling.

A hidden Markov model (HMM) is a statistical model in which the system being modeled is assumed to be a Markov process with unknown parameters; the challenge is to determine the hidden parameters from the observable data. In a hidden Markov model, the state is not directly visible, but variables influenced by the state are visible. Each state has a probability distribution over the possible output tokens. Therefore the sequence of tokens generated by an HMM gives some information about the sequence of states. A hidden Markov model can be considered a generalization of a mixture model where the hidden variables which control the mixture component to be selected for each observation, are related through a Markov process rather than independent of each other.

HMM creates stochastic models from known utterances and compares the probability that the unknown utterance was generated by each model. This uses theory from statistics in order to (sort of) arrange our feature vectors into a Markov matrix (chains) that stores probabilities of state transitions. That is, if each of our code words were to represent some state, the HMM would follow the sequence of state changes and build a model that includes the probabilities of each state progressing to another state.

HMMs are more popular because they can be trained automatically and are simple and computationally feasible to use HMM considers the speech signal as quasi- static for short durations and models these frames for recognition. It breaks the feature vector of the signal into a number of states and finds the probability of a signal to transit from one state to another. HMMs are simple networks that can generate speech (sequences of cepstral vectors) using a number of states for each model and modeling the short-term spectra associated with each state with, usually, mixtures of multivariate Gaussian distributions (the state output distributions). The parameters of the model are the state transition probabilities and the means, variances and mixture weights that characterize the state output distributions . This uses theory from statistics in order to (sort of) arrange our

feature vectors into a Markov matrix (chains) that stores probabilities of state transitions. That is, if each of our code words were to represent some state, the HMM would follow the sequence of state changes and build a model that includes the probabilities of each state progressing to another state.

3.4 Predictive model

Predictive modeling is the process of using known results to create, process, and validate a model that can be used to forecast future outcomes [22] .Predictive modeling is a statistical technique using machine learning to predict and forecast likely future outcomes with the aid of historical and existing data. It works by analyzing current and historical data and projecting what it learns on a model generated to forecast likely outcomes.

Through Hidden Markov Model(HMM) the models were built.Then we have to evaluate the models for checking the correctness of the system.Testing data is used to evaluate the system. In this step,it predicts the output based on maximum likelihood.Whenever the testing data is given as input it will compare with each existing models,the model with highest probability is selected and gives the output as recognized word. So far we have discussed the data set collection and feature extraction techniques and modelling of the sentences. The Implementation details and results will be shown in the next chapter.

Chapter 4

Implementation Details And Results

This chapter discusses the details of the project implementation along with the obtained results. The details presented in this chapter are categorized as follows:

1. Data Set Description
2. Python Libraries
3. Implementation of Speech Processing module

4.1 Data set Description

The Telugu speech recordings are recorded in simple and easy words that every person can use the application especially for the elder and physically challenged people [23] . The Speech recording is done in the Telugu language collecting the data from different speakers saying 10 sentences in Telugu language [24] . Each sentecne is recorded as 20 utterances. The speech recordings are recorded in an Android application called 'Audio Lab'. The speech recording is done in .WAV file format.
In mono mode, data were recorded at a sampling rate of 16 kHz with a 16-bit coding rate. The data collection is voices of the speaker with background noise.

Table 4.1 Recording parameters

S.No	Parameter	Description
1.	Sampling rate	16kHz
2.	No. of Channels	Mono
3.	Bits per sample	32

The present data set is aimed to design speech recognition systems for digit recognition in a mobile device. Therefore, the words are chosen to suit the needs of the applications. The complete list of words in the data set is described in Table IV with individual word count along with the phonetic units in each word. The data set consists of an equal number of samples for each class to

Table 4.2 List of Telugu Sentences

S.No	Sentence	Description
1.	ఫ్యాన్ తిప్పండి	Fan on
2.	ఫ్యాన్ ఆపండి	Fan off
3.	బల్బ్ వెలిగించండి	Bulb on
4.	బల్బ్ ఆపండి	Bulb off
5.	లైట్ వెలిగించండి	Light on
6.	లైట్ ఆపండి	Light off
7.	టీవీ పెట్టండి	Tv on
8.	టీవీ కట్టేయండి	Tv off
9.	ఏసీ వేయండి	Ac on
10.	ఏసీ ఆపండి	Ac off

avoid the class imbalance problem. Each word has a combination of vowels and consonants. The presence of vowels in each word in a phoneme plays a significant role in the recognition process. In fact, the chosen words can also help to study the nature of Telugu vowels while giving enough useful clues. In a study done by Mykala et al. the vowels are examined through the changes occur in the acoustic features of the speech signals. Therefore, the words used in the dataset can be used for multiple studies [25]

4.2 Python Libraries

This section gives a broad view of the libraries used for implementing the modules in the system with the issues that were handled. The following modules were used for implementing the speech recognition module [26] .

1. librosa
2. python_speech_features
3. hmmlearn
4. pickle
5. numpy

The speech recognizer was implemented in two different ways:

- Initially, a speech recognition system was developed using the Google Speech library for testing the hardware platform. This module uses English for communicating with the Raspberry Pi.
- Next, a customized speech recognition system was developed using different python libraries to recognize the commands in the Telugu language.

The initially developed system was tested using English commands by different users and found to be effective. However, the system doesn't work for the Telugu language as it was trained for the language which opens the reason for building a Telugu speech recognition system [27].

librosa

Librosa is a Python package for music and audio analysis. Librosa is basically used when we work with audio data like in music generation(using LSTM's), Automatic Speech Recognition.It provides the building blocks necessary to create the music information retrieval systems . Librosa helps to visualize the audio signals and also do the feature extractions in it using different signal processing techniques [28] .

Command: pip install librosa

NumPy

NumPy is a Python library used for working with arrays. It also has functions for working in the domain of linear algebra, Fourier transform, and matrices [29] . NumPy was created in 2005 by Travis Oliphant. It is an open-source project and you can use it freely [30] . NumPy stands for Numerical Python. We can install NumPy as:

Command: pip install numpy

hmmlearn

The hmmlearn is a set of algorithms for learning and inferring Hidden Markov Models without supervision. The hmmlearn implements the hidden Markov models. Hidden Markov Models are used for training the speech files and creating models. So that Modeling helps us in recognizing and matching the recording statements [31] .
These are the required dependencies to use hmmlearn
Python >= 3.5
NumPy >= 1.10
scikit-learn >= 0.16
And we also require
Matplotlib >= 1.1.1 to run the examples and pytest >= 2.6.0 to run the tests.
It requires C compiler and python headers.

Command: pip install –upgrade –user hmmlearn

pip install hmmlearn

HMM

HMMs is a Python module for Hidden Markov Models. It's a simple, general-purpose library that includes all of the necessary sub-methods for training, analyzing, and experimenting with data models [32].

command: pip install hmms

Table 4.3 Parameters used in HMM

S. No.	Parameter	Description
1.	Number of states in HMM	5
2.	Co-variance Matrix Type	Full
3.	Number of iterations for convergence	150
4.	Emission Probability	Gaussian Mixture

OS

In Python, the OS module has functions for dealing with the operating system. Python's standard utility modules include OS [33] . This module allows you to use operating system-dependent functions on the go. Many functions to interface with the file system are included in the *os* and *os. path* modules [34] .

command: pip install os-win

Pickle

Pickle is for serializing and de-serializing Python object structures, the pickle package is utilized. Pickling, serialization, flattening, and marshaling are all terms for the process of converting any type of Python object (list, dictionary, etc.) into byte streams (0s and 1s).
Pickle is generally used in Python to serialize and de-serialize Python object structures. To put it another way, it's the process of transforming a Python object into a byte stream in order to save it to a file/database, maintain program state across sessions, or send data over the network [35] .

command: pip install pickle-mixin

pip install pickle5

4.3 Implementation of Telugu Speech Recognition System

Hidden Markov Model(HMM), Linear prediction coefficients (LPC),Linear prediction cepstral coefficients (LPCC),Mel frequency cepstral coefficients (MFCC) were used to recognize Telugu speech.

- Voice recordings will be detected and speech files will be trained in the HMM. After the voice files have been trained, models for those voice files will be developed. One model will be constructed for each speech file [36] .

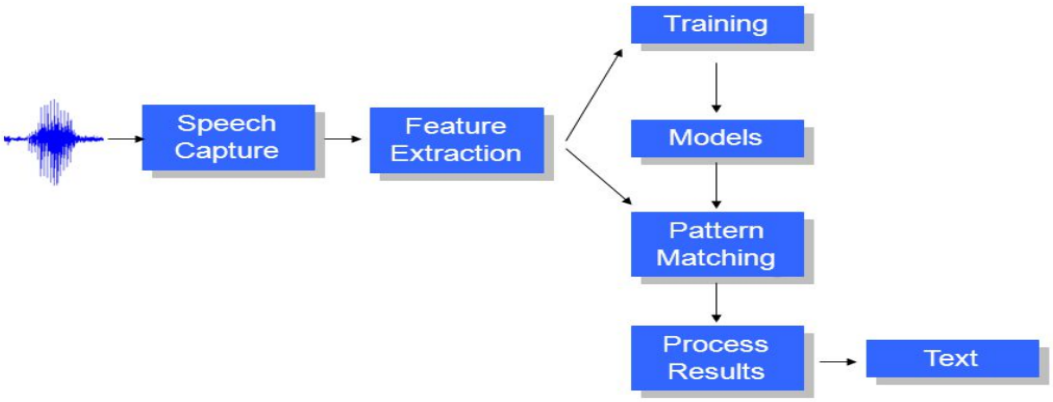


Figure 4.1 Modeling of speech files

4.4 Sample Code Snippets

feature_extraction.py

```
import matplotlib.pyplot as plt
import matplotlib.axes as ax
num_ceps = 13
lifter = 2
normalize = True
path="F:\BTP2022\digits"
dirlist=os.listdir(path)
files=[]
windowsize=10000
for i in range(10):
        str=os.path.join(path,dirlist[i])
        files.append(str)
        #print(files)
        temp=files[i].split("\\")
```

```
        #print(temp)
        a=temp[3].split(".")[0]
        #print(files)
        inputsignal=[]
        (srate,sig)=wav.read(files[i])
        #plt.plot(sig[4000:6000],color="black")
        #plt.tick_params(axis='both',labelsize=20)
        if len(sig)<windowsize:
                for k in range(0,len(sig)):
                        inputsignal.append(sig[k])
                for k in range(len(sig),windowsize):
                        inputsignal.append(0.0)
        # OTHERWISE USE ONLY WINDOWSIZE NUMBER OF SAMPLES
        if len(sig)>windowsize:
                inputsignal=sig[0:windowsize]
        signal=[float(item) for item in inputsignal]
        signal=np.asarray(signal,dtype=float)
        mfcc_feat=lpc(sig=signal, fs=srate, num_ceps=13)
```

hmm_train1.py

```
## PROGRAM TRANSFORMS THE GIVEN SPEECH SIGNALS TO MFCC FEATURES AND CREATES MODELS FOR THE SENTENCES ##

from python_speech_features import mfcc
from spafe.features.lpc import lpc, lpcc
from python_speech_features import logfbank
import scipy.io.wavfile as wav
import os,sys
import traceback
import numpy as np
import hmmlearn
from hmmlearn import hmm
import pickle
import os
import statistics

path="Inputs"
#path="17761A0592"
```

```
class SIG_PROCESS:
        def __init__(self,dirs,wsize,modelfolder):
                try:
                        self.windowsize=wsize
                        self.dirs=dirs
                        self.modelfolder=modelfolder
                        print(self.dirs)
                        self.files=os.listdir(os.path.join(path,dirs))
                        self.filelist=[]
                    # PREPARE THE LIST OF FILES
                        for i in range(0,len(self.files)):
                                if self.files[i].endswith('wav'):
                                        str=os.path.join(path,dirs,self.files[i])
                                        self.filelist.append(str)
                    #print(os.path.join(path,dirs,self.files[i]))
                except Exception as e:
                        print(e)
                        traceback.print_exc()
```

hmm_test1.py

```
## PROGRAM TRANSFORMS THE GIVEN SPEECH SIGNALS TO MFCC FEATURES AND CREATES MODELS FOR THE SENTENCES ##
from python_speech_features import mfcc
from python_speech_features import logfbank
import scipy.io.wavfile as wav
import os,sys
import traceback
import numpy as np
import hmmlearn
from hmmlearn import hmm
import pickle
import os
import statistics

TestPath=["SPEAKERDEPENDENTANALYSIS/TESTDATA/testdataperson1","SPEAKERDEPENDENTANALYSIS/
class HMM_TEST:
        def __init__(self,path):
                try:
                        self.files=os.listdir(os.path.join(path))
```

```
            self.filelist=[]
        # PREPARE THE LIST OF FILES
            for i in range(0,len(self.files)):
                    str=os.path.join(path,self.files[i])
                    self.filelist.append(str)
                #print(os.path.join(path,dirs,self.files[i]))
            print('File List: ',self.filelist)

    except Exception as e:
            print(e)
            traceback.print_exc()
```

4.5 Results

When the audio file is given as input the librosa function will read in the path to an audio file, and return a tuple with two items. The first item is an 'audio time series' (type:array) corresponding to audio track [37] . The second item in the tuple is the sampling rate that was used to process the audio. Audio files contains the 20 sentences which are spoken in Telugu language by different speakers. The resulted arrays plotted as shown in below. Table shows the graphs of each input signal of the sentences which are spoken in Telugu language. X-axis represents the Sample and Y-axis represents the Amplitude of input signal.The graphs are plotted over few samples of input signal .

We can't take the raw auido signal as input to our model because the actual data is very large in size and requires more computational power to process and there will be a lot of noise in the audio signal .So to reduce the complexity the actual data will be summarized which it can be transformed into reduced set of features. It is observed that extracting features from the audio signal and using it as input to the base model will produce much better performance than directly considering raw audio signal as input [38] .It helps to reduce the redundancy of data and reduces the data which helps to build the models with less machine effort and increase the speed of learning.This process is called feature extraction as it transforms raw data into sentence features that can be processed while preserving the information in the original dataset. It yields better results than applying machine learning directly to raw data. Linear prediction coefficients (LPC), Linear prediction cepstral coefficients (LPCC), Mel frequency cepstral coefficients (MFCC) are the techniques used to extract the features. In feature extraction it divides the input signal into frames. [9] .

The following tables represents the graphs of input signals of 10 sentecnes spoken in telugu language after applying LPC,LPCC,MFCC feature extraction techniques [39] .

X-axis represents the frame number, and Y-axis represents the coefficient values and each line represents the corresponding feature [40] .

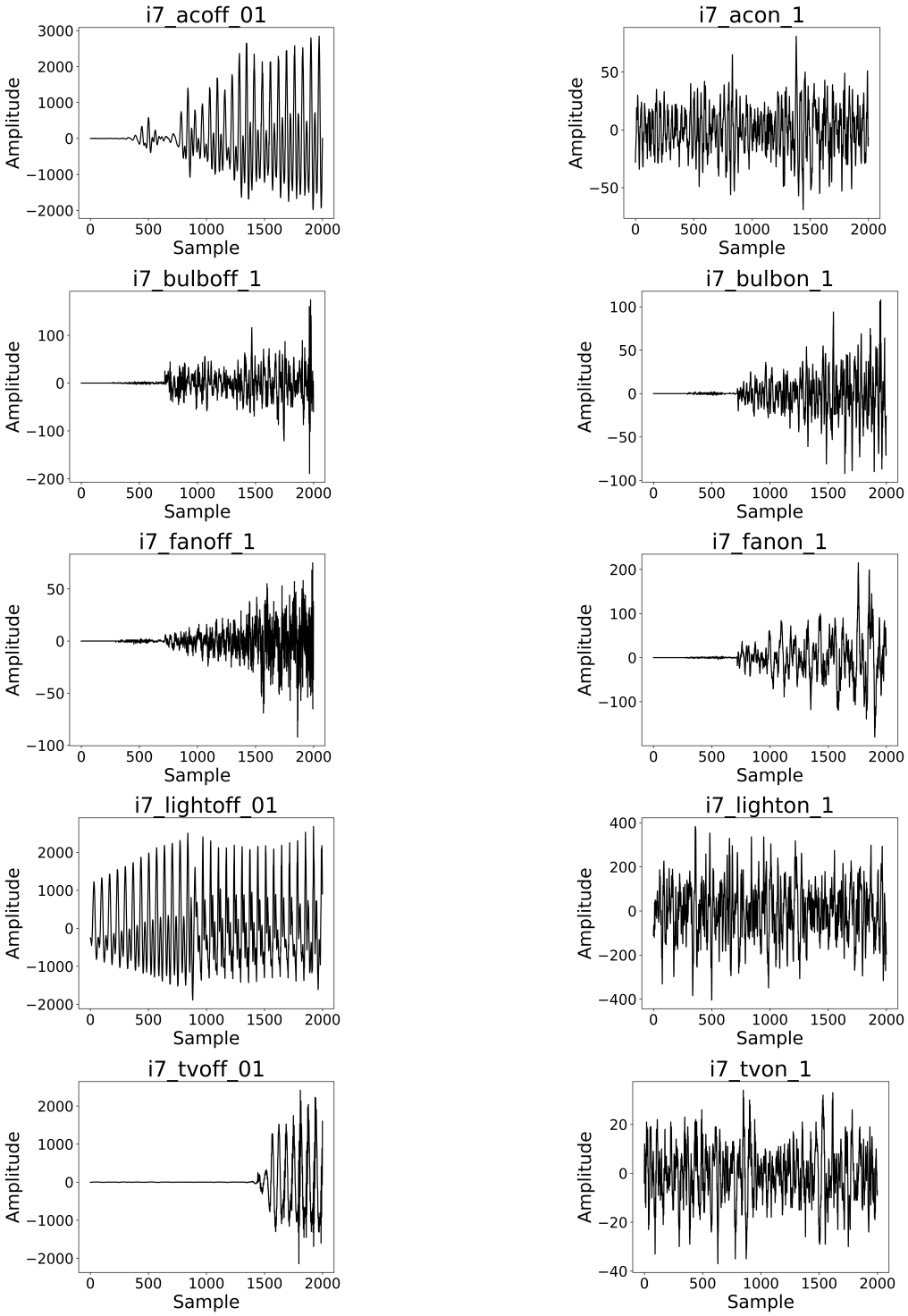


Figure 4.2 Utterances of the sentences

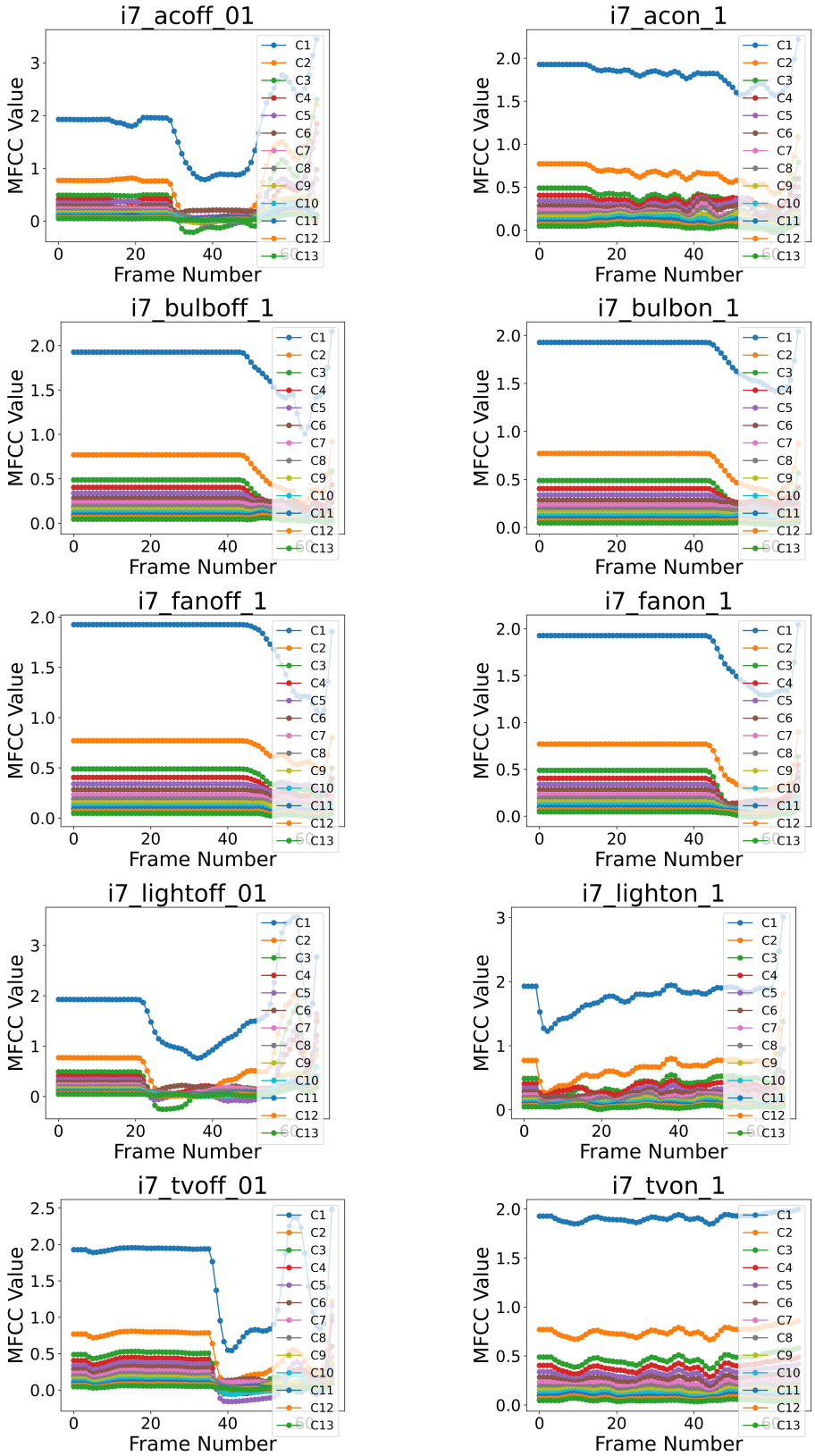


Figure 4.3 Visualization of Linear Prediction Coefficients (LPC)

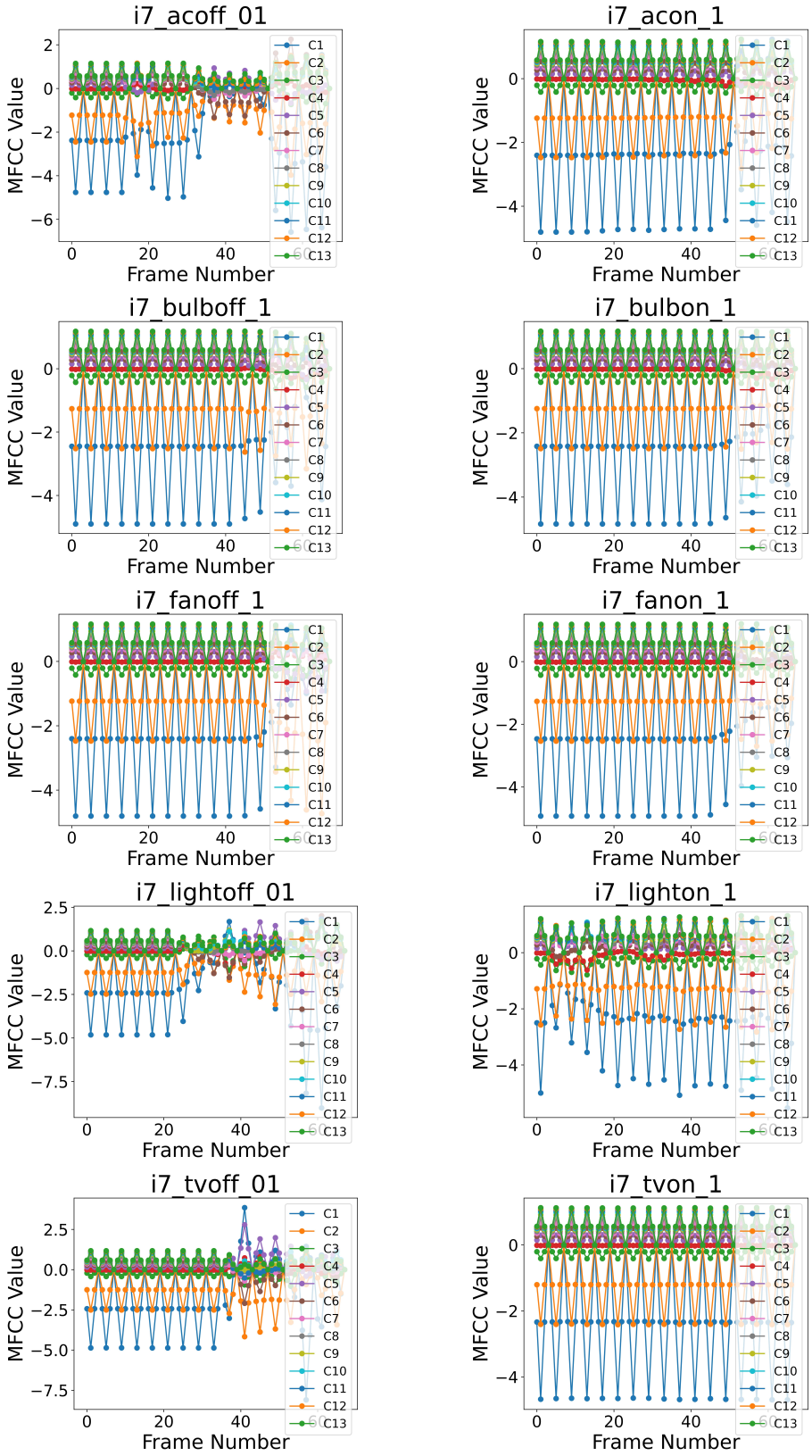


Figure 4.4 Visualization of Linear Prediction Cepstral Coefficients (LPCC)

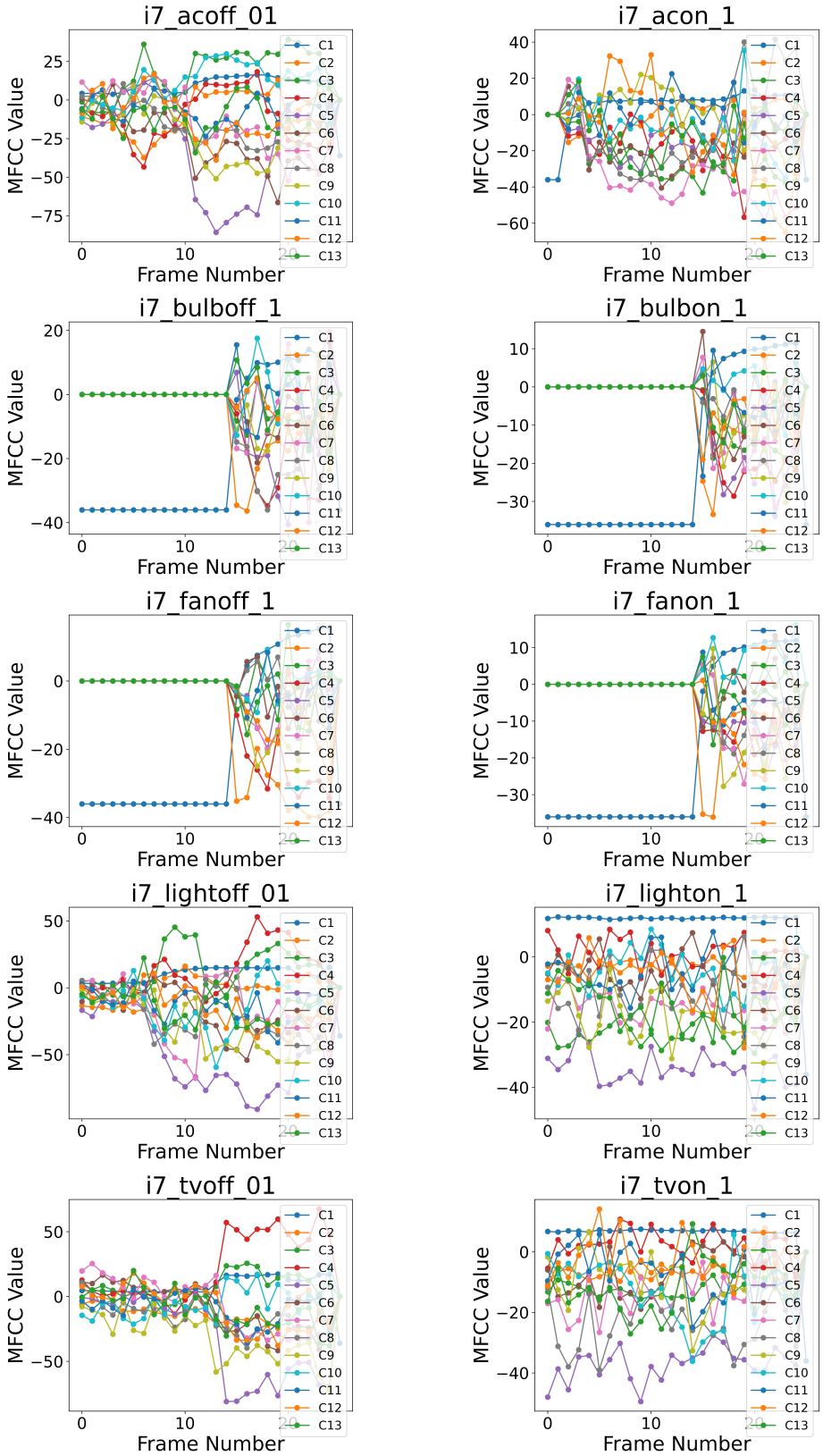


Figure 4.5 Visualization of Mel Frequency Cepstral Coefficients (MFCC)

Feature Extraction gives the feature vectors which can be used to built the models through modelling. In this Project Hidden Markov Model is used to built the models [41] . It is a state transition model in which it works on the probability distribution. HMM creates stochastic models from known utterances and compares the probability that the unknown utterance was generated by each model. This uses theory from statistics inorder to arrange our feature vector into a Markov Matrix that stores probabilities of state transition. That is, if each of our code words were to represent some state, the HMM would follow the sequence of state changes and build a model that includes the probabilities of each state progressing to another state. After executing the hmm_train1.py program, the models will be created. The following figure shows the models for the sentences created through Hidden Markov Model (HMM). The models are created in .pkl format.PKL file is a file created by pickle, a python module that enables objects to be serialized to files on disk and deserialized back into the program at runtime. It contains a byte stream that represents the objects [42] .

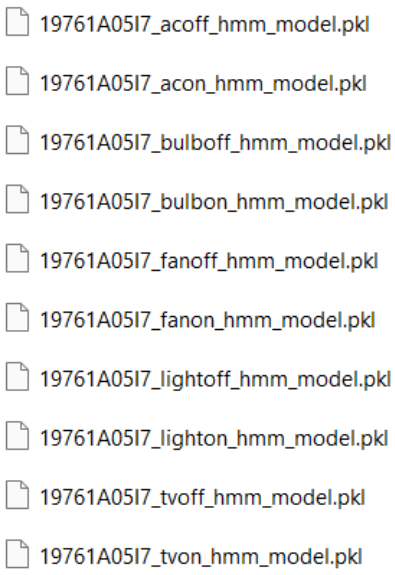


Figure 4.6 Models

HMM breaks the feature vector of the signal into number of states and finds the probability of a signal to transmit from one state to other. HMMs are simple networks that can generate speech (sequences of cepstral vectors) using a number of states for each model and modeling the short-term spectra associated with each state with, usually, mixtures of multivariate Gaussian distributions (the state output distributions). The parameters of the model are the state transition probabilities and the means, variances and mixture weights that characterize the state output distributions.

After the models were created through training data we have to evaluate the models through the testing data. Whenever the testing data is given as input,after the feature extraction it will compare the features with the existing models. The model which give highest probablity will be selected and gives corresponding digit as output. To evaluate models we have to run the hmm_test1.py program.After the execution of this program it will give the percentage of correctly recognized digits [43] . The following Figure 4.17 shows the recognition rate for one speaker [44] .

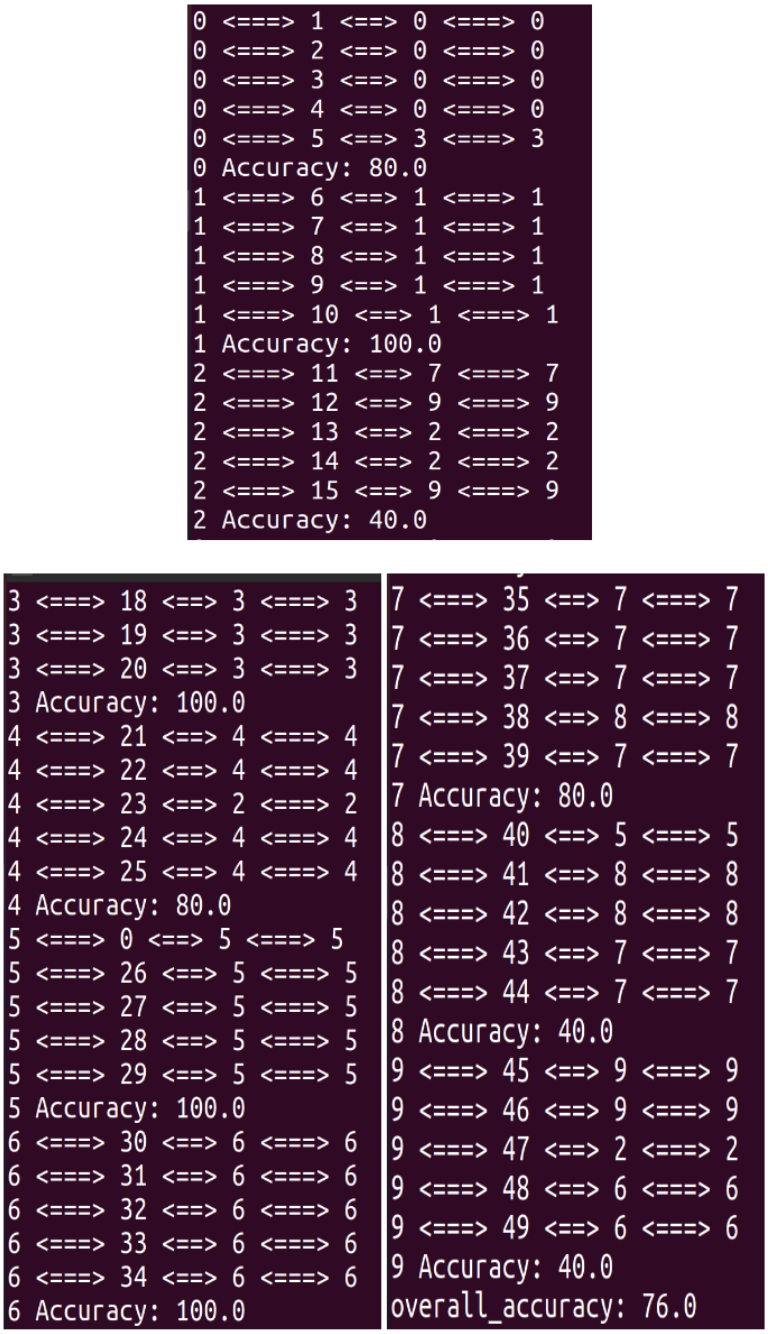


Figure 4.7 Recogniton rate

Similarily we have to train and evaluate the models for remaining speakers with LPC,LPCC,MFCC feature extraction techniques. The results of the speech recognition experiments for 61 speakers are summarized in Table 4.4

Speaker No	**LPC**	**LPCC**	**MFCC**	**Speaker No**	**LPC**	**LPCC**	**MFCC**
Speaker 1	90	65	45	Speaker 25	89	47.5	45
Speaker 2	77.77	72.22	50	Speaker 26	81.08	60	56.756
Speaker 3	97.22	69.44	55.55	Speaker 27	77.77	72.2	50
Speaker 4	90	67.5	50	Speaker 28	87.5	62.5	52
Speaker 5	87.5	62.5	50	Speaker 29	50	67.5	52
Speaker 6	92.5	65	62.5	Speaker 30	91.68	83.33	72.2
Speaker 7	91.666	83.33	72.22	Speaker 31	85	80	77.54
Speaker 8	92.5	87.5	80	Speaker 32	77.77	72.22	50.4
Speaker 9	93	64.1025	69.23	Speaker 33	87	67.5	57
Speaker 10	91.666	70.833	58.33	Speaker 34	87	67.5	57
Speaker 11	60	45.714	51.42	Speaker 35	81	70.2	56.7
Speaker 12	92.5	75	72.5	Speaker 36	85	80	77
Speaker 13	97.5	55	40	Speaker 37	86	70.8	58.334
Speaker 14	90	65	65	Speaker 38	81.08	75	72
Speaker 15	85	80	77.5	Speaker 39	92.5	60	55
Speaker 16	87	67.5	57.5	Speaker 40	81.084	70	56
Speaker 17	95	70	57.5	Speaker 41	90	65	65
Speaker 18	93	72.5	57.5	Speaker 42	87	67.3	56.2
Speaker 19	77	72.22	55	Speaker 43	87	71.77	47
Speaker 20	97.5	57.5	60	Speaker 44	89.5	70.8	58.3
Speaker 21	78	80	77.5	Speaker 45	81	67.3	56
Speaker 22	89	63.63	52.27	Speaker 46	90	65	60
Speaker 23	93	72.5	55	Speaker47	47	67.5	57.5
Speaker 24	95	70	60	Speaker 48	81	67.3	56

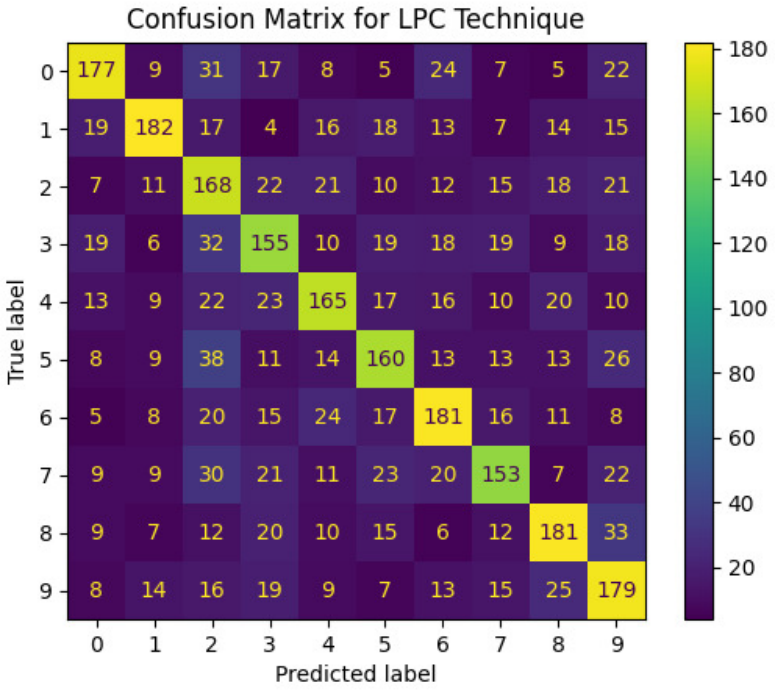
Table 4.4 Recognition rates for LPC, LPCC, MFCC

Confusion Matrix

A confusion matrix, also known as an error matrix, is a summarized table used to assess the performance of a classification model. The number of correct and incorrect predictions are summarized with count values and broken down by each class [45] .

- Positive (P): Observation is positive
- Negative (N): Observation is not positive
- True Positive (TP): Outcome where the model correctly predicts the positive class.
- True Negative (TN): Outcome where the model correctly predicts the negative class.
- False Positive (FP): Also called a type 1 error, an outcome where the model incorrectly predicts the positive class when it is actually negative.
- False Negative (FN): Also called a type 2 error, an outcome where the model incorrectly predicts the negative class when it is actually positive.

The figures show the confusion matrices for LPC,LPCC and MFCC with size 10X10. X-axis represents the predicted label and Y-axis represents the true label of digits 0-9. The diagonal elements represent the number of points for which the predicted label is equal to the true label, while off-diagonal elements are those that are mislabeled by the classifier. The higher the diagonal values of the confusion matrix the better, indicating many correct predictions.



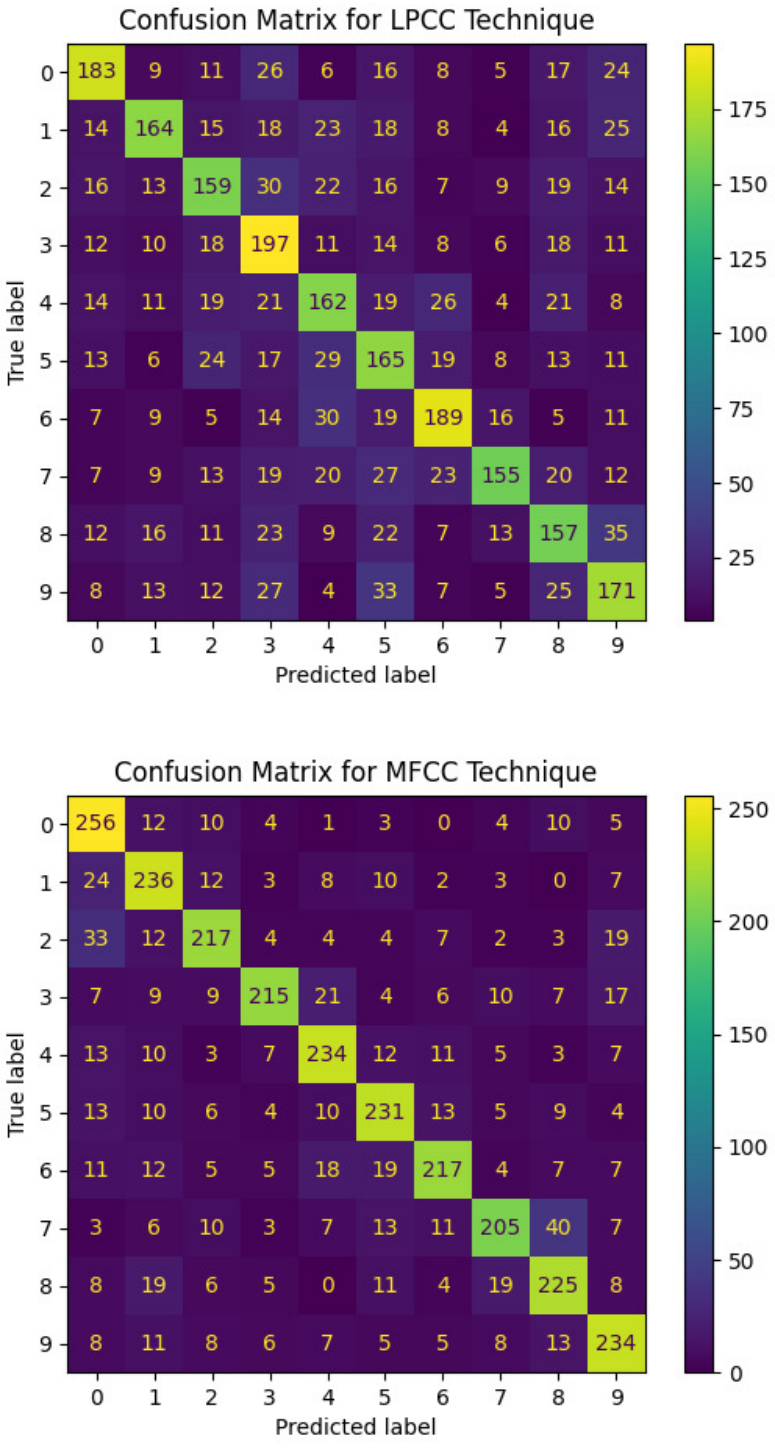


Figure 4.8 Confusion Matrix

Classification Report

It is one of the performance evaluation metrics of a classification-based machine learning model. It displays your model's precision, recall, F1 score and support. It provides a better understanding of the overall performance of our trained model [46] .

- Accuracy : This is simply equal to the proportion of predictions that the model classified correctly.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

- Precision : Precision is defined as the ratio of true positives to the sum of true and false positives.

$$Precision = \frac{TP}{TP + FP}$$

- Recall : Recall is defined as the ratio of true positives to the sum of true positives and false negatives.

$$Recall = \frac{TP}{TP + FN}$$

- F1 Score : The F1 is the weighted harmonic mean of precision and recall. The closer the value of the F1 score is to 1.0, the better the expected performance of the model is.

$$F1score = \frac{2 * precision}{precision + recall} = \frac{2TP}{2TP + FP + FN}$$

- Support : Support is the number of actual occurrences of the class in the dataset. It doesn't vary between models, it just diagnoses the performance evaluation process.

The following tables show the classification reports for LPC,LPCC and MFCC techniques

	Precision	**Recall**	**f1-score**	**Support**
ఫ్యాన్ తిప్పండి	0.57	0.54	0.56	305
ఫ్యాన్ ఆపండి	0.55	0.52	0.54	305
బల్బ్ వెలిగించండి	0.57	0.59	0.58	305
బల్బ్ ఆపండి	0.57	0.50	0.53	305
లైట్ వెలిగించండి	0.60	0.59	0.60	305
లైట్ ఆపండి	0.51	0.59	0.54	305
టీవీ పెట్టండి	0.65	0.58	0.61	305
టీవీ కట్టేయండి	0.69	0.60	0.64	305
ఏసీ వేయండి	0.44	0.55	0.49	305
ఏసీ ఆపండి	0.50	0.51	0.51	305
Accuracy			0.56	3050
Macro avg	0.56	0.56	0.56	3050
Weighted avg	0.56	0.56	0.56	3050

Table 4.5 Classification report for LPC

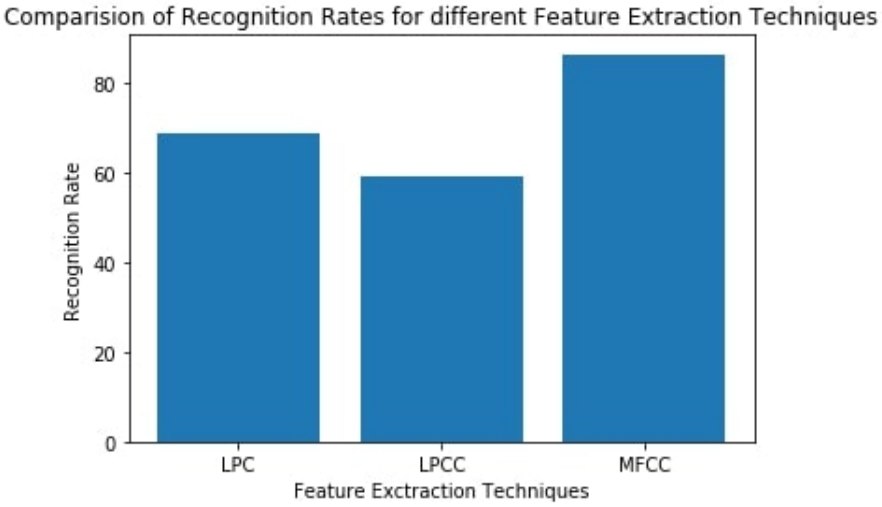
	Precision	**Recall**	**f1-score**	**Support**
ఫ్యాన్ తిప్పండి	0.51	0.53	0.52	305
ఫ్యాన్ ఆపండి	0.47	0.54	0.50	305
బల్బ్ వెలిగించండి	0.63	0.62	0.62	305
బల్బ్ ఆపండి	0.69	0.51	0.58	305
లైట్ వెలిగించండి	0.50	0.51	0.51	305
లైట్ ఆపండి	0.53	0.56	0.55	305
టీవీ పెట్టండి	0.64	0.60	0.62	305
టీవీ కట్టేయండి	0.63	0.54	0.58	305
ఏసీ వేయండి	0.55	0.52	0.54	305
ఏసీ ఆపండి	0.50	0.65	0.57	305
Accuracy			0.56	3050
Macro avg	0.57	0.56	0.56	3050
Weighted avg	0.57	0.56	0.56	3050

Table 4.6 Classification report for LPCC

	Precision	Recall	f1-score	Support
ఫ్యాన్ తిప్పండి	0.74	0.76	0.75	305
ఫ్యాన్ ఆపండి	0.79	0.71	0.75	305
బల్బ్ వెలిగించండి	0.77	0.67	0.72	305
బల్బ్ ఆపండి	0.71	0.74	0.72	305
లైట్ వెలిగించండి	0.74	0.77	0.75	305
లైట్ ఆపండి	0.68	0.84	0.75	305
టీవీ పెట్టండి	70	0.77	0.74	305
టీవీ కట్టేయండి	0.76	0.71	0.73	305
ఏసీ వేయండి	0.84	0.70	0.77	305
ఏసీ ఆపండి	0.75	0.77	0.76	305
Accuracy			0.74	3050
Macro avg	0.75	0.74	0.74	3050
Weighted avg	0.75	0.74	0.74	3050

Table 4.7 Classification report for MFCC

The accuracy comparision between LPC,LPCC and MFCC feature extraction techniques are depicted in the following figure. In the form of bar graph, we demonstrated the recognition rates for LPC,LPCC and MFCC feature extraction techniques.On an average the recognition rate with LPC feature extraction technique is 68.97%,for LPCC it is 59.112% and for MFCC we got 86.5273%.By comparing the three feature extraction techniques we can say that with the help of Mel Frequency Cepstral Coefficients (MFCC) feature extraction technique we got more accuracy. In the following bar chart the X-axis represent Feature Extraction Techniques and Y-axis represent Recognition Rate of recognition rate(%).



Chapter 5

Conclusions and Future work

5.1 Conclusions

Our project's main objective is to identifying the Telugu sentences spoken in a particular utterance. So that it is useful for elders and physically challenged people without having trouble while moving from one place to another. For this we have collected 48 data sets i.e; 20 voice recordings with 10 sample sentences from different persons irrespective of age and gender. The voice recordings are recorded in Telugu Language.

5.2 Future Work

In the future, we are trying to extend our project by recording more data sets and using multiple languages. We are planning to collect more data sets. We will include multiple languages to control home appliances so that people with disability and elder people can use this application . It is useful for all the people who use different languages. Every individual will have their own language like in every place there will be a different language. So by using multiple languages the application can be useful for all kinds of people.

5.3 Publication

- *Parabattina Bhagath, Vanga Lasya and P Dhyeya* , **"TELUGU VAKYALU: Telugu Sentence Dataset For IoT Applications,"** 13th International Conference on Pattern Recognition (ICPRS 2023), July 04-07 Guayaquil - Ecuador, (Submitted)

Bibliography

[1] P. Bhagath and P. K. Das, "Characterization of spoken english vowels using tree structures," in *TENCON 2019-2019 IEEE Region 10 Conference (TENCON)*, pp. 1758–1763, IEEE, 2019.

[2] L. Wang, R. Tong, C.-C. Leung, S. Sivadas, C. Ni, and B. Ma, "Cloud-based automatic speech recognition systems for southeast asian languages," in *2017 International Conference on Orange Technologies (ICOT)*, pp. 147–150, 2017.

[3] K. Ramasamy, N. K., P. S., and T. Subha, "Voice and speech recognition in tamil language," pp. 288–292, 02 2017.

[4] T. Tambe, E.-Y. Yang, G. G. Ko, Y. Chai, C. Hooper, M. Donato, P. N. Whatmough, A. M. Rush, D. Brooks, and G.-Y. Wei, "9.8 a 25mm2 soc for iot devices with 18ms noise-robust speech-to-text latency via bayesian speech denoising and attention-based sequence-to-sequence dnn speech recognition in 16nm finfet," in *2021 IEEE International Solid- State Circuits Conference (ISSCC)*, vol. 64, pp. 158–160, 2021.

[5] R. Parikh and H. Joshi, "Gujarati speech recognition -a review," 10 2020.

[6] Z. Ma, Y. Liu, X. Liu, J. Ma, and F. Li, "Privacy-preserving outsourced speech recognition for smart iot devices," *IEEE Internet of Things Journal*, vol. 6, no. 5, pp. 8406–8420, 2019.

[7] M. A. Menacer, O. Mella, D. Fohr, D. Jouvet, D. Langlois, and K. Smaili, "An enhanced automatic speech recognition system for Arabic," in *Proceedings of the Third Arabic Natural Language Processing Workshop*, (Valencia, Spain), pp. 157–165, Association for Computational Linguistics, Apr. 2017.

[8] J. Brousseau, C. Drouin, G. Foster, P. Isabelle, Y. Normandin, and P. Plamondon, "French speech recognition in an automatic dictation system for translators: the transtalk project," 09 1995.

[9] P. D'Orta, M. Ferretti, A. Martelli, and S. Scarci, "An automatic speech recognition system for the italian language," in *Proceedings of the Third Conference on European Chapter of the Association for Computational Linguistics*, EACL '87, (USA), p. 80–83, Association for Computational Linguistics, 1987.

[10] P. Kumar, T. Yadava, and H. Jayanna, "Continuous kannada speech recognition system under degraded condition," *Circuits, Systems, and Signal Processing*, vol. 39, 01 2020.

[11] A. Anand, S. Devi, J. Stephen, and V. Bhadran, "Malayalam speech recognition system and its application for visually impaired people," pp. 619–624, 12 2012.

[12] H. Sarfraz, S. Hussain, R. Bokhari, A. A. Raza, I. Ullah, Z. Sarfraz, S. Pervez, A. Mustafa, I. Javed, and R. Parveen, "Large vocabulary continuous speech recognition for urdu," in *Proceedings of the 8th International Conference on Frontiers of Information Technology*, FIT '10, (New York, NY, USA), Association for Computing Machinery, 2010.

[13] J. H. Tailor and D. B. Shah, "Hmm-based lightweight speech recognition system for gujarati language," in *Information and Communication Technology for Sustainable Development* (D. K. Mishra, M. K. Nayak, and A. Joshi, eds.), (Singapore), pp. 451–461, Springer Singapore, 2018.

[14] M. Garg and N. Aggarwal, "Punjabi speech recognition: A survey," 05 2014.

[15] M. Sarma and K. Sarma, "Speech recognition in indian languages—a survey · hidden markov model (hmm) · gaussian mixture model (gmm) · artificial neural network (ann)," *Recent Trends in Intelligent and Emerging Systems, Signals and Communication Technology, Springer*, 05 2015.

[16] P. Bhagath, M. Pullagura, P. K. Das, V. K. Yandra, and S. S. Thetla, "Telugu ankelu: A telugu spoken digits corpora for mobile speech recognition," in *2022 12th International Conference on Pattern Recognition Systems (ICPRS)*, pp. 1–6, 2022.

[17] C. Gao, S. Braun, I. Kiselev, J. Anumula, T. Delbruck, and S.-C. Liu, "Real-time speech recognition for iot purpose using a delta recurrent neural network accelerator," in *2019 IEEE International Symposium on Circuits and Systems (ISCAS)*, pp. 1–5, 2019.

[18] A. Beg and S. Hasnain, "A speech recognition system for urdu language," vol. 20, pp. 118–126, 04 2008.

[19] D. Srinivasan, B. B, T. Durairaj, and S. K. B, "SSNCSE_NLP@LT-EDI-ACL2022: Speech recognition for vulnerable individuals in Tamil using pre-trained XLSR models," in *Proceedings of the Second Workshop on Language Technology for Equality, Diversity and Inclusion*, (Dublin, Ireland), pp. 317–320, Association for Computational Linguistics, May 2022.

[20] S. Yang, Z. Gong, K. Ye, Y. Wei, Z. Huang, and Z. Huang, "Edgernn: A compact speech recognition network with spatio-temporal features for edge computing," *IEEE Access*, vol. 8, pp. 81468–81478, 2020.

[21] R. Deshmukh, Y. Gedam, S. Bhosle, and A. Dabhade, "Development of automatic speech recognition of marathi numerals-a review," *International Journal of Engineering and Innovative Technology*, vol. 3, pp. 198–203, 03 2014.

[22] P. Bhagath, S. Parihar, and P. K. Das, "Speech recognition for indian spoken languages towards automated home appliances," in *2021 2nd International Conference for Emerging Technology (INCET)*, pp. 1–5, 2021.

[23] Y. Kumar and N. Singh, "An automatic spontaneous speech recognition system for punjabi language," in *Speech and Language Processing for Human-Machine Communications* (S. S. Agrawal, A. Devi, R. Wason, and P. Bansal, eds.), (Singapore), pp. 57–66, Springer Singapore, 2018.

[24] I. I. Mekki, *Automatic Speech Recognition in the French Language*. PhD thesis, 09 2020.

[25] P. Bhagath, S. Parisa, S. D. Reddy, and F. Banu, "An android based mobile spoken dialog system for telugu language to control smart appliances," in *2021 IEEE XXVIII International Conference on Electronics, Electrical Engineering and Computing (INTERCON)*, pp. 1–4, 2021.

[26] B. Parabattina, P. Chandra, V. Sharma, and P. K. Das, "Voice-controlled assistance for robot navigation using android-based mobile devices," in *2021 Sixth International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET)*, pp. 21–25, 2021.

[27] A. Abdulkareem, T. Somefun, O. Chinedum, A. Agbetuyi, and T. Shomefun, "Design and implementation of speech recognition system integrated with internet of things," *International Journal of Electrical and Computer Engineering*, vol. 11, pp. 1796–1803, 04 2021.

[28] D. Vazhenina, I. Kipyatkova, K. Markov, and A. Karpov, "State-of-the-art speech recognition technologies for russian language," 03 2012.

[29] L. Selvaraj and M. Devi, "Speech recognition system using enhanced mel frequency cepstral coefficient with windowing and framing method," *Cluster Computing*, vol. 22, 09 2019.

[30] P. Bhagath, S. Parihar, and P. K. Das, "Speech recognition for indian spoken languages towards automated home appliances," in *2021 2nd International Conference for Emerging Technology (INCET)*, pp. 1–5, 2021.

[31] C. P. Silva, N. Neto, A. Klautau, A. Adami, and I. Trancoso, "Speech recognition for brazilian portuguese using the spoltech and ogi-22 corpora," 09 2008.

[32] H. Ali, A. Jianwei, and K. Iqbal, "Automatic speech recognition of urdu digits with optimal classification approach," *International Journal of Computer Applications*, vol. 118, pp. 1–5, 05 2015.

[33] M. Shigenaga, Y. Sekiguchi, and C.-h. Lai, "Speech recognition system for spoken Japanese sentences," in *COLING 1980 Volume 1: The 8th International Conference on Computational Linguistics*, 1980.

[34] N. Neto, C. P. Silva, A. Klautau, and I. Trancoso, "Free tools and resources for brazilian portuguese speech recognition," *J. Braz. Comp. Soc.*, vol. 17, pp. 53–68, 03 2011.

[35] A. Nakamura, S. Matsunaga, T. Shimizu, M. Tonomura, and Y. Sagisaka, "Japanese speech databases for robust speech recognition," pp. 2199–2202, Dec. 1996. Proceedings of the 1996 International Conference on Spoken Language Processing, ICSLP. Part 1 (of 4) ; Conference date: 03-10-1996 Through 06-10-1996.

[36] A. Abdulkareem, T. Somefun, O. Chinedum, A. Agbetuyi, and T. Shomefun, "Design and implementation of speech recognition system integrated with internet of things," *International Journal of Electrical and Computer Engineering*, vol. 11, pp. 1796–1803, 04 2021.

[37] Y. Ma and M. Yi, "Russian speech recognition system design based on hmm," *International Conference on Logistics, Engineering, Management and Computer Science, LEMCS 2014*, 05 2014.

[38] S. Supriya and S. M. Handore, "Speech recognition using htk toolkit for marathi language," in *2017 IEEE International Conference on Power, Control, Signals and Instrumentation Engineering (ICPCSI)*, pp. 1591–1597, 2017.

[39] L. Selvaraj and M. Devi, "Speech recognition system using enhanced mel frequency cepstral coefficient with windowing and framing method," *Cluster Computing*, vol. 22, 09 2019.

[40] O. Iakushkin, G. Fedoseev, A. Shaleva, A. Degtyarev, and O. Sedova, "Russian-language speech recognition system based on deepspeech," 12 2018.

[41] A. Singh and V. Kadyan, "Automatic speech recognition system for tonal languages: State-of-the-art survey," *Archives of Computational Methods in Engineering*, vol. 28, 02 2020.

[42] S. Paulose, S. Nath, and S. K, "Marathi speech recognition," pp. 230–233, 08 2018.

[43] L. Hu and J. Jia, "Smart speech recognition system for chinese language learning enhancement," *Scientific Programming*, vol. 2022, pp. 1–11, 06 2022.

[44] S. Furui, "Speech recognition for japanese spoken language," pp. 122 – 126 vol.1, 05 1994.

[45] A. Karpov, K. Markov, I. Kipyatkova, D. Vazhenina, and A. Ronzhin, "Large vocabulary russian speech recognition using syntactico-statistical language modeling," *Speech Communication*, vol. 56, pp. 213–228, 2014.

[46] Y. Wang, R. Jia, C. T. Lam, K. Choi, K. Ng, X. Yang, and S. Im, *Automatic Speech Recognition for Portuguese with Small Data Set*, pp. 1–13. 01 2022.

TELUGU VAKYALU: Telugu Sentence Dataset For IoT Applications

***Abstract*—Spoken data sets are very critical components for speech research in various languages. The new developments in this area heavily relies on the availability of data sets. The unavailability of data is not prominent in well-developed spoken languages, but it seen in the case of low and under resourced languages. Indian languages are considered low-resource languages with respect to the accessibility of speech. IoT is a field of research where speech processing can contribute a large portion in developing the interfaces. This requires speech recognition frameworks that deal with the problems in the domain on IoT. This paper releases a speech corpus that can help researchers to develop spoken dialog or interfaces in an IoT environment. The detailed process of the dataset creation is discussed in the paper.**

***Index Terms*—IoT, Telugu Sentence Corpus, Speech Recognition, Under-resourced language**

I. Introduction

Speech recognition is an area that has wide applications in the domains of IoT, Human Machine Interaction, Brain Computer Interaction (BCI), etc. The research can contribute by developing application, datasets and frameworks. Though the research in speech processing has been started a few decades back, there is a huge scope in certain languages which can be called as low-resource languages. To continue research in these languages, we require a variety of datasets to deal with different problems. The spoken datasets comprises vowels, digits, words, sentences, emotions, etc. The datasets need to be prepared specific to the problems since a generic solutions cannot exist for all the problems. The availability of such datasets is very minimal for low-resource languages spoken in Asian and African continents. India a vast country with wide cultures and many languages spoken in different parts of the country. The constitution of India recognized 24 languages as scheduled languages spoken by the people living in 28 states and 8 union territories. Despite the fact that the languages are similar in terms of syntax and vocabulary, the contribution of automated spoken language comprehension is little for the majority of languages and IoT-related datasets. Telugu is spoken majorly in the states of Andhra Pradesh, Telangana, and Puducherry. The language has 3 main dialects and rich in its grammar and literature.

Speech recognition takes different forms depends in the platform that holds the recognition engine. One such an important field is mobile speech recognition. Though there were limitations faced in this area with respect to the computational competence, it has unfold a wide variety of opportunities to the researchers. The new mobile platforms are able to provide necessary computational power to place the speech engines. As an IoT component it gives computation power to control the appliances and different devices. In [1], the authors proposed a mobile based speech recognition system Indian accented English and Hindi languages for controlling home appliances. A very limited vocabulary was used to meet the requirements. A Telugu spoken sentence recognition for a small set of 10 speakers was built to control the household appliances [2]. However, the contribution towards developing Automatic Speech Recognition (ASR) systems and speaker recognition systems is minimal. This is due to insufficiency of datasets to address ASR problems in the Telugu language.

This paper contributes to the speech research community by providing a new spoken dataset for the Telugu language. The content of the paper is presented in different sections. In the Section II, we describe various works done in this direction. The procedure for developing the speech corpus is elaborated in Section III. Finally, the paper is concluded by giving a few future insights in Section IV.

II. Literature Survey

This section discusses the previous work of the spoken data sets reported in the literature.

Multilingual data sets are in high demand right now because language translation and dictionary creation are important applications in the field of voice processing. Researchers have been interested in producing data sets meticulously due to the high need for data sets in many languages. The procedure resulted in massive data sets being made available to the scholarly community. Narayan Chowdary and D. G. Rao gives an overall access to datasets that comprises a total of 13 scheduled Indian languages, gathered in varied locations across the country from a total of 5662 individuals of various age groups, with an overall size of more than 1552 hours [14]. In [14] LDC-IL was taken into account Telugu has three distinct variations, thus we gathered speech data from Telangana, Rayalaseema, and Coastal Andhra. The LDC-IL Telugu Speech data collection is made up of many datasets such as word lists, sentences, running texts, and date formats. Each speaker recorded these datasets, which were chosen at random from a source dataset. The speech is in .wav format, while the metadata is in .txt.

K. Samudravijaya developed a speaker independent continuous ASR for Hindi. Their system recognized spoken requests in Hindi in the context of a request for a train reservation. The speech sentence corpus contain 320 sentences and the vocabulary size was 161 words. and in [3] he was focusing on developing an automated speech recognition (ASR) system

Table I
Some Low Level Resource Languages

S. No.	Language	Speaking Population (in million)	Places where the language is widely spoken
1.	Awadhi	3.9	Majorly spoken in Uttar Pradesh
2.	Bhojpuri	50	regions of Bihar and Uttar Pradesh
3.	Telugu	85	Andra Pradesh, Telangana, Karnataka, Tamil Nadu, etc.
4.	Kannada	47	Karnataka, Andra Pradesh, Goa, Kerala
5.	Sanskrit	0.024	Karnataka, Madhya Pradesh , Odisha, Rajasthan
6.	Marathi	83	Region of Maharastra and Bihar
7.	Gujarati	46	Gujarat and Pakistan
8.	Tamil	64	Tamil nadu, Karnataka, Andhra Pradesh, Telangana, Northern and Eastern Srilanka
9.	Kashmiri	6.8	Kashmir Valley and Some regions of Jammu and Kashmir
10.	Urdu	70	Uttar Pradesh, Madhya Pradesh, Bihar, Pakistan, Kashmir
11.	Bangla	300	Bangladesh, India, Tripura, Assam
12.	Konkani	2.4	Goa, Karnataka, Mangalore, Maharashtra and some parts of Kerala, Gujarat
13.	Malayalam	34	Kerala, Lakshadweep, Andaman and Nicobar Islands, Puducherry, Karnataka, Tamil Nadu
14.	Hindi	400	Northern regions of India and some regions in southern India
15.	Santhali	7.9	Assam, Bihar and West Bengal
16.	Sindhi	30.26	Sindh region of Pakistan
17.	Oriya	35	Odisha, West Bengal , Chhattisgarh
18.	Nepali	16	Nepal, Bhutan, Himalayas
19.	Chinese	13.4	Hong Kong, China, Taiwan, Singapore, etc.
20.	Manipuri	1.76	Manipur, Tripura
21.	Bahasa Bugis	4.3	southern region of Sulawesi, Indonasia
22.	Minang Kabau	4.2	West Sumatera
23.	Punjabi	123	Punjab, Pakistan, England, Canada, United Arab Emirates, the United States, New Zealand, Italy, and the Netherlands.
24.	Saraiki	26	Pakistan
25.	Bolachi	8.8	Bolachistan
27.	Korean	79.3	North and South Korea and North-East part of China
28.	Dari	20.5	Pakistan, Khyber, Pakhtunkhwa
29.	Najdi Arabic	14.6	Iraq, Jordan
30.	Hejazi Arabic	10.3	Bahan, Jizan
31.	Malay	33.12	Brunei, Malaysia, Indonesia, East Timor, Singapore and southern Thailand.
32.	Mandarin	13.0	Hong Kong, Taiwan, Macao, Singapore, China, etc.
33.	Yue	86	Beijing, Shanghai, Guangdong and Guangxi.
34.	Indonesian	210	Indonesia

for the Marathi language. A total of 106 speakers independent Marathi isolated words were identified in this investigation. These distinct Marathi terms are used to instruct and assess. They constructed the voice corpus using isolated Marathi words said by persons of mixed gender.

R. K. Aggarwal and Mayank D created a Speech Recognition System Interface for Indian Languages [5]. Their data preparation efforts were concentrated on background noise reduction, pre emphasis, blocking, windowing, etc. The initial objective is to digitise analogue electrical signals, which is to transform them into a discretetime, discrete valued signal. This analogue to digital conversion technique consists of two steps: sampling and quantization.

Bengali Language Speech Recognition system was developed by Amit Kumar Das et al. [6]. There were 508 speakers in total in the entire sample. This dataset includes roughly 200,000 wav formatted audio files, but only 33000 audio files were used for our work. 80% of the 33000 data points were utilised for training, 90% for validation, and 10% for testing. The overall length of 33000 data sets was roughly 33 hours.

Table II
Speech datasets of various Indian languages

S. No.	Name of the dataset	Chronicle	Description	Languages enclosed
1.	Hindi Speech Dataset [3]	speech dataset for continuous ASR system in Hindi	320 sentences, vocabulary size of 161 words	Hindi
2.	IndicSpeech [4]	Text to speech dataset	9916 Hindi words and 19954 Malayalam words	Hindi and Malayalam
3.	IndianSpeech dataset [5]	ASR dataset	-	Indian Languages
4.	Bengali Speech Dataset [6]	Audio books	508 speaker utterances of 33 hours data	Bengali
5.	Bengali Data set [7]	Anandabazar	19,640 unique words	Bengali
6.	Odia Speech Dataset [8]		104 Speakers utterances	Odia
7.	Agmarknet Dataset[9]	Agriculture data that includes different district names of Karnataka	35557 speech utterances	Kannada
8.	Assamese speech Dataset [10]	Development of assamese continuous speech recognition systems	2777 unique words and 1000 unique sentences	Assamese
9.	Kannada Dataset [11]	Noise dataset that contains Kannada spoken units	46 phonemes	Kannada
10.	Kannadaspeech corpus [12]	Robust continuous kannada speech recognition using kaldi toolkit	25 isolated words, 600 continuous sentences, 20 hours contribution of 100 speakers	Kannada
11.	Marathi Data set [13]	Marathi speech dataset for ASR	28,420 isolated words, 17,470 continuous sentences and from 500 different	Marathi
12.	scheduled Language Dataset [14]	Speech datasets of under resourced languages of north-east india	5662 speaker utterances of 1552 hours	13 schedules Indian Languages
13.	Telugu digit dataset [15]	Telugu spoken digits corpora for ASR	10 distinct words , 9.8 hours of data	Telugu
14.	T120 corpus [16]	Spoken digits corpora	6506 files, 16 individual utterances of 20 individual words	English
15.	Indonesian Speech Dataset [17]	spontaneous sentences	22 speakers utterances , 13 hour 25 minutes duration	Indonesian
16.	TI46 speech corpus [18]	46 word vocabulary	digits from 0 to 9, alphabets from a to z and some basic commands like start, stop, yes, go, help, no, enter, etc.	English

Audio dataset material was transformed into wav files. The sampling rate was 16000 Hz, and there was just one channel used. Diacritic characters were present in the dataset. As a result, breaking down Bengali audio files is more difficult. Biswajit Das and Sandipan Mandal also created a Bengali voice recognition system, selecting 47 Bengali phonemes [3]. The text corpus has 7500 distinct phrases and 19640 distinct words. 70 male and 40 female speakers recorded sentences. Sentences are gathered from different domains of Anandabazar text corpus.

Biswajit Karan et al. developed an ASR based up on Odia language [8]. They collected data from 104 speakers . During data collecting, speakers of various dialects and ages were present. For 30 districts and 5 phonetically rich phrases, each speaker's voice is captured.

Barsha Deka et al. provided an overview of an ongoing attempt to create speech corpora for under resourced languages in North-east India, especially Assamese and Nepali and even Bengali [19]. The voice corpora are being developed for the development of an Assamese ASR system as well as a Language Identification system. The Assamese text corpus consists of 1000 phrases gathered from various sources. The Assamese speech collection now comprises around 5000 utterances from 27 native speakers.

S. R. Nirmala, Barsha Deka and K. Samudravijaya developed a continuous Speech Recognition system for Assamese Language [10]. The sentences comprising the text corpus were gathered from many sources. The selected sentences ranged

in length from 5 to 10 words. This work's text corpus has 1000 unique continuous sentences made up of 2777 unique words. The collection of 1000 sentences was divided into 50 subsets, each with 20 sentences. Each subset has a one-digit sequence, four proverbs, and fifteen phrases chosen at random from diverse sources in the proportions given.

Kannada Langugae based speech recognition system was developed by Sharada C. Sajjan and Vijaya C [11], where it contains 46 phonemes, 12 of which constitute vowels and 34 of which constitute consonants. M.Swamy developed an ASR for recognizing the continuous Kannada speech sentences [12]. It comprises of isolated numbers, 25 isolated wods, and 600 spontaneous and continuously read utterances. The dataset consisted of 20-hour contributions from 100 speakers. Recordings were made in common room setting in the presence of different noises from the nearby road, which was positioned 5 metres away. Matlab R2019b software is used to record at 16KHz and 16-bit resolution. Mahadevaswamy Shanthamallappa and D. J. Ravi also developed an ASR for Kannada language where Kannada speech database and English speech database were created by 100 native Kannada male and female speakers [20]. Gaikwad et al. [13] constructed a Marathi language voice corpus for vowels and consonants. The whole data set includes 28,420 isolated words and 17,470 phrases from 500 different speakers.

Dandy Arif Rahman and Dessi Puji Lestari developed an Indonesian spontaneous Speech recognition system [17]. The authors collected random spontaneous data from Youtube podcast recordings. The tape was downloded and it is labeled manually. The data was collected from 22 speakers and it has 13 hour 25 minutes duration. The data was gathered from 20 speakers having almost 14 hours duration. S.I. Safie developed a spoken digit recognition system using CNN [16]. The dataset contains a total of 6505 files in which 1931 recordings from total is used for training purpose. The corpus comprises recordings of 16 people utterances of 20 single spoken syllables in which 8 of them are male. Mark Liberman et al. prepared TI46 dataset. In the corpus, the vocabulary covers 46 different words that consists of digits ranging from 0 to 9, alphabets from a to z and movement commands such as start, stop, go, no, help, enter, yes, etc [18]. Table II summarizes the data sets currently accessible for various languages spoken in Asian region.

In this paper, we are releasing a spoken Telugu sentence dataset useful for developing IoT based speech recognition systems. The complete procedure involved in the data set creation is explained in the next section.

III. Data Set Collection

As previously said, the dataset is critical in the construction of an ASR system. The dataset collection involves several phases which starts with the training procedure. Initially, appropriate speakers are to be selected who are the native speakers of the language. Secondly, the speakers to be trained by educating them with the recording procedure. Next, the recorded data is to be verified by validating the labels and content. This process can be imagined as depicted in the Figure 1.

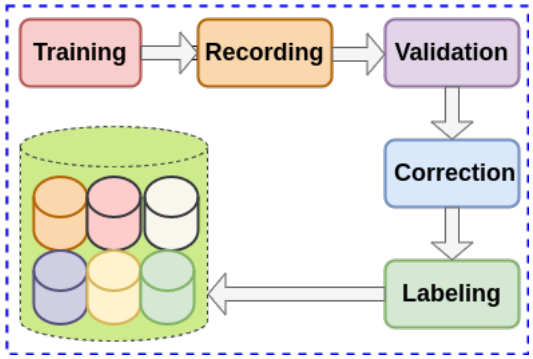


Figure 1. Process of Dataset Creation

A. Training the speakers

The data was collected from native Telugu speakers belonging to different districts of Andhra Pradesh and Telangana. This group includes students, homemakers, and employed professionals. The age of the speakers vary from 20 to 45 years including both the genders (male, female). Irrespective of their background, all the speakers underwent through a training procedure. During the training, the speakers were acquainted with a mobile recording application. This application is an Android application that allows to specify the recording parameters such as number of channels, sampling rate, bits per sample, etc. A screenshot of this application during the recording process is shown in Figure 2. In the training phase, the speakers were educated with to maintain the uniformity while recording the data. Figure 3 shows the speakers while recording the data in well-equipped computer laboratory.

B. Recording Procedure

The recording process was carried out in a quiet environment. This is a crucial step where a speaker needs to select the appropriate parameters required for creating the **.wav** files. The parameters used in the recording environment are as follows:

1) Channel type
2) Sampling rate

The channel type defines the number of channels used in the recording process. This is usually mono or stereo. For speech recognition experiments, mono channel is generally preferred. The present dataset was prepared with mono channel. The second parameter is the sampling rate that has been decided by the fact that humans hearing frequency ranges from 16Hz to 20kHz. The sampling rate of 16kHz frequency is suggested by Nyquist rate which means the sampling rate must be higher than twice the maximum required frequency. It is understood from the studies that 8kHz is the frequency at which speech can be perceived. Therefore, many speech corporas use sampling rate as 16kHz.

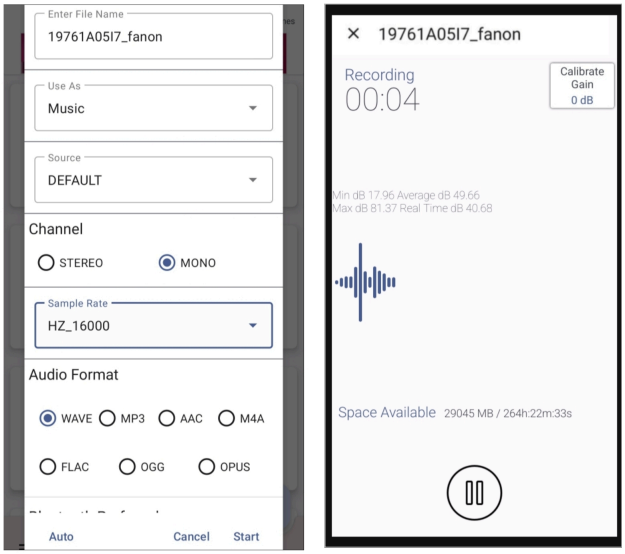


Figure 2. Screenshots of the Recording Module process

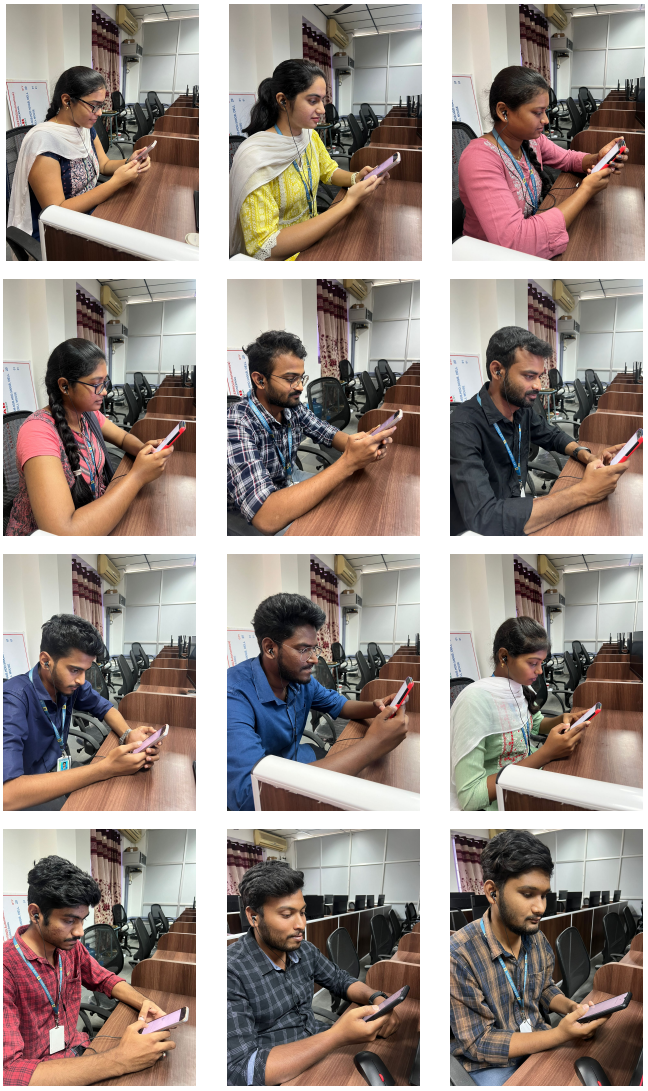


Figure 3. Students recording the speech data

C. Validation

The second phase is to validate the recorded data in which the issues are identified and addressed which is tedious job. In general, each recorded sample has to be verified to ensure the parameters discussed in the previous section. To recognize the mistakes, a Python script was used to extract the sampling rate and channel type from the header part of each file. The log data of the script was manually verified to see the mistakes in the spoken utterances. This step is followed by data correction that is elaborated in the next subsection.

D. Correction

The correction of data usually involves re-recording the samples with required parameters. This is a common mistake done by speaker's while they overlook the parameters and realized at the end. Though re-recording is a difficult task, this is required to achieve the required number of samples.

E. Sentence Choice

The current data set is intended to be used in the development of recognition algorithms for sentences in an IoT environment. Accordingly the keywords are picked to meet the requirements of the applications. In home automation, the targeted recognition systems need to deal with the appliances such as bulb, fan, TV and air conditioners. To control these devices, the equivalent words of Telugu must be used. The Telugu speakers use the English equivalent words to refer these appliances. There are two activities in controlling the appliances i.e. switching on and switching off. The corresponding words for 'switching on' and 'switching off' are different for controlling each device. For that reason, a variety of controlling words are used in the dataset. The sentences associated with each action are listed in Table III and IV. The table gives Telugu sentence and the meaning of the same in English. The count column is the total number of sentences in the data set. The last column gives the count of the words in each sentence. Each sentence used in the corpus contains 2 words. Each sentence is recorded for 960 times to maintain the uniformity across the dataset that can help to avoid the class imbalance problem [21]. Each sentence contains 2 different words and the individual words consists of vowels and consonants. The presence of these phonetic units will help to design robust recognition systems.

F. Sentence Labeling

The spoken sentence corpus is released with the name "TELUGU VAKYALU", which translates to "Telugu sentences". The entire dataset is organized into 48 folders consisting of individual speaker's data. This folder is labeled with the **Speaker ID**. This folder accommodates 10 different folders in which each folder is named against the sentence such as "acoff", "acon", etc. This folder contains the speech

Table III
Data Set Description - I

Sentence Number	Sentence	English Description	Count	No. of words
1	ఫ్యాన్ ఆన్ చేయండి	Switch on the Fan	960	2
2	ఫ్యాన్ ఆపండి	Switch off the fan	960	2
3	బల్బ్ వెలిగించండి	Turn Bulb on	960	2
4	బల్బ్ ఆపండి	Turn Bulb off	960	2
5	లైట్ వెలిగించండి	Turn Light on	960	2
6	లైట్ ఆపండి	Turn Light off	960	2
7	టీవీ పెట్టండి	Switch on the TV	960	2
8	టీవీ కట్టేయండి	Switch off the TV	960	2
9	ఏసీ వేయండి	Turn AC on	960	2
10	ఏసీ ఆపండి	Turn off the AC	960	2

Table IV
Data Set Description - II

Sentence Number	Sentence	English Description	Count	No. of words
1	ఒకటవ బల్బ్ వెలిగించండి	Turn on Bulb 1	1000	3
2	ఒకటవ బల్బ్ ఆపండి	Turn off Bulb 1	1000	3
3	రెండవ బల్బ్ వెలిగించండి	Turn on Bulb 2	1000	3
4	రెండవ బల్బ్ ఆపండి	Turn off Bulb 2	1000	3
5	మూడవ బల్బ్ వెలిగించండి	Turn on Bulb 3	1000	3
6	మూడవ బల్బ్ ఆపండి	Turn off Bulb 3	1000	3
7	ఒకటవ ఫ్యాన్ ఆన్చండి	Switch on the 1st fan	1000	3
8	ఒకటవ ఫ్యాన్ ఆపండి	Switch off the 1st fan	1000	3
9	రెండవ ఫ్యాన్ ఆన్చండి	Switch on the 2nd fan	1000	3
10	రెండవ ఫ్యాన్ ఆపండి	Switch off the 2nd fan	1000	3

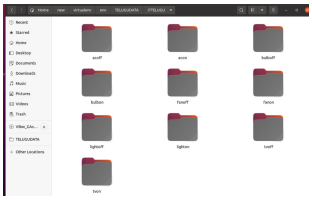


Figure 4. Folder of a single speaker

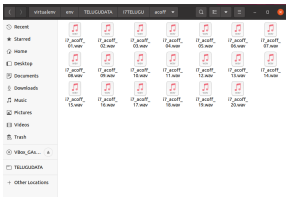


Figure 5. Folder Organization

Table V
Speech Corpus Description - I

Attributes	Values
Number of audible sentences in the corpus	10
Total number of speakers data	48
Sessions used in the process	60
Absolute number of utterances	9,600
Total duration of all the files	290 mins (4.8 hours)

data of the sentence. A sample folder structure is shown in Figure 4. Figure 5 gives the organization of each sentence folder.

IV. Future Work and Conclusions

The present paper discusses the process of speech corpus creation for Telugu language. The dataset will be publicly available for the research purpose. It can be used to develop speech recognition frameworks and models for the IoT environment. Further, detailed studies of phonetic units can be conducted in Telugu language that can improve the state-of-the-art in new directions. This can also help in language identification tasks that includes Telugu as one of the languages. Though the number of speakers involved is less, this is essentially an useful corpus towards the low-resource languages development.

References

[1] P. Bhagath, S. Parisa, S. D. Reddy, and F. Banu, "An android based mobile spoken dialog system for telugu language to control smart appliances," in *2021 IEEE XXVIII International Conference on Electronics, Electrical Engineering and Computing (INTERCON)*, pp. 1–4, 2021.

Table VI
Speech Corpus Description - II

Attributes	Values
Number of audible sentences in the corpus	10
Total number of speakers data	50
Sessions used in the process	60
Absolute number of utterances	10,000
Total duration of all the files	320 mins (5.3 hours)

[2] P. Bhagath, S. Parihar, and P. K. Das, "Speech recognition for indian spoken languages towards automated home appliances," in *2021 2nd International Conference for Emerging Technology (INCET)*, pp. 1–5, 2021.

[3] B. Das, S. Mandal, and P. Mitra, "Bengali speech corpus for continuous automatic speech recognition system," *2011 International Conference on Speech Database and Assessments, Oriental COCOSDA 2011 - Proceedings*, 10 2011.

[4] N. Srivastava, R. Mukhopadhyay, P. K R, and C. V. Jawahar, "IndicSpeech: Text-to-speech corpus for Indian languages," in *Proceedings of The 12th Language Resources and Evaluation Conference*, (Marseille, France), pp. 6417–6422, European Language Resources Association, May 2020.

[5] R. Aggarwal and M. Dave, "Implementing a speech recognition system interface for indian languages," 01 2008.

[6] J. Islam, M. Mubassira, M. Islam, and A. Das, "A speech recognition system for bengali language using recurrent neural network," pp. 73–76, 02 2019.

[7] B. Das, S. Mandal, and P. Mitra, "Bengali speech corpus for continuous auutomatic speech recognition system," in *2011 International Conference on Speech Database and Assessments (Oriental COCOSDA)*, pp. 51–55, 2011.

[8] B. Karan, J. Sahoo, and P. Sahu, "Automatic speech recognition based odia system," pp. 353–356, 12 2015.

[9] Y. G. Thimmaraja and H. S. Jayanna, "Creating language and acoustic models using kaldi to build an automatic speech recognition system for kannada language," in *2017 2nd IEEE International Conference on Recent Trends in Electronics, Information Communication Technology (RTEICT)*, pp. 161–165, 2017.

[10] B. Deka, S. Nirmala, and S. K., "Development of assamese continuous speech recognition system," pp. 215–219, 08 2018.

[11] S. C. Sajjan and V. C, "Continuous speech recognition of kannada language using triphone modeling," in *2016 International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET)*, pp. 451–455, 2016.

[12] M. Swamy, "Robust automatic speech recognition system for kannada speech sentences in the presence of noise," 2022.

[13] S. Gaikwad, B. Gawali, and S. Mehrotra, "Creation of marathi speech corpus for automatic speech recognition," in *2013 International Conference Oriental COCOSDA held jointly with 2013 Conference on Asian Spoken Language Research and Evaluation (O-COCOSDA/CASLRE)*, pp. 1–5, 2013.

[14] A. Thakallapelli, S. Ghosh, and S. Kamalasadan, "Real-time frequency based reduced order modeling of large power grid," in *2016 IEEE Power and Energy Society General Meeting (PESGM)*, pp. 1–5, 2016.

[15] P. Bhagath, M. Pullagura, P. K. Das, V. K. Yandra, and S. S. Thetla, "Telugu ankelu: A telugu spoken digits corpora for mobile speech recognition," in *2022 12th International Conference on Pattern Recognition Systems (ICPRS)*, pp. 1–6, 2022.

[16] S. Safie, "Spoken digit recognition using convolutional neural network," in *2022 Applied Informatics International Conference (AiIC)*, pp. 24–27, 2022.

[17] D. A. Rahman and D. P. Lestari, "Indonesian spontaneous speech recognition system using deep neural networks," in *2020 7th International Conference on Advance Informatics: Concepts, Theory and Applications (ICAICTA)*, pp. 1–3, 2020.

[18] M. Liberman, "Ti46 speech corpus," *Linguistic data consortium*, 1993.

[19] B. Deka, J. Chakraborty, A. Dey, S. Nath, P. Sarmah, S. Nirmala, and K. Samudravijaya, "Speech corpora of under resourced languages of north-east india," 05 2018.

[20] M. Shanthamallappa and D. Ravi, "Robust continuous kannada speech recognition using kaldi toolkit," *Materials Today: Proceedings*, 03 2021.

[21] A. N. Tarekegn, M. Giacobini, and K. Michalak, "A review of methods for imbalanced multi-label classification," *Pattern Recognition*, vol. 118, p. 107965, 2021.

Gmail

ICPRS-23: New Submission Received. Contribution ID: 43

1 message

Sergio A Velastin <sergio.velastin@gmail.com> Fri, Feb 24, 2023 at 12:26 PM
To: drbhagath.lbrce@gmail.com

Dear Dr. BHAGATH PARABATTINA,

We have received your submission. Thank you.

Contribution Details
==========
ID : 43
Title : TELUGU VAKYALU: Telugu Sentence Dataset For IoT Applications
Author(s) : Parabattina Bhagath, Vanga Lasya, P Dhyeya

Uploaded Files
==========
ICPRS_2023_TELUGU_SENTENCE_DATABASE_(4).pdf
Last Upload: 24/Feb/2023

With best regards,
Your ICPRS-23 organizers.

--
13th International Conference on Pattern Recognition Systems
http://s836450039.websitehome.co.uk/conftool_icprs23/