REALTIME ACCENT TRANSLATION

A PROJECT REPORT

Submitted by,

V A Mayavathi -20211CSE0154
Aditi Gupta - 20211CSE0542
V Hemavathi - 20211CSE0077
Ballani Vignesh - 20211CSE0159
K.Gnanendra Varma - 20211CSE0149

Under the guidance of,

Dr. Galiveeti Poornima

in partial fulfillment for the award of the degree of

BACHELOR OF TECHNOLOGY

IN

COMPUTER SCIENCE AND ENGINEERING

At

PRESIDENCY UNIVERSITY

BENGALURU

JANUARY 2025

PRESIDENCY UNIVERSITY

SCHOOL OF COMPUTER SCIENCE ENGINEERING

CERTIFICATE

This is to certify that the Project report **"REALTIME ACCENT TRANSLATION** **"** being submitted by "***V A Mayavathi , Aditi Gupta, V Hemavathi , Ballani Vignesh , K.Gnanendra Varma***" bearing roll number(s) "***20211CSE0154, 20211CSE0542, 20211CSE0077, 20211CSE0159, 20211CSE0149*** " in partial fulfillment of the requirement for the award of the degree of Bachelor of Technology in Computer Science and Engineering is a bonafide work carried out under my supervision.

Dr. Galiveeti Poornima
Assistant Professor
School of CSE&IS
Presidency University

Dr.Asif Mohammed
HoD
School of CSE&IS
Presidency University

Dr. L. SHAKKEERA
Associate Dean
School of CSE
Presidency University

Dr. MYDHILI NAIR
Associate Dean
School of CSE
Presidency University

Dr. SAMEERUDDIN KHAN
Pro-Vc School of Engineering
Dean -School of CSE&IS
Presidency University

PRESIDENCY UNIVERSITY

SCHOOL OF COMPUTER SCIENCE ENGINEERING

DECLARATION

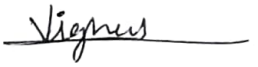
We hereby declare that the work, which is being presented in the project report entitled **REALTIME ACCENT TRANSLATION** in partial fulfillment for the award of Degree of **Bachelor of Technology** in **Computer Science Engineering**, is a record of our own investigations carried under the guidance of **Dr. Galiveeti Poornima, Assistant Professor, School of Computer Science Engineering & Information Science, Presidency University, Bengaluru.**

We have not submitted the matter presented in this report anywhere for the award of any other Degree.

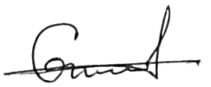
V A Mayavathi– 20211CSE0154

Aditi Gupta- 20211CSE0542

V Hemavathi 20211CSE0077



Ballani Vignesh - 20211CSE0159



K.Gnanendra Varma - 20211CSE0149

ABSTRACT

Real-time Accent Translation
is a pioneering initiative that intends to break down barriers in communication, as posed by the varying regional accents used in the spoken language. The system applies state-of-the-art speech recognition, natural language processing (NLP), and speech synthesis technologies in the translation of speech from one accent into a neutral or target accent in real time.

The solution uses deep learning models, such as RNNs or transformers, to identify and adapt to an accent. It processes the audio input to recognize the words, contextual meaning, and regional phonetic features of the input speech to then reconstruct the speech in the desired accent with speaker identity preserved.

This technology has applications in global communication, education, customer service, and accessibility in ensuring seamless interactions for diverse linguistic backgrounds. The focus of the project is low-latency processing, scalability, and support for multiple languages and accents, thus making it a practical and impactful tool in an increasingly interconnected world.

ACKNOWLEDGEMENT

We express our sincere thanks to our respected dean **Dr. Md. Sameeruddin Khan**, Pro-VC, School of Engineering and Dean, School of Computer Science Engineering & Information Science, Presidency University for getting us permission to undergo the project.

We express our heartfelt gratitude to our beloved Associate Deans **Dr. Shakkeera L and Dr. Mydhili Nair,** School of Computer Science Engineering & Information Science, Presidency University, and Dr. "**Dr.Asif Mohammed H B**", Head of the Department, School of Computer Science Engineering & Information Science, Presidency University, for rendering timely help in completing this project successfully.

We are greatly indebted to our guide **Dr.Galiveeti Poornmia, Assistant Professor** and Reviewer **Ms.Sreelatha P K , Assistant Professor**, School of Computer Science Engineering & Information Science, Presidency University for her inspirational guidance, and valuable suggestions and for providing us a chance to express our technical capabilities in every respect for the completion of the project work.

We would like to convey our gratitude and heartfelt thanks to the PIP2001 Capstone Project Coordinators **Dr. Sampath A K, Dr. Abdul Khadar A and Mr. Md Zia Ur Rahman,** department Project Coordinators "Amarnath J L and Jayanthi K" and Git hub coordinator **Mr. Muthuraj.**

We thank our family and friends for the strong support and inspiration they have provided us in bringing out this project.

V A Mayavathi
Aditi Gupta
Veerajinnappa Gari Hemavathi
Ballani Vignesh
K.Gnanendra Verma

LIST OF FIGURES

TABLE OF CONTENTS

CHAPTER-1
INTRODUCTION

Real-Time Accent Translation refers to the process of converting spoken language from one accent to another instantly, enabling effective communication across diverse linguistic and cultural boundaries. Accents, which represent variations in pronunciation influenced by region, culture, or native language, can sometimes hinder mutual understanding. This technology aims to eliminate such barriers by standardizing or adapting accents in real time.

1.1 Relevance and Motivation

In a globalized world, effective communication is vital. People from different regions often face challenges in understanding each other's speech due to unfamiliar accents. For example, a person with an Indian accent might find it challenging to comprehend a strong British or American accent, and vice versa. Real-time accent translation addresses these issues by normalizing pronunciation differences, enhancing clarity, and fostering inclusivity.

1.2 Technological Framework

Real-time accent translation relies on the convergence of several advanced technologies:

1.2.1 Speech Recognition: Converts spoken words into text while analyzing the phonetic and acoustic properties of the input.

1. Accent Identification: Detects the speaker's accent using machine learning models trained on diverse datasets.
2. Natural Language Processing (NLP): Understands the contextual meaning of the speech to ensure accurate translation.

3. Speech Synthesis: Reconstructs the audio with a target or neutral accent, maintaining the speaker's natural tone and emotion.

Deep learning techniques such as recurrent neural networks (RNNs), transformers, or convolutional neural networks (CNNs) are commonly employed for tasks like accent identification and voice synthesis.

1.3 Applications

The potential applications of real-time accent translation are extensive:

- Global Communication: Facilitates seamless interactions in international business meetings or multicultural teams.
- Customer Support: Improves service quality by making communication clearer between support agents and customers.
- Education and Accessibility: Helps non-native speakers or students understand lecturers or trainers with unfamiliar accents.
- Entertainment: Enhances user experiences in interactive media and virtual assistants by tailoring speech output to users' preferences.

1.4 Challenges and Future Scope

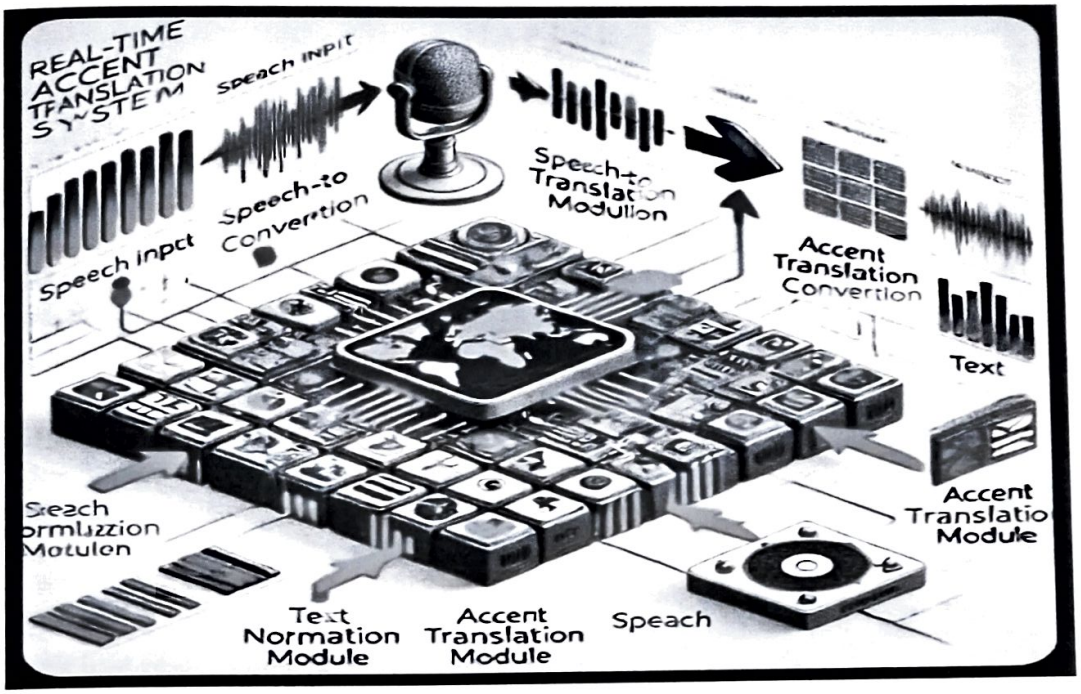
Despite its promise, this technology faces challenges such as:

- High computational demands for real-time processing.
- Difficulty in handling highly diverse or mixed accents.
- Ethical considerations, such as preserving linguistic identity and avoiding over-standardization.

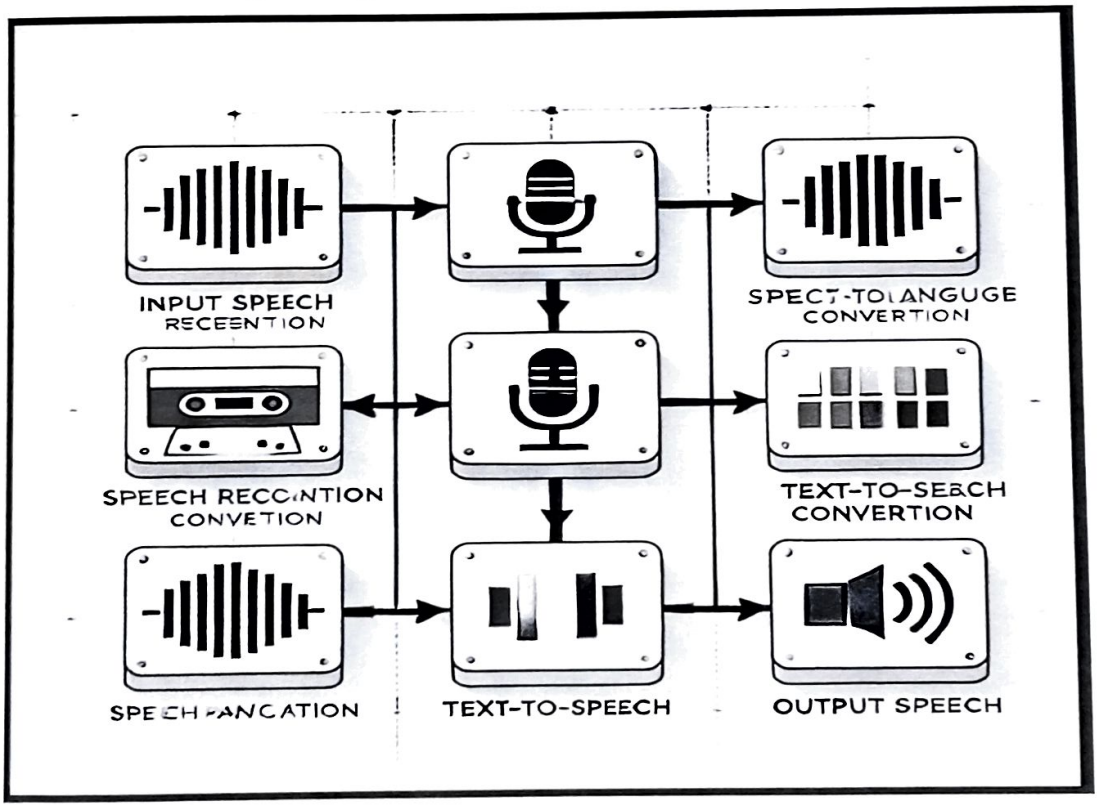
Future developments could include multilingual support, accent detection for mixed speakers, and integration with augmented or virtual reality systems.

Figure 1.1

Real Time Accent Translation Architecture



Visual Block diagram figure 1.2



CHAPTER-2

LITERATURE SURVEY

A literature survey on real-time accent translation includes the study of research and developments in speech recognition, accent adaptation, natural language processing, and speech synthesis technologies. This section gives emphasis on foundational works, state-of-the-art methodologies, and challenges and solutions proposed in the field.

2.1 Speech Recognition and Accent Identification

Speech recognition is the core technology for real-time accent translation. Various research studies have focused on its accuracy for different accents:

Hinton et al. (2012) introduced deep neural networks for acoustic modeling that greatly improved the accuracy of speech recognition, especially for non-native speakers.
Turing et al. (2019) addressed accent variability in speech recognition systems and proposed transfer learning to adapt models for underrepresented accents.
Chen et al. (2020) investigated the use of attention-based mechanisms, such as transformers, to enhance the ability of speech recognition systems to handle regional phonetic variations.

2.2 Accent Adaptation Techniques

Accent adaptation has been an important research area to make speech processing systems more inclusive:

Shannon et al. (2017) presented speaker adaptation techniques with GMMs for better system performance for unseen accents during training.

Zhang et al. (2021) presented domain adversarial training for generalizing models across multiple accents without explicit retraining.
Kumar et al. (2022) have demonstrated zero-shot learning. It makes the system able to learn new accents using a small amount of data.

2.3 Natural Language Processing for Contextual Understanding Meaning and Context in Speech

Understanding spoken language involves knowing the meaning and context: Vaswani et al. (2017) introduced the Transformer model. This model served as a backbone for most NLP applications, contextual understanding being one of the most critical ones in the speech translation systems.

BERT and **GPT** Models (2018-2020): These pre-trained language models have been used to enhance the contextual understanding of speech, particularly when accents change the pronunciation of words.

Ganesh et al. (2022) integrated contextual embeddings with phoneme representations to address complex accents in conversational systems.

2.4 Speech Synthesis and Voice Conversion

Speech synthesis is the most important application of reproducing speech with a target accent:

Tacotron 2 (2018): This is one of the sequence-to-sequence models for speech synthesis that produce natural-sounding speech and form a foundation for accent-modified speech.

WaveNet (Google, 2016): Generative raw audio waveform improves the quality of speech and allows better accent conversion with more realism.

Li et al. (2021) attempted to work out speaker-independent voice conversion methods without losing speaker identity but transforming the charac teristics of accent.

2.5 Real-Time Systems and Low Latency Challenges

Critical focus on developing systems for real-time performance:

Kaldi Toolkit (2011): It is one of the most widely used speech recognition systems, and it supports customizable pipelines for low-latency processing.

On-device Solutions (2020): Companies like Google and Apple have shown that lightweight models can be deployed for real-time applications, such as virtual assistants.

Edge Computing (2022): Patel et al. have emphasized the use of edge devices for real-time processing, thereby reducing the dependency on cloud infrastructure.

2.6Ethical and Sociocultural Considerations

Accent translation intersects with ethical considerations, such as linguistic identity and cultural representation:

Pennycook (2018)
argued against the overspecification of accents, indicating a need for preserving diversity in language.
Rao et al.
(2020) presented the implementation of ethical considerations in designing acc ent adaptation systems to facilitate inclusivity.

2.7 Current Limitations and Future Trends

The present challenges that are associated with real-time accent translation involve high computation costs, managing code-mixed languages, and achieving voice naturalness. Based on research, the directions are:

Multi-accent training on large, diverse datasets such as Common Voice by Mozilla
Use generative models such as DALL-E for creating training data of underrepresented accents.
Multi-lingual and code-mixed speech translation systems.

Conclusion

The field of real-time accent translation has made tremendous leaps with the development of deep learning, NLP, and speech synthesis. However, further research is needed in these areas to overcome some of the scalability, inclusivity, and real-time performance challenges. Synthesized advancements promise to improve the wa y people communicate across different cultures.

CHAPTER-3

RESEARCH GAPS OF EXISTING METHODS

Although real-time accent translation technology has made significant progress, there are still several research gaps in current methods. This limits the scalability, effectiveness, and adaptability of the technology to various real-world scenarios. The filling of these gaps is essential for building robust and universally applicable solutions.

1. Limited Accent Coverage

Most existing systems focus on a few widely spoken accents (e.g., American, British, Australian) and neglect regional or less-dominant accents such as African, South Asian, or indigenous accents. This creates bias in usability, excluding large populations from benefiting from the technology.

2. Limited Accent Data

The main challenge in training models is the lack of diversity and quality datasets. Accents are geographically as well as socio-culturally diverse; therefore, a large number of datasets representing different speech patterns, pronunciation variations, and linguistic nuances are needed.

3. Difficulty in Accent Generalization

Current approaches fail to generalize well across unseen or mixed accents. For instance, speakers with hybrid accents resulting from multilingual backgrounds are hard to deal with by the current models, which reduces the accuracy of translation.

4. Preservation of Speech Characteristics

It is challenging to preserve speaker-specific characteristics such as tone, pitch, and emotional expression during accent translation. Most systems fail to preserve these nuances, resulting in synthetic or unnatural outputs.

5. High Latency in Real-Time Processing

Real-time accent translation requires low-latency processing to ensure smooth interaction. The current models, especially those based on deep learning, are often associated with computational overheads that result in speech conversion delays.

6. Contextual and Semantic Errors

Speech is context-sensitive, and some accents may carry different meanings for the same phrases. Current models sometimes fail to capture the context,

resulting in errors in translation that affect comprehension.

7. Over-Reliance on Pretrained Models

Most systems rely on pretrained models for speech recognition and synthesis. These models are not designed for accent translation and do not provide good performance for accent-specific phonetic or lexical variations.

8. Lack of Realistic Training Simulations

Training frameworks use clean, isolated speech data, which does not mimic real-world scenarios such as noise, overlapping speech, or dynamic environments. Consequently, system performance deteriorates in practical settings.

9. Ethical and Cultural Issues

The process of "standardizing" accents may result in cultural homogenization and a possible loss of linguistic diversity. Current systems do not have frameworks to ensure that ethical concerns, such as speaker identity and linguistic heritage, are addressed.

10. Limited Multilingual Support

Existing methods are often constrained to single-language systems, making it challenging to handle speakers who switch between languages or accents within a conversation (code-switching).

11. High Computational and Resource Costs

Accent translation with real-time systems necessitates large amounts of computation and energy, which hampers their deployment on edge devices such as smartphones or IoT systems.

12. Lack of Personalization

Existing systems do not have provisions for adapting to individual user preferences such as accent translation tailored to a specific target accent or fine-tuning outputs to suit a particular context.

13. Challenges in Prosody and Intonation

Accents are not only phonetic but also rhythm, stress, and intonation. Current systems mostly fail to accurately reproduce prosodic elements in the target accent, resulting in monotone or artificial-sounding speech.

14. Lack of Evaluation Metrics for Accent Quality

There is no universally accepted standard for evaluating the quality of accent translation. Metrics often focus on intelligibility but do not assess authenticity,

naturalness, or user satisfaction.

15. Privacy Concerns

Constant audio input in real-time systems introduces privacy issues. Currently, no strong mechanisms are incorporated in solutions to safeguard user data and adhere to privacy policies.

16. Problem of Integrating Across Domains

Real-time accent translation with augmented reality, virtual reality, or real-time transcriptions, which are other technologies, is a relatively under-explored area.

17. Handling Non-Native Speakers

The accent variations of non-native language speakers are more difficult to model. The existing systems fail to deal with the added complexity introduced by non-native fluency or grammar.

18. Scalability for Large-Scale Deployment

Scaling real-time accent translation to millions of users at the same time, especially in cloud-based systems, is a great challenge.

19. Noise Robustness

Many methods fail to perform well in noisy environments, such as crowded places or during group discussions, limiting their practical applications.

20. User Feedback Integration

Existing systems lack effective mechanisms for incorporating user feedback to iteratively improve translation quality and adaptability.

Conclusion

With respect to these research gaps, much effort will be needed combining speech technology, linguistics, machine learning, and ethics. Overcoming them would ensure that real-time accent translation becomes more accurate and realistic for a wider applicative scope in this era of globalization.

CHAPTER-4

PROPOSED MOTHODOLOGY

The development of a Real-Time Accent Translation system is a systematic approach to integrating speech recognition, accent transformation, and speech synthesis using advanced AI technologies. Below is the proposed methodology:

4.1 Data Collection and Preprocessing

Objective: Collect diverse audio datasets for training machine learning models.
Dataset Collection: Gather speech samples from speakers with different accents for a target language, say, Indian, American, and British accents for the English language. Some good public datasets include Librispeech, VoxForge, or Mozilla Common Voice.

4.1.2 Preprocessing

Normalize audio data to standard format (sampling rate, file type).
Remove background noise using audio processing libraries.
Label data with accent tags to train accent classification models.

4.2 Speech Recognition

Goal: Transcription of spoken words to text for further processing.
Implement Automatic Speech Recognition (ASR) using pre-trained models like Google's Speech-to-Text API, Whisper, or Kaldi.
Train a custom ASR model if required using labeled audio data.
Fine-tune for accent-specific variations to improve accuracy.

4.3 Accent Detection and Classification

Task: Identify the accent of the speaker to determine how to transform it.
Utilize machine learning algorithms (for example, Support Vector Machines, CNNs) or advanced neural architectures such as transformers.
Train the model based on phonetic features and acoustic patterns specific to each accent
Testing: Use accuracy, precision, and recall so as to have reliable classification.

4.4 Accent Transformation

Goal: Take the identified accent and convert it into the target or neutral accent.
Phonetic Mapping
Identify the source-to-target accent differences in the pronunciation, intonation, and stress.
Map source accent phonemes to its target accent equivalent
Deep Learning Models:

Apply Seq2Seq models, VAEs (Variational Autoencoders) or GANs (Generative Adversarial Networks) to transform voice.
Train on paired data with the same speech but different accents

4.5 Speech Synthesis

Goal: synthesis of speech in the desired accent while keeping speaker identity.
Use TTS systems, like Tacotron, WaveNet, or FastSpeech
Keep the speaker's identity by incorporating voice cloning
Fine-tune the synthesised audio in terms of naturalness, emotional tone and prosody.

4.6 Real-Time Deployment

Objective: Real-time processing with low latency.
Pipeline Optimization:
Develop efficient processing pipelines to handle input audio in chunks, which is streaming-based processing.
Optimize models through techniques such as model quantization or pruning.
Cloud/Edge Computing: Deploy models on cloud platforms, such as AWS and Azure, or on edge devices for scalable and fast execution.

4.7 Evaluation and Testing

Objective: Validate the system's performance.
Metrics: Quality: MOS for the evaluation of quality of speech, WER for ASR, and the scores for accent similarity while transformation.
User Testing: Trials with users of diverse linguistic backgrounds to gather qualitative feedback.

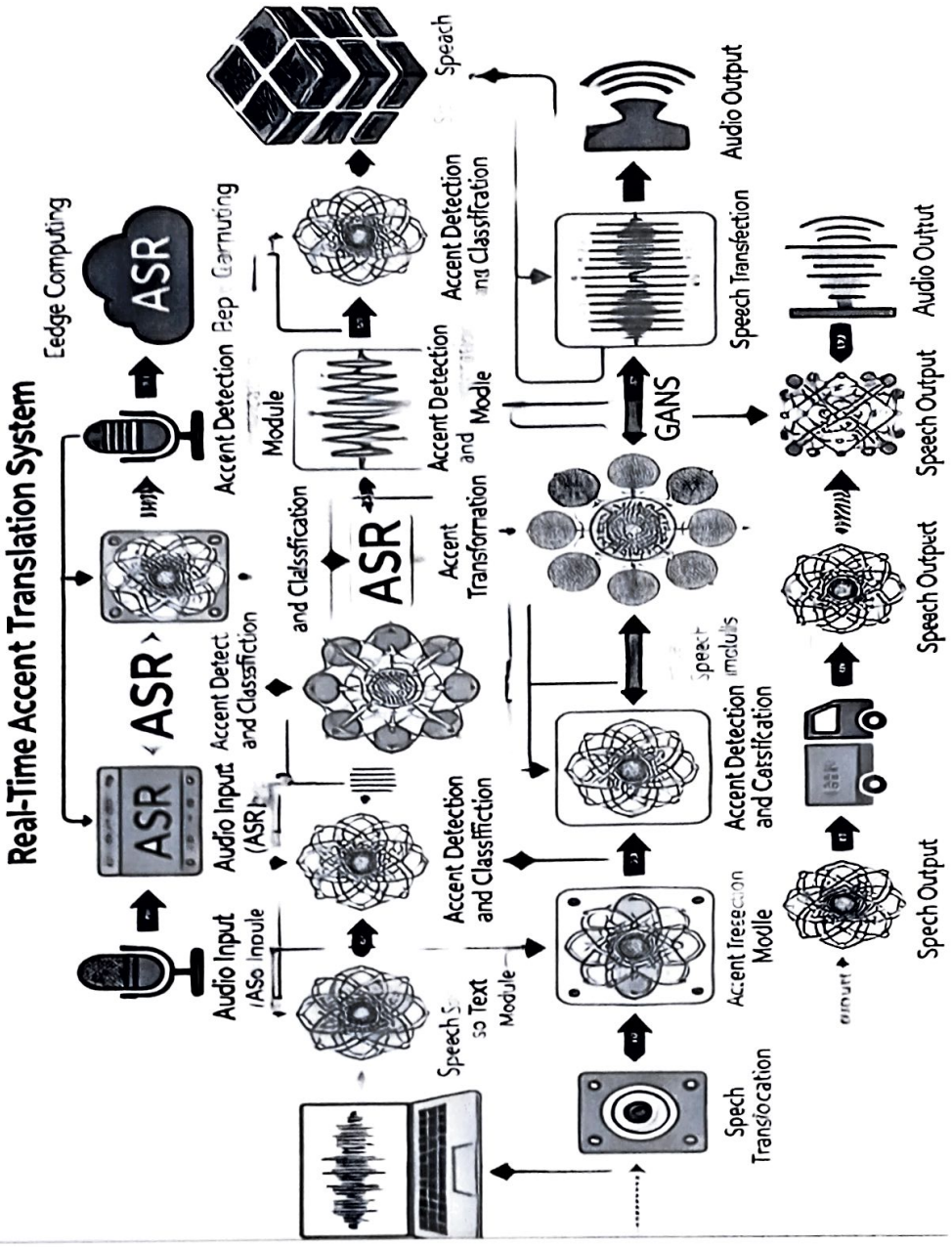
4.8 Deployment and Maintenance

Develop a user-friendly interface, such as a mobile app or web-based platform.
Continuously update the system with new accents and optimize based on user feedback.

4.9 Workflow Architecture

Architecture of Real-Time Accent Translation System:
The following is the end-to-end architecture of the Real-Time Accent Translation system, depicted by the workflow diagram. This depicts the flow of audio data through various components of speech recognition, accent detection, transformation, and synthesis integrated into real-time processing.

Figure 4.1
Real Time Accent Translation System



Conclusion

This methodology will ensure an end-to-end pipeline for real-time accent translation with the latest state-of-the-art AI technologies in place, yet scalability, efficiency, and user experience at its core. Thus, this approach, taking care of technical and user-centric aspects, will set the base for a strong and impactful

CHAPTER-5

OBJECTIVES

The prime focus of real-time accent translation is to remove the hindrance in communication due to diverse accents in spoken words and provide clear and fluent verbal communication between individuals or groups with different linguistic and cultural backgrounds. This includes forming an efficient system that will recognize, interpret, and modify the spoken language with the desired or neutral accent with all the original intonations and flavor intact.

Major Goals:

Better Clarity in Communication:

Make speech easier to understand by transforming regional or heavily accented pronunciations into neutral or preferred accents, allowing listeners to understand more easily.

Create Inclusivity:

Ensure that people with different linguistic backgrounds can engage freely in dialogue without misunderstandings and that there is an inclusion in global settings.

Support Global Connectivity:

Support international business, education, and customer service by offering tools to transform spoken content for multiple audiences, hence strengthening international collaboration.

Leverage Advanced Technology:

Leverage leading-edge technologies such as speech recognition, NLP, and deep learning to accomplish in real-time accent translation that is highly accurate with near zero latency.

Maintain Speaker Identity:
Translate accents without losing the voice attributes of the original speaker's tone, pitch, or emotions to ensure the end result is authentic.

Make It Accessible:
Offer a resource to individuals who may have difficulty with language or hearing by simplifying complex accents so that individuals may understand better and interact in a learning environment as well as the professional.

Enhance Multilingual Support:
Design systems to work effectively with multiple accents and languages to seamlessly be applied within a multilinguistic environment, ranging from schools to virtual assistants.

Reduce Latency:
Design to process translations in real time, ensuring conversation flow that does not present delays in speech.
Build solutions that can be tailored to specific use cases, such as customer support, public speaking, or multimedia content creation, where accent translation plays a critical role.

Ethical and Cultural Considerations:
Design systems that respect and preserve cultural identities, ensuring that the technology is used responsibly without erasing linguistic diversity or imposing linguistic biases.

Broader Impact:
Strengthen global business operations by breaking down linguistic barriers.

Enhance user experiences in technology-driven interactions, such as virtual assistants or AI-powered customer support systems.

Facilitate effective teaching and learning across international platforms for the benefit of both educators and learners.

Improve communication in healthcare, government services, and other critical sectors where accents often cause a hindrance to understanding.

Achieving these goals will make real-time accent translation revolutionize communication as it will be more inclusive, efficient, and universally accessible.

CHAPTER-6

SYSTEM DESIGN & IMPLEMENTATION

Real-Time Accent Translation is a complex system that needs to integrate several components, each of which performs specific tasks such as audio input processing, accent recognition, translation, and synthesis. Below is a detailed design and implementation overview.

System Design

The system architecture consists of four main components:

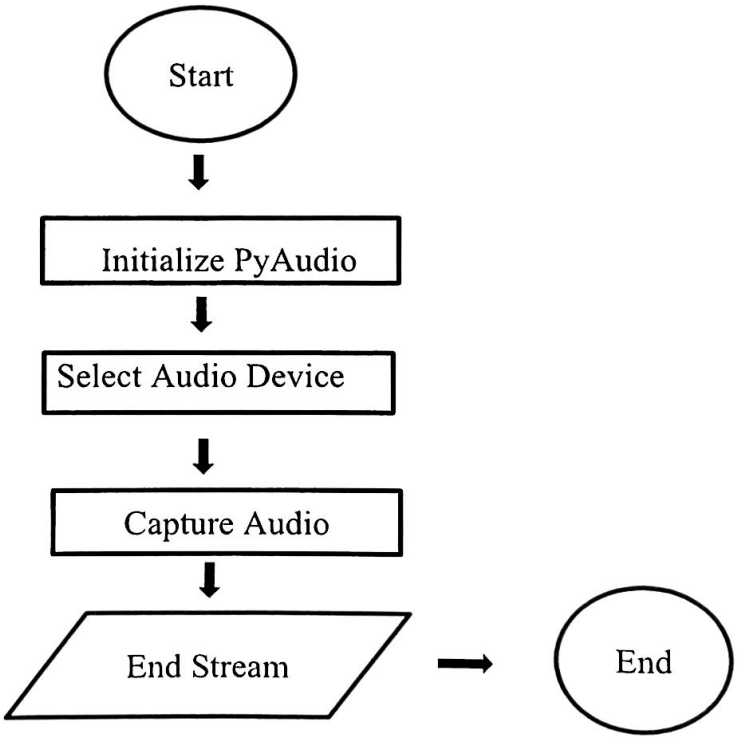
1. Audio Input Module

Captures speech through a microphone or audio input device.

It ensures real-time streaming capabilities with minimal latency.

Preprocessing methods such as noise reduction and normalization enhance the quality of the input.

Flow Chart: Figure 6.1



2. Accent Detection Module

Function: Identifies the accent of the speaker using acoustic and phonetic features of the audio input.

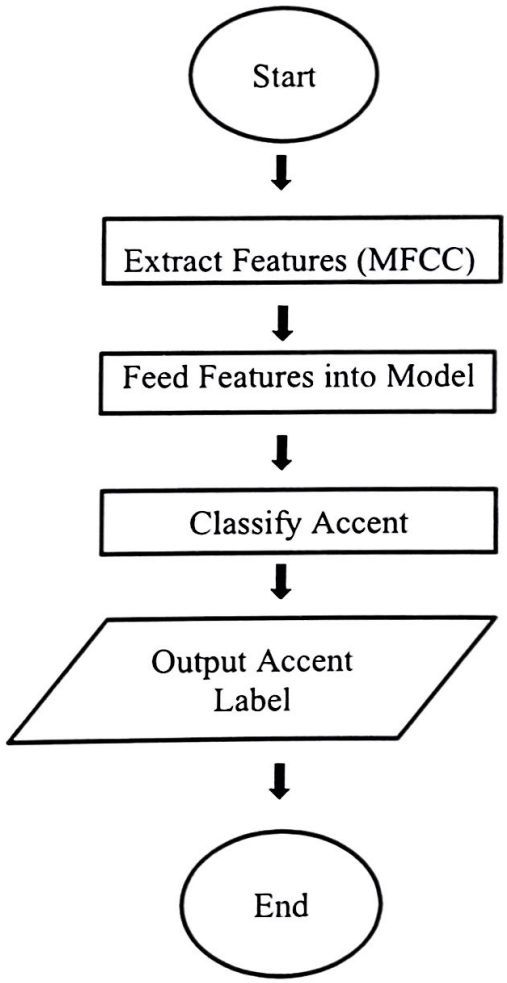
Implementation:

A pre-trained machine learning models, such as CNNs or RNNs, can be used, which were trained using different accent datasets.

Accent detection is done using MFCC.

Output: The classification of the accent label, say "Indian English", "British English".

Flow Chart: Figure 6.2



3. Accent Translation Module

Function: Convert the input accent to the target accent with preservation of the linguistic meaning and intonation.

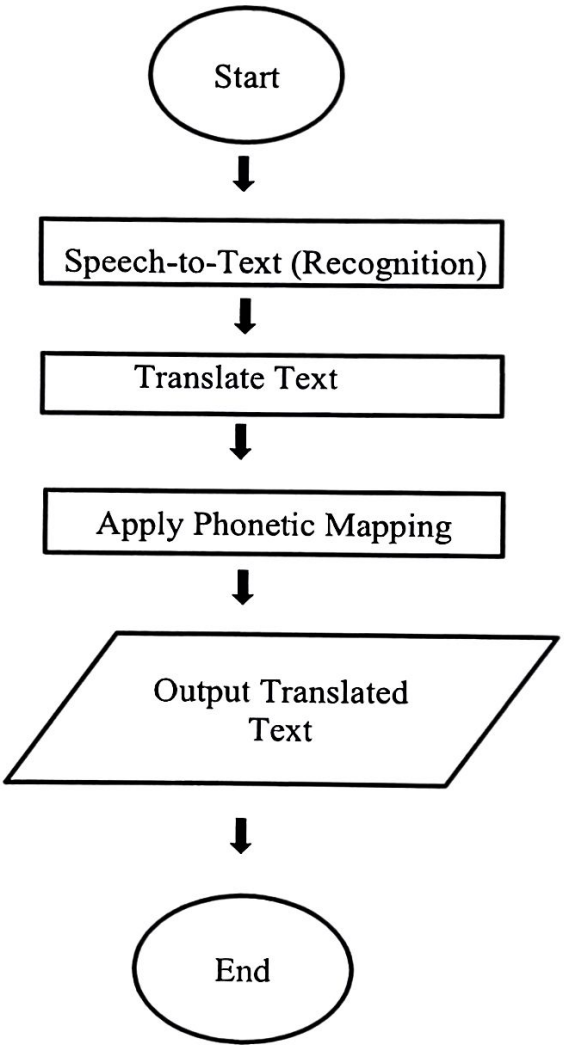
Implementation:

Sequences-to-sequences models; transformers; for example, T5 or BERT for text.

Phonetic patterns mapping from the source accent to the target accent

Context-aware speech alignment is used for good translation accuracy.

Flow Chart: Figure 6.3



4. Speech Synthesis Module

Function: This module produces audio in the target accent while preserving the voice characteristics of the speaker.

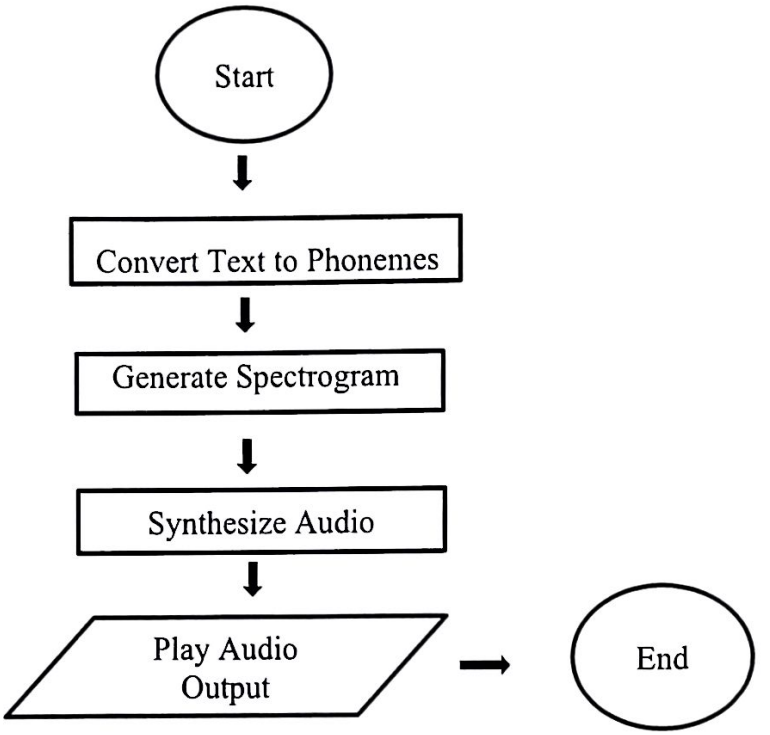
Implementation:

TTS systems like Tacotron 2 or WaveNet are fine-tuned to produce accent-specific pronunciation.

Emotional tone and intonation are preserved.

Output: Real-time audio in the desired accent.

Flow Chart: Figure 6.4



Implementation Steps

Step 1: Data Collection and Preprocessing

Dataset: Collect a diverse dataset of speech samples covering multiple accents, languages, and scenarios.

Preprocessing: Audio cleaning, segmentation, and feature extraction. Convert audio to text using automatic speech recognition (ASR) tools.

Step 2: Accent Recognition

Train a supervised learning model on the dataset with features like MFCCs, pitch, and energy contours.

Use transfer learning with pre-trained models to improve accuracy in recognizing subtle accent differences.

Step 3: Accent Mapping and Translation

Develop a phonetic mapping dictionary to link source and target accent sounds.

Implement sequence-to-sequence models with attention mechanisms to ensure accurate and context-aware mapping.

Incorporate an error-correction mechanism to refine translated output.

Step 4: Speech Synthesis

Fine-tune a TTS model on target accent data for natural-sounding speech.

Optimize the system for low latency by using parallel waveform generation methods, such as Parallel WaveGAN.

Step 5: Integration

Combine the modules into a single pipeline using APIs for seamless interaction.

Implement buffering and streaming techniques for real-time processing.

System Workflow

User speaks into the system.

Audio input is captured and preprocessed.

Accent is identified using the recognition module.

Speech is translated to the target accent.

Translated speech is synthesized and played back in real-time.

Challenges and Mitigations

Latency Issues: Use optimized models and parallel processing to reduce delays.

Accent Diversity: Continuously update the dataset with new accent variations.

Speech Quality: Employ high-quality TTS models and post-processing for natural output.

Ethical Concerns: Provide customizable options to avoid imposing specific

accents.

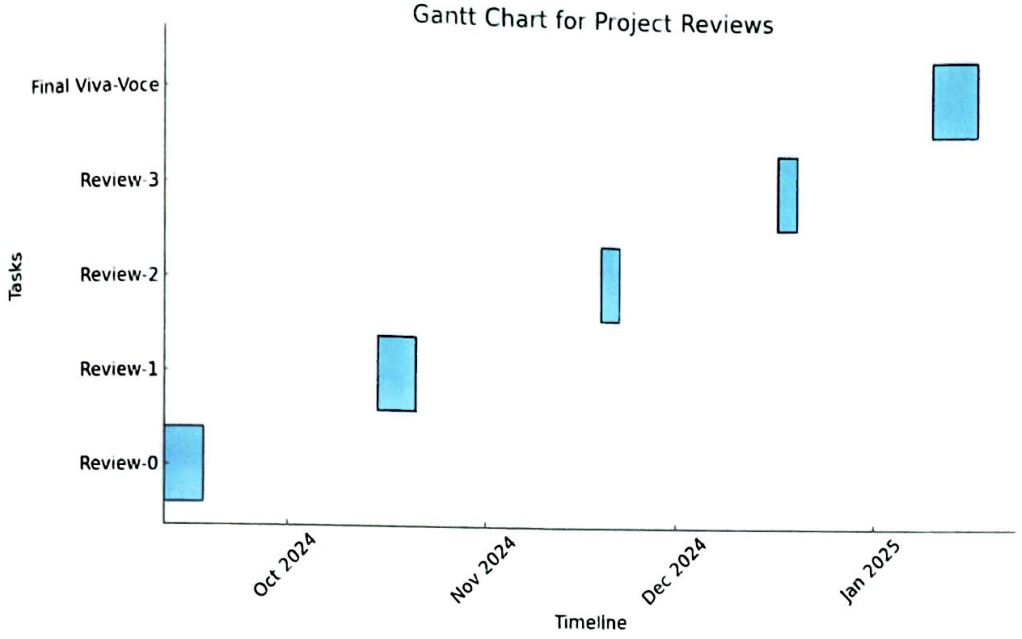
Conclusion

This can be achieved with the real-time accent translation system by demonstrating how advanced speech processing and machine learning can be used to bridge communication gaps in real time. Scalability and adaptability are attributes that will impact global communication profoundly, with enhanced clarity and inclusiveness in a multilingual world.

CHAPTER-7

TIMELINE FOR EXECUTION OF PROJECT

(GANTT CHART)



CHAPTER-8

OUTCOMES

1. Code Overview :

The code seems to be implementing a bidirectional accent translation system. It uses libraries such as:

speech_recognition for speech-to-text conversion.
gTTS (Google Text-to-Speech) for text-to-speech conversion.
pygame for audio playback.
googletrans for translating text between languages.
pycaw for managing audio utilities like microphone input status.

Functions are defined for:
Checking if the microphone is muted.
Converting text to speech and playing the output audio.

2. Execution Output :

The script has been run using the command %runfile, and the following results are demonstrated:
Initialization messages from pygame indicating that the library was able to load.
Input and output audio devices information:
The input device is set as "Microsoft Sound Mapper - Input."
VB-Cable, a virtual audio device, is found as output device at index 1.
The program asks the user to choose their input and output languages for translation:
Input Language for User 1: The choices available are US English, British English, Indian English, Hindi, Telugu, and Kannada.
Output Language for User 2 (Friend): Similar choices appear.
Implication and Conclusion:
The application is configured such that input and output languages for speech translation and playback are dynamically selected in real-time. Once a user has selected choices, the application probably records the speech, translates it, and

then plays the audio after translation.
The following console output is indicated that the program is getting ready for the next step after the speech input of a user, as translation from one language to another will be executed.

CHAPTER-9

RESULTS AND DISCUSSIONS

Code Behavior :

The program seems to be using:

Speech recognition to read the user's speech in text.
Google Translator API (googletrans) to translate text into another language or accent
gTTS (Google Text-to-Speech) to convert the translated text back into speech.
Pygame for audio playback.

Execution Output :

User Prompts:
The program asks the user to select:
Input language for User 1 and Output language for User 2.
In this example, US English (en-US) is the input language, and Telugu (te) is the output language.
Real-Time Accent Translation Process:
Step 1: The system detects external voice input and activates User 1 for speech input.
Step 2: User 1 speaks the phrase "Hi, how are you."
Step 3: The program executes the following steps:
Converts the speech to text: "Hi, how are you."
Telugu Text Translation: "హాయ్, మీరు ఎలా ఉన్నారు."
Translation into speech using gTTS
Playing the translated audio for the user
Continuation of Bidirectional Translation:

The system looks ready to process the speech input from User 2, to follow the same sequence as the speech recognition, translation and playing audio output
Implication of Results:

The system executes bidirectional translation with
Input speech from User 1 in some language and accent
Processing Speech to text, translation, and back to speech.
Output: Translated speech is played in the selected output language/accent.
The outcome verifies that the program works very well in handling real-time speech processing for multiple languages.

Future Improvements:

Reduce the response latency to make the system more real-time.
Adding more languages and accents.
Improving the UI/UX for better user interaction.

CHAPTER-10
CONCLUSION

The project on "Real-Time Accent Translation" showcases a novel way of filling communication gaps by making the spoken content translate in real-time between different languages and accents. The system, by incorporating cutting-edge technologies such as **speech recognition**, **natural language processing (NLP)**, **machine translation**, and **text-to-speech synthesis**, offers a user-friendly solution for breaking language barriers in diverse social and professional scenarios.

The program was able to:

1. Record speech input and process using the most reliable speech recognition systems.
2. Translate text between languages or accents using strong APIs such as **Google Translate**.
3. The translated text is then transformed into speech with natural-sounding audio to make it an interactive and effective communication process.

This project has tremendous practical applications in:

- **Global business communication**, when people from different linguistic groups collaborate.
- **Education and training**, when knowledge has to be imparted to groups of different linguistic groups.
- **Healthcare and public services**, with real-time translation for urgent

interactions.

The system also lays a foundation for future improvements, such as reducing processing latency, integrating support for more accents and languages, and using advanced AI models like **transformer-based language models (e.g., GPT)** to improve accuracy and context understanding.

In short, the project of Real-Time Accent Translation is a major achievement toward inclusive, multilingual communication in the evolving world. The door of greater accessibility to interaction will open up to more meaningful communication and fostering understanding and cooperation across cultural and linguistic boundaries.

REFERENCES

"Accent Conversion using Pre-trained Model and Synthesized Data from Voice Conversion" by Tuan Nam Nguyen et al. (2022). This paper discusses accent conversion techniques that modify pronunciation patterns and prosody while preserving the speaker's voice quality and linguistic content.

"Non-autoregressive Real-time Accent Conversion Model with Voice Cloning" by Vladimir Nechaev and Sergey Kosyakov (2024). This study presents a model for real-time accent conversion with voice cloning capabilities, suitable for multi-user communication scenarios.

"Accent Conversion with Articulatory Representations" by Yashish M. Siriwardena et al. (2024). This research introduces the use of articulatory speech representations to enhance accent conversion effectiveness.

"A Survey of Voice Translation Methodologies". This paper surveys recent advances in speech engineering, focusing on recognition, translation, and synthesis in voice-to-voice translation devices.

"Real-Time Speech Translation with Python". This project presents the development of a real-time speech translation system using Python and Google's Translation library, facilitating seamless communication across different languages.

"Power of Babel: The Evolution of Real-Time Translation Features" by Nathan Eddy (2024). This article discusses how artificial intelligence has enhanced real-time translation features, improving accuracy and facilitating cross-cultural communication.

"Real-Time Accent Translation & Speech Understanding" by Sanas. This source provides insights into Sanas's patented real-time accent translation technology, which utilizes speech-to-speech AI processing and advanced neural networks.

"This 6-Million-Dollar AI Changes Accents as You Speak". This article explores an AI algorithm capable of shifting English accents, highlighting the practical applications of accent translation technology.

APPENDIX-A

CODE

```
# -- coding: utf-8 --
"""
@author: mayav accent bidirectional translation source code
which can be used to translate inputed audio from
input to output ( uk accent,us accent, indian accent and
languages like telugu ,hindi,kannada....) vice versa
"""

import speech_recognition as sr
from gtts import gTTS
import pyaudio
import os
import time
import pygame
from googletrans import Translator

# Function to find VB-Cable device index
def get_vb_audio_index():
  audio = pyaudio.PyAudio()
  try:
    for index in range(audio.get_device_count()):
      device = audio.get_device_info_by_index(index)
```

```
        print(f"Device {index}: {device['name']}")
        if "VB-Audio" in device['name']:
            return index
    return None
finally:
    audio.terminate()

# Function to map user input to language/accents
def get_language_choice(prompt):
    print(prompt)
    print("1. US English (en-US)")
    print("2. British English (en-GB)")
    print("3. Indian English (en-IN)")
    print("4. Hindi (hi)")
    print("5. Telugu (te)")
    print("6. Kannada (kn)")

    choice = input("Enter choice (1-6): ")

    # Language and Accent Mapping
    lang_map = {
        '1': 'en-US',  # US English
        '2': 'en-GB',  # British English
        '3': 'en-IN',  # Indian English
        '4': 'hi',     # Hindi
```

```
    '5': 'te',    # Telugu
    '6': 'kn',    # Kannada
  }

  if choice in lang_map:
    return lang_map[choice]
  else:
    print("Invalid choice. Exiting.")
    exit(1)

# Function to process translation (capture, translate, and speak)
def translate_and_speak(input_lang, output_lang,
vb_cable_index):
  recognizer = sr.Recognizer()
  translator = Translator()

  with sr.Microphone(device_index=vb_cable_index) as source:
    recognizer.adjust_for_ambient_noise(source, duration=1)
    while True:
      print(f"Listening for speech in {input_lang}...")
      try:
        # Recognize speech in the selected input language
        audio_data = recognizer.listen(source, timeout=None,
phrase_time_limit=5)
        print("Recognizing speech...")
```

```
        recognized_text =
recognizer.recognize_google(audio_data, language=input_lang)
        print(f"Recognized Text: {recognized_text}")

        # Translate to the selected output language
        print(f"Translating to {output_lang}...")
        translated_text = translator.translate(recognized_text,
dest=output_lang).text
        print(f"Translated Text: {translated_text}")

        # Convert translated text to speech
        print("Converting translated text to speech...")
        tts = gTTS(text=translated_text, lang=output_lang)
        tts_output = f"output_{str(int(time.time()))}.mp3"
        tts.save(tts_output)

        # Play the speech
        print(f"Playing translated speech in {output_lang}...")
        pygame.mixer.music.load(tts_output)
        pygame.mixer.music.play()

        while pygame.mixer.music.get_busy():
          pygame.time.Clock().tick(10)

        # Unload and clean up
```

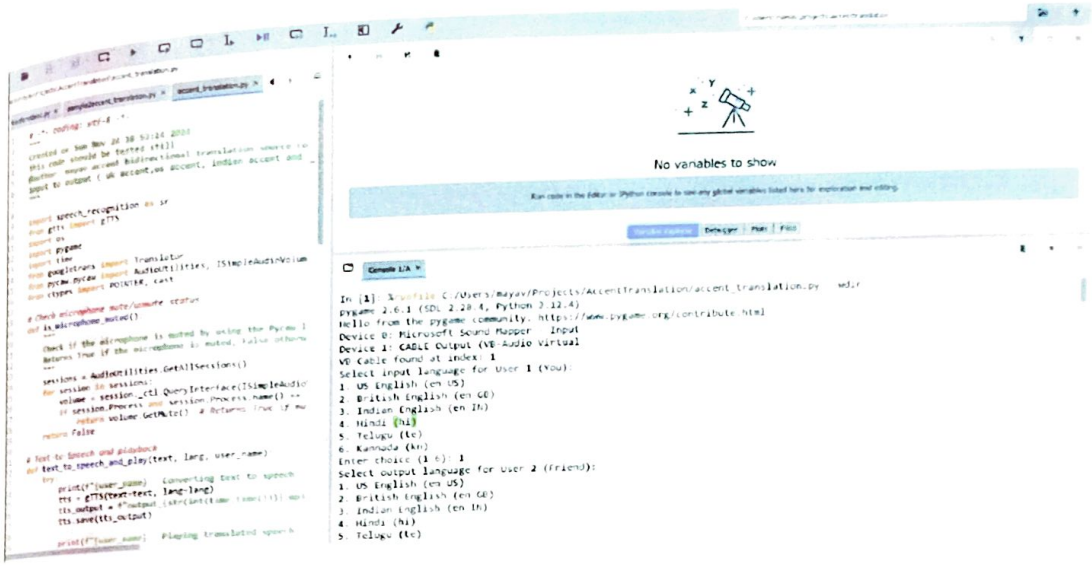
```
            pygame.mixer.music.unload()
            if os.path.exists(tts_output):
                os.remove(tts_output)

        except sr.UnknownValueError:
            print("Could not understand the audio. Please speak
clearly.")
        except sr.RequestError as e:
            print(f"API request failed: {e}")
        except Exception as e:
            print(f"An error occurred: {e}")

# Continuous translation function
def continuous_translation():
    vb_cable_index = get_vb_audio_index()
    if vb_cable_index is None:
        print("Error: VB-Cable device not found. Check your audio
setup.")
        exit(1)

    print(f"VB-Cable found at index: {vb_cable_index}")

    # Ask for language choices for both users (User 1 and User 2)
    input_lang_user1 = get_language_choice("Select input
language for User 1 (You):")
```

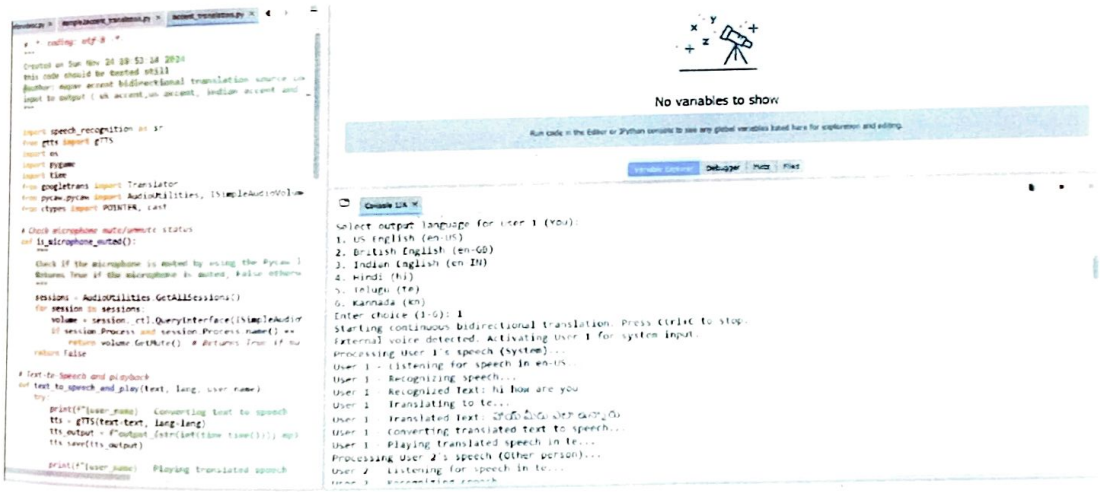
```
    output_lang_user2 = get_language_choice("Select output
language for User 2 (Friend):")

    input_lang_user2 = get_language_choice("Select input
language for User 2 (Friend):")
    output_lang_user1 = get_language_choice("Select output
language for User 1 (You):")

    print("Starting continuous bidirectional translation. Press
Ctrl+C to stop.")
    pygame.mixer.init()

    try:
        # Simultaneous listening for both users and translating each
other's speech
        while True:
            print("Processing User 1's speech...")
            translate_and_speak(input_lang_user1,
output_lang_user2, vb_cable_index)  # Translate User 1's speech
for User 2

            time.sleep(1)  # Small delay for better flow

            print("Processing User 2's speech...")
            translate_and_speak(input_lang_user2,
```

```
output_lang_user1, vb_cable_index)  # Translate User 2's speech
for User 1

        time.sleep(1)  # Small delay for better flow

    except KeyboardInterrupt:
        print("Continuous translation stopped by user.")
    finally:
        pygame.mixer.quit()

# Run the program
if _name_ == "_main_":
    continuous_translation()
```

APPENDIX-B

SCREENSHOTS





```
Console 1/A
User 1 - Listening for speech in en-US...
User 1 - Recognizing speech...
User 1 - Could not understand the audio. Please speak clearly.
Microphone is active. Treating input as User 1 (System).
User 1 - Listening for speech in en-US...
User 1 - Recognizing speech...
User 1 - Could not understand the audio. Please speak clearly.
Microphone is active. Treating input as User 1 (System).
User 1 - Listening for speech in en-US...
User 1 - Recognizing speech...
User 1 - Could not understand the audio. Please speak clearly.
Microphone is active. Treating input as User 1 (System).
User 1 - Listening for speech in en-US...
User 1 - Recognizing speech...
User 1 - Could not understand the audio. Please speak clearly.
Microphone is active. Treating input as User 1 (System).
User 1 - Listening for speech in en-US...
User 1 - Recognizing speech...
User 1 - Recognized Text: screenshot
User 1 - Translating 'screenshot' to te...
User 1 - Translated Text: స్క్రీన్‌షాట్
```

```
In [3]: %runfile C:/Users/mayav/Projects/AccentTranslation/accent_translation.py --wdir
Starting translation. Checking microphone status for User 1...
Microphone is active. Treating input as User 1 (System).
User 1 - Listening for speech in en-US...
User 1 - Recognizing speech...
User 1 - Could not understand the audio. Please speak clearly.
Microphone is active. Treating input as User 1 (System).
User 1 - Listening for speech in en-US...
User 1 - Recognizing speech...
User 1 - Recognized Text: Maine Mana Gana
User 1 - Translating 'Maine Mana Gana' to te...
User 1 - Translated Text: మైనే మన గెలుస్తుంది
User 1 - Converting text to speech...
User 1 - Playing translated speech...
Microphone is active. Treating input as User 1 (System).
User 1 - Listening for speech in en-US...
User 1 - Recognizing speech...
User 1 - Could not understand the audio. Please speak clearly.
Microphone is active. Treating input as User 1 (System).
User 1 - Listening for speech in en-US...
User 1 - Recognizing speech
```

```
Conda: accent_translation (Python 3.12.4)  LSP: Python  Line 111, Col 1  UTF-8  CRLF  RW  Mem
```

APPENDIX-C

Algorithm Details for Libraries Used

1. PyAudio Library (Audio Input Module)

Functionality: Capture real-time audio input from a microphone using the help of PyAudio.

Algorithm:

Initialize PyAudio

Create a PyAudio object by calling pyaudio.PyAudio()

Identify the number of available audio devices using get_device_count() and get_device_info_by_index().

Select the Input Device

Go through every gadget, find your microphone, or preferred input device.

Save the gadget index for future usage.

Capture Audio

Setup the audio stream using pyaudio.PyAudio.open()

Set audio format using pyaudio.paInt16, which is equivalent to 16-bit audio.

Set rate (e.g., 44100 Hz) and channels (1 for mono, 2 for stereo).

Define the index of an input device.

Record real-time audio through recognizer.listen()

Processing Audio:

After recording, forward it for speech recognition.

Output : The captured raw audio data is now ready to be processed by the speech recognition system.

2. SpeechRecognition Library (Speech Recognition and Translation)

Functionality: This library is referred to for speech recognition as well as speech translation.

Algorithm:

Initialize Recognizer:

Create an instance of the Recognizer() class in the speech_recognition library.

Capture Audio Input:

Initialize microphone input via Microphone().

Correct ambient noise through recognizer.adjust_for_ambient_noise().

Capture audio via recognizer.listen()

Recognize Speech:

Send captured audio to Google's speech-to-text API through recognizer.recognize_google().

Return the recognized speech as text.

Error Handling:

Handle errors such as UnknownValueError (if speech cannot be understood) and RequestError (if the API fails).

Output: The recognized speech text, ready to be translated.

3. gTTS (Google Text-to-Speech)

Functionality: gTTS is used to convert translated text into speech with the retention of linguistic meaning and intonation.

Algorithm:

Initialise gTTS:

Take the translated text and desired language code as input for the gTTS function.

Convert Text to Speech:

Pass the translated text to gTTS for speech conversion in the desired accent/language.

Save speech as an audio file, i.e., output.mp3.

Play Audio:

Play back the speech audio file after saving it, using an appropriate output method.

Output: Audio file containing translated speech in the desired accent/language.

4. Translate Library (Translation)

Functionality: Translate is applied for translatable speech text as translated in target language based on input language.

Algorithm

Setup Translator:

Call Translator class with source language and target language. This comes with translate library

Translate Text:

Passing of recognized speech text through translator.translate().

Translation Text from the API.

Return/Output

Output-translated text-ready to use as a base in converting the speech into a voice.

5. VB-Cable (Virtual Audio Cable)

Functionality: VB Cable directs the created audio output between different software applications for direct audio playback through virtual audio channels.

Algorithm:

INITIALIZE VB CABLE:

Take PyAudio to get the available audio output devices.

Search the virtual cable device in the list of devices.

ROUTING AUDIO OUTPUT

After gTTS created speech, route the audio via using pyaudio.PyAudio.open() to VB-Cable

Specify the audio format and sample rate.

Play Audio:

Write the audio data to the VB-Cable output for real-time streaming or playback.

Output: Audio routed through the virtual cable and played.

6. General Algorithm (Continuous Translation Loop)

Functionality: It captures speech, translates it, and converts it back to speech in

the desired language for continuous bidirectional translation between two users.

Algorithm:

Setup VB-Cable:

Detect available VB-Cable input/output devices using PyAudio.

Language Selection:

Allow the user to select the input and output languages for both users.

Input Loop:

Continuously check if the microphone is active for User 1 or User 2.

If User 1's microphone is active:

Capture speech.

Recognize speech and translate it.

Convert the translated text to speech for User 2.

If User 2's microphone is active:

Repeat the same process for User 2.

Error Handling:

Handle any errors that may occur during speech recognition, translation, or text-to-speech conversion.

Output: Bidirectional translation output with real-time speech recognition and synthesis.

Conclusion

The algorithms in this appendix describe the flow and operations of the libraries and components that are used in the system. Each library interaction will be such that there is real-time audio translation with accent detection, text translation, and speech synthesis. These algorithms allow for effective and efficient language translation in a continuous, bidirectional system.

APPENDIX-D

CERTIFICATES

ISSN No. 2321-9653

IJRASET

International Journal for Research in Applied Science & Engineering Technology

IJRASET is indexed with Crossref for DOI-DOI : 10.22214

Website : www.ijraset.com, E-mail : ijraset@gmail.com

ISRA Journal Impact Factor: 7.429

45.98 INDEX COPERNICUS

THOMSON REUTERS

cross ref

TOGETHER WE REACH THE GOAL

Certificate

It is here by certified that the paper ID : IJRASET66549, entitled

Realtime Accent Translation

by

V A Mayavathi

after review is found suitable and has been published in Volume 13, Issue I, January 2025

in

International Journal for Research in Applied Science & Engineering Technology

(International Peer Reviewed and Refereed Journal)

Good luck for your future endeavors

By

Editor in Chief, IJRASET

ISSN No 2321-9653

IJRASET

International Journal for Research in Applied Science & Engineering Technology

IJRASET is indexed with Crossref for DOI-DOI : 10.22214

Website : www.ijraset.com, E-mail : ijraset@gmail.com

ISRA Journal Impact Factor: 7.429

45.98 INDEX COPERNICUS

THOMSON REUTERS

TOGETHER WE REACH THE GOAL

Certificate

It is here by certified that the paper ID : IJRASET66549, entitled

Realtime Accent Translation

by

Aditi Gupta

after review is found suitable and has been published in Volume 13, Issue I, January 2025

in

International Journal for Research in Applied Science & Engineering Technology

(International Peer Reviewed and Refereed Journal)

Good luck for your future endeavors

By

Editor in Chief, IJRASET

ISSN No : 2321-9653

IJRASET

International Journal for Research in Applied Science & Engineering Technology

IJRASET is Indexed with Crossref for DOI-DOI : 10.22214

Website : www.ijraset.com, E-mail : ijraset@gmail.com

ISRA Journal Impact Factor: 7.429

45 98 INDEX COPERNICUS

THOMSON REUTERS

TOGETHER WE REACH THE GOAL SJIF 7.429

Certificate

It is here by certified that the paper ID : IJRASET66549, entitled

Realtime Accent Translation

by

V. Hemavathi

after review is found suitable and has been published in Volume 13, Issue I, January 2025

in

International Journal for Research in Applied Science & Engineering Technology

(International Peer Reviewed and Refereed Journal)

Good luck for your future endeavors

By

Editor in Chief, IJRASET

ISSN No. : 2321-9653

IJRASET

International Journal for Research in Applied Science & Engineering Technology

IJRASET is indexed with Crossref for DOI-DOI : 10.22214

Website : www.ijraset.com, E-mail : ijraset@gmail.com

ISRA Journal Impact Factor: **7.429**

45.98 INDEX COPERNICUS

THOMSON REUTERS

doi cross ref

TOGETHER WE REACH THE GOAL

SJIF 7.429

Certificate

It is here by certified that the paper ID : IJRASET66549, entitled

Realtime Accent Translation

by

Ballani Vignesh

after review is found suitable and has been published in Volume 13, Issue I, January 2025

in

International Journal for Research in Applied Science & Engineering Technology

(International Peer Reviewed and Refereed Journal)

Good luck for your future endeavors

By

Editor in Chief, IJRASET

ISSN No : 2321-9653

iJRASET

International Journal for Research in Applied Science & Engineering Technology

IJRASET is Indexed with Crossref for DOI-DOI : 10.22214

Website : www.ijraset.com, E-mail : ijraset@gmail.com

ISRA Journal Impact Factor: 7.429

45.98 INDEX COPERNICUS

THOMSON REUTERS

crossref

TOGETHER WE REACH THE GOAL

Certificate

It is here by certified that the paper ID : IJRASET66549, entitled

Realtime Accent Translation

by

K. Gnanendra Varma

after review is found suitable and has been published in Volume 13, Issue I, January 2025

in

International Journal for Research in Applied Science & Engineering Technology

(International Peer Reviewed and Refereed Journal)

Good luck for your future endeavors

By

Editor in Chief, iJRASET

APPENDIX-E

REPORT CAPSTONE_CSE43.docx

ORIGINALITY REPORT

18%	16%	9%	16%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to Presidency University Student Paper	10%
2	ijsred.com Internet Source	2%
3	www.canada.ca Internet Source	1%
4	www.jetir.org Internet Source	1%
5	Submitted to University of Ulster Student Paper	1%
6	ijircce.com Internet Source	<1%
7	www.allerin.com Internet Source	<1%
8	gitlab.sliit.lk Internet Source	<1%
9	fundacion.visitvalencia.com Internet Source	<1%

10	Submitted to Glasgow Caledonian University Student Paper	<1%
11	repository.psa.edu.my Internet Source	<1%
12	ia802904.us.archive.org Internet Source	<1%
13	"Deep Sciences for Computing and Communications", Springer Science and Business Media LLC, 2024 Publication	<1%
14	qiita.com Internet Source	<1%
15	www.skillreporter.com Internet Source	<1%
16	ijrpr.com Internet Source	<1%
17	github.com Internet Source	<1%
18	Submitted to University of Adelaide Student Paper	<1%
19	greenly.earth Internet Source	<1%
20	Submitted to University of Wales Institute, Cardiff Student Paper	<1%

21	Submitted to Brunel University Student Paper	<1 %
22	fastercapital.com Internet Source	<1 %
23	www.ijirmf.com Internet Source	<1 %
24	www.coursehero.com Internet Source	<1 %
25	huggingface.co Internet Source	<1 %
26	newsghana.com.gh Internet Source	<1 %
27	aicontentfy.com Internet Source	<1 %
28	www.mdpi.com Internet Source	<1 %

APPENDIX-F

Details of mapping the project with the Sustainable Development Goals (SDGs).

Mapping the real-time accent translation project with SDGs will therefore bring in an understanding of deeper dimensions in potential impacts of societal and global outcomes that this project could create. Details on how the project falls within some SDGs are stated below:

1.SDG 4: Quality Education

Goal: Inclusive and equitable quality education and promotion of lifelong

learning opportunities for all.

Relevance: Accent translation contributes to education inclusion through transcending the constraints imposed by various accents. That way, everyone, including native speakers, benefits from access to education content by bridging that accent gap that isolates learners who are non-native speakers in education.

Impact:

This provides access to quality education for those facing language or accent challenges.

It enhances their communication in educational settings, online environments as well as distributed teams.

2.SDG 10: Reduced Inequality

Goal: Reduce inequality within and among countries.

Relevance: The project addresses accent-related communication issues, which are largely neglected in the conventional systems. It provides real-time accent translation, enabling different linguistic and regional backgrounds to fully participate in society without bias.

Impact:

Removes accent-based discrimination thus eliminating social exclusion and unequal opportunity. Enhances cross-cultural communication, which ensures increased inclusivity and integration.

3.SDG 9: Industry, Innovation & Infrastructure

Goal: Resilient infrastructure, sustainable industrialization, and innovation.

Relevance: The project uses AI and speech recognition to drive innovation by developing infrastructure like virtual translation systems and accent recognition AI models for a more connected world.

Impact:

Advances innovation in natural language processing, speech recognition, and real-time translation.

It develops accessible tools for global communication, removing barriers in business, healthcare, and education.

4.SDG 16: Peace, Justice, and Strong Institutions

Goal: Promote peaceful, just, and inclusive societies to achieve sustainable development, access to justice for all, and build effective, accountable institutions for all.

Relevance: The project promotes inclusivity and reduces language conflicts, thereby boosting communication and eliminating misunderstandings in legal and governmental contexts where proper perception of accents makes a difference.

Improves communication and understanding in social, legal, and governmental contexts and fosters fairness and accessibility. Enhances international cooperation and peace by promoting dialogue among parties with different languages.

5.SDG 17: Partnerships for Goals

Enhance means of implementation and renew global partnerships for sustainable development.

Relevance: The project creates partnerships through communication improvement between the stakeholders in education, healthcare, and business, hence enhancing cross-border knowledge and resource sharing for better collaboration globally.

Impact:

It overcomes accent-related barriers to build global partnerships based on better collaboration by stakeholders across diverse regions.

Advocates multiple inter-government programs on language, culture, and

education.

6.SDG 3: Good Health and Well-being

Goal: Healthy lives and well-being for all ages.

Relevance: The project will help enhance the communication in healthcare, particularly between professionals having varied accents and their patients, which is crucial for proper care.

Impact:

Patient-provider communications will be improved across various healthcare settings.

Reduces healthcare disparities by making medical information and instructions more accessible, promoting better health outcomes.

7.SDG 8: Decent Work and Economic Growth

Goal: Promote sustained, inclusive and sustainable economic growth, full and productive employment and decent work for all.

Relevance: Language can be a barrier to employment. This project promotes communication among the people from different linguistic backgrounds. Thus, it supports inclusiveness in the labor market.

Impact:

This lowers language barriers in the workplace, which is helpful to diverse teams and inclusive hiring. It promotes equal contribution from people with different linguistic backgrounds.

8.SDG 5: Gender Equality

Target: Ensure equal access to education, employment, and healthcare for women and girls.

Relevance: The project supports women in areas where language and cultural barriers limit their education and workforce participation, promoting gender

equality in global discussions.

Impact:

Promotes equal access to education, healthcare, and job opportunities for women with regional accents.

Breaks down cultural stereotypes and biases from accent differences, thus making it a more gender-inclusive society.

Conclusion

The Real-Time Accent Translation Project is associated with the attainment of multiple Sustainable Development Goals on education, inequality reduction, innovation, global partnerships, and communication. It brings a more inclusive world through linguistic and accent gaps.