IJARSCT

ISSN (Online) 2581-9429

International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)

International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal

Impact Factor: 7.53

Volume 4, Issue 2, June 2024

Multilingual Speech Transcription and Translation System

Dheeraj K N, Vaibhav Ravindra, Prof. A S Vinay Raj
Department of Information Science
Global Academy of Technology, Bangalore, India

Abstract: *This project implements a multilingual speech recognition and translation system using Python. The system leverages libraries such as `pygame` for audio playback, `gTTS` for text-to-speech conversion, `speech_recognition` for capturing and recognizing spoken language, and Machine Translation model for translating text between languages. Users interact with the system via a Streamlit-based web interface, where they can select the input and target languages and control the start and stop of the translation process. The program can auto-detect the input language if the user chooses, and it provides real-time feedback by displaying the recognized text, translating it into the chosen target language, and converting the translated text into speech.*

At the core of the system is a continuous loop that listens for speech input through the microphone, processes the audio to recognize the spoken words, and then translates the recognized text into the desired language. The translated text is not only displayed but also spoken aloud using the text-to-speech functionality. The interface is user-friendly, with clear indications of the current status (listening, processing, recognized text, translated text) and handles errors gracefully, providing feedback in case of issues. This project demonstrates a practical application of speech recognition and translation technologies, offering a seamless and interactive multilingual communication tool

Keywords: Multilingual speech recognition, translation system, Python, gTTS, speech_recognition, Machine Translation model, Streamlit, web interface.

I. INTRODUCTION

In an increasingly interconnected world, the need for effective communication across different languages is more critical than ever. The domain of speech recognition and translation technology addresses this need by enabling seamless interaction between speakers of different languages. This project focuses on the development of a multilingual speech recognition and translation system, designed with the aim of aiding visually impaired or disabled individuals in their daily communication.

The system is built using several key technologies:

- Pygame: for audio playback, ensuring that translated text can be heard.
- gTTS (Google Text-to-Speech): for converting translated text into speech, making the output accessible to those who cannot read.
- Speech Recognition: for capturing and recognizing spoken language, allowing users to input commands and communicate verbally.
- Streamlit: for creating an intuitive web interface, providing a user-friendly platform for interaction.

The primary use of this project is to support visually impaired or disabled individuals who may face challenges in reading text or navigating traditional translation tools. By converting spoken language into text and then translating and vocalizing it in real-time, the system allows these users to communicate more effectively and independently.

The methodology involves several steps: The user selects the input and target languages from a predefined list or chooses an auto-detect option for the input language. The system continuously listens for speech input via the microphone. The captured audio is processed to recognize the spoken words using Google's speech recognition capabilities.The recognized text is translated into the selected target language using Google's translation services. The

DOI: 10.48175/IJARSCT-18843

translated text is converted into speech using gTTS and played back to the user via Pygame. The recognized, translated text and its English equivalent are displayed on the interface, providing visual feedback for those who can see.

This project exemplifies the integration of modern speech and language processing technologies to create an inclusive tool that bridges communication gaps and enhances accessibility for individuals with visual impairments or disabilities.

1.1 Briefing About the Project

The development of our multilingual speech recognition and translation system for visually impaired or disabled individuals involves several critical phases to ensure a comprehensive and effective solution:

- Requirement Analysis Phase: Identifying and documenting the specific needs of visually impaired or disabled individuals for real-time speech recognition and translation.
- Design Phase: Creating detailed plans and architecture, focusing on user accessibility and seamless integration of speech recognition and translation technologies.
- Development Phase: Writing and compiling the code to implement functionalities such as speech recognition, language detection, translation, and text-to-speech conversion.
- Testing Phase: Verifying that the software accurately recognizes speech, detects the correct language, translates effectively, and provides clear audio output.
- Release: Deploying the final product to the target audience, ensuring it is easy to use and accessible.

A well-structured design is essential for achieving a high degree of extensibility, reusability, and maintainability. This focus on robust design not only enhances the quality and adaptability of the project but also significantly reduces the overall development cost. Therefore, identifying an efficient method to analyze and refine the design of this speech recognition and translation system is crucial for its success in assisting visually impaired or disabled individuals.

1.2 Existing System

Several existing systems and technologies have been developed to assist visually impaired or disabled individuals with speech recognition and translation:

1. Google Translate: Google Translate offers robust speech recognition and translation capabilities in multiple languages. It allows users to speak into the app and receive translations both in text and audio form. However, it is primarily designed for general use and may not be fully optimized for visually impaired users.
2. Microsoft Translator: Microsoft Translator provides similar functionalities to Google Translate, with the ability to recognize speech, translate it into various languages, and offer text-to-speech output. It includes features like conversation mode for real-time translation between speakers of different languages. Yet, its user interface may not be fully accessible for those with visual impairments.
3. Apple's VoiceOver with Siri: Apple's VoiceOver is a screen reader integrated into iOS devices that works alongside Siri. Siri can perform speech recognition and translation, helping visually impaired users navigate and interact with their devices. While VoiceOver improves accessibility, Siri's translation capabilities are somewhat limited compared to dedicated translation apps.
4. Amazon Alexa: Alexa, Amazon's virtual assistant, supports speech recognition and can interact in multiple languages. It provides basic translation services and can read out translations. Despite its accessibility features, its primary focus is not on translation, making it less specialized for the needs of visually impaired individuals requiring robust translation support.
5. Jaws (Job Access with Speech): Jaws is a popular screen reader for Windows that assists visually impaired users by reading out screen content. It can work with various applications, including translation services. While effective as a screen reader, it does not inherently offer speech recognition and translation features.
6. Seeing AI: Developed by Microsoft, Seeing AI is an app designed for visually impaired users, using AI to describe people, text, and objects around the user. While it includes text recognition and basic language translation, it does not offer comprehensive speech recognition and translation services.

These existing systems provide foundational functionalities, but they often lack a comprehensive, integrated approach tailored specifically for visually impaired or disabled individuals needing real-time multilingual speech recognition and

DOI: 10.48175/IJARSCT-18843

translation. Our project aims to bridge this gap by offering an accessible, user-friendly solution that combines these features into a single, cohesive system.

1.3 Limitations of Existing System

Google Translate:

- Primarily designed for general use, not optimized for visually impaired users.
- User interface may not be fully accessible or user-friendly for those with visual impairments

Microsoft Translator:

- User interface might not be entirely accessible for visually impaired individuals.
- Translation features, while robust, are not specifically tailored for the needs of visually impaired users.

Apple's VoiceOver with Siri:

- Siri's translation capabilities are limited compared to dedicated translation applications.
- VoiceOver provides accessibility but does not offer a fully integrated translation and speech recognition solution.

Amazon Alexa:

- Basic translation services, not specialized for robust translation needs.
- Focuses more on general virtual assistant tasks rather than specific translation and accessibility functions.

Jaws (Job Access With Speech):

- Effective as a screen reader but lacks built-in speech recognition and translation features.
- Requires additional applications or services to provide translation functionalities.

Seeing AI:

- Includes basic text recognition and language translation but does not offer comprehensive speech recognition.
- Primarily focuses on object and scene recognition rather than multilingual speech translation.

These limitations highlight the need for a more integrated and specialized solution that caters specifically to the needs of visually impaired or disabled individuals, offering seamless and user-friendly speech recognition and multilingual translation capabilities.

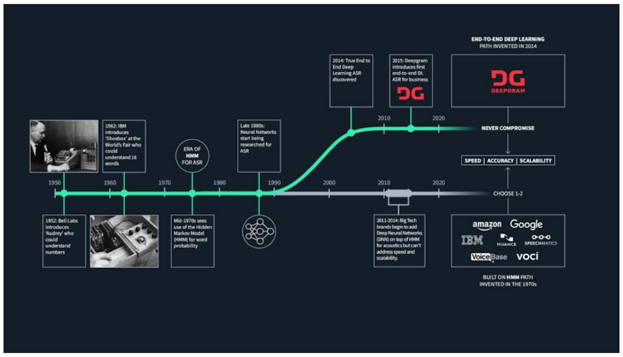

END-TO-END DEEP LEARNING PATH INVENTED IN 2014
2014: True End to End Deep Learning ASR discovered
2015: Deepgram introduces first end-to-end DL ASR for business
DG
DG DEEPGRAM
1962: IBM introduces 'Shoebox' at the World's Fair who could understand 16 words
ERA OF HMM FOR ASR
Late 1980s: Neural Networks start being researched for ASR
2010
2020
NEVER COMPROMISE
SPEED | ACCURACY | SCALABILITY
1950
1960
1970
1980
1990
2000
2010
2020
CHOOSE 1-2
1952: Bell Labs introduces 'Audrey' who could understand numbers
Mid-1970s sees use of the Hidden Markov Model (HMM) for word probability
2011-2014: Big Tech brands begin to add Deep Neural Networks (DNN) on top of HMM for acoustics but can't address speed and scalability.
amazon
Google
IBM
NUANCE
SPEECHMATICS
VoiceBase
voci
BUILT ON HMM PATH INVENTED IN THE 1970s

Figure 1.3.1 Existing System

1.4 Proposed System

Our proposed system is a comprehensive solution designed to address the limitations of existing systems by offering the following key features:

DOI: 10.48175/IJARSCT-18843

- Real-time Multilingual Speech Recognition: The system will leverage advanced speech recognition algorithms to accurately transcribe spoken language into text in real-time. This functionality will enable users to interact with the system effortlessly using their voice.
- Integrated Translation Services: Building upon the foundation of speech recognition, the system will seamlessly translate the transcribed text into multiple languages in real-time. Users will have the flexibility to choose their preferred target language, allowing for effective communication across linguistic barriers.
- Accessibility for Visually Impaired Users: Accessibility is at the forefront of our design, with a user interface optimized for visually impaired individuals. The system will feature compatibility with screen readers and other assistive technologies, ensuring that all users can navigate and interact with the application effortlessly.
- Customization and Personalization: To cater to the diverse needs of users, the system will offer customization options such as language preferences, speech recognition accuracy settings, and personalized user profiles. This level of customization will enhance user experience and improve overall usability.
- Reliability and Efficiency: Our system will prioritize reliability and efficiency, with robust algorithms and backend infrastructure to ensure smooth and uninterrupted operation. Users can rely on the system for accurate and timely translation of speech, enhancing their communication and productivity

By combining cutting-edge technology with a focus on accessibility and user experience, our proposed system aims to empower visually impaired individuals with seamless multilingual communication capabilities, ultimately enhancing their independence and inclusion in an increasingly globalized world.

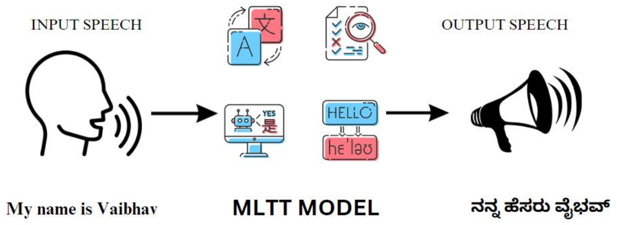


Figure 1.4.1 Proposed System

II. LITERATURE SURVEY

In [1], a framework is presented for developing speech-to-text translation (ST) systems using only monolingual speech and text corpora. The system initialization involves a cross-modal bilingual dictionary derived from monolingual corpora, enabling word-by-word translation for unseen speech utterances. Experimental results indicate similar BLEU scores to supervised models, rendering it applicable for language pairs with limited resources.

In [2], the authors suggest the utilization of Hidden Markov Models (HMM) for speech-to-text conversion, aiming to improve text understanding and supporting visually impaired users. The synthesized speech aims to deliver a comprehensible output from audio inputs, employing Digital Signal Processing (DSP) algorithms.

In [3], a fresh end-to-end ST framework with two decoders is presented to address deeper connections between source language audio and target language text. By utilizing paired source language audio and target language text in training, the proposed approach exhibits enhanced performance compared to traditional cascaded systems.

In [4], the emphasis is on improving speech-to-speech translation (S2ST) models by examining the effect of synthesized target speech. A multi-task framework is suggested to enhance S2ST systems with multiple targets from various text-to-speech (TTS) systems, yielding consistent enhancements over baselines.

In [5] investigates the application of Text-to-Speech (TTS) and Speech-to-Text (STT) technologies in developing a tool for educational documentation in English. The article offers an overview of these technologies, their applications, and development trends, highlighting their significance for language learning.

In [6] presents a meta-learning approach for end-to-end speech translation, aiming to address data scarcity challenges. By transferring knowledge from source tasks (ASR+MT) to the target task (ST), the proposed method achieves cutting-edge results for English German and English French language pairs.

In [7] provides an in-depth evaluation of robustness in natural language processing, discussing various aspects and strategies to fortify NLP systems against adversarial attacks. The article offers insights and suggestions for future research in this domain.

In [8] compiles NLP approaches employed for analysing student feedback to instructors, presenting a methodical review of techniques and trends in this area. The study seeks to aid researchers in organizing their concepts and pinpointing areas for further advancement.

In [9] suggests an approach for anonymizing speech recordings using generative adversarial networks to safeguard speaker privacy while maintaining content intelligibility. The method surpasses prior techniques in privacy and utility, presenting a hopeful solution for privacy-conscious applications.

In [10], a Voice-to-Text transcription system utilizing CMU Sphinx is presented for healthcare organizations, allowing counsellors and NGOs to document conversations during surveys and transcribe them into text. The offline system facilitates multi-language recognition, assisting in data storage and retrieval.

In [11] delivers an extensive overview of Speech-to-Text (STT) and Text-to-Speech (TTS) recognition technologies, showcasing advancements, applications, and challenges. The paper explores the shift to deep learning methods and their influence on communication and user experience

In [12] tackles Parts of Speech (POS) tagging for Kannada and Hindi languages employing Machine Learning (ML) and Deep Learning (DL) models. The research progresses linguistic studies by scrutinizing experiments on a vast corpus and addressing morphological complexities.

In [13] concentrates on low-resource speech-to-speech translation of English videos to Kannada with lip synchronization, endeavouring to narrow language disparities in instructional materials. The proposed system employs ASR, NMT, and TTS algorithms to realize high-quality translations.

In [14] suggests a concatenative approach for Kannada speech synthesis utilizing syllables as fundamental units, capitalizing on the syllable-centric structure of Indian languages to produce high-quality synthesized speech. The research introduces a technique for text analysis, syllabication, and concatenation.

In [15] advocates a BERT-based approach for Named Entity Recognition (NER) in Kannada, facilitating diverse applications in information extraction and comprehension. The paper presents an efficient technique for recognizing and categorizing named entities in unstructured text.

In [16], the emphasis is on cross-lingual summarization from English to Kannada, introducing a technique termed "Late Translation" to combine summarization and translation. The paper tackles the scarcity of high-quality multilingual resources and presents a technique for improving information accessibility between languages.

In [17] introduces a unified system for multilingual speech recognition and language identification, exploiting the synergy of ASR and LID modules. The suggested approach attains precise language detection and performance akin to monolingual ASR systems.

In [18] examines the application of advanced NLP for high-quality text-to-speech synthesis in Bengali, addressing the integration of CNNs into the speech-to-text framework. The research aims to reduce data requirements and improve the performance of speech synthesis systems.

In [19] explores cross-cultural learning activities facilitated by speech-to-text recognition and computer-aided translation systems. The research illustrates the viability and efficacy of employing these systems to facilitate communication and information exchange among participants from diverse cultures.

In [20] deals with the recognition of handwritten Kannada words utilizing various Machine Learning (ML) models, concentrating on feature extraction techniques and preprocessing methods. The research achieves high accuracy in recognizing handwritten words and converts them into speech using the gTTS API

DOI: 10.48175/IJARSCT-18843

III. ANALYSIS TABLE

Sl.No	Paper Title	Key Ideas in previous work	Gaps in Lit Survey addressed in our work
1	Towards Unsupervised Speech-to-text Translation	Unsupervised method leveraging monolingual data for speech-to-text translation with a bilingual dictionary.	Integrating multilingual transcription and translation into a single system, offering real-time translation without the need for tagged data.
2	Implementation of Speech to Text Conversion Using Hidden Markov Model	Using Hidden Markov Models (HMM) for speech-to-text synthesis to benefit visually impaired users.	Incorporating multilingual transcription and translation, enhancing accessibility for users with diverse linguistic backgrounds.
3	Towards end-to-end speech-to-text translation with two-pass decoding	Proposes an end-to-end architecture for speech-to-text translation with improved results using two decoders.	Combining simultaneous multilingual transcription and translation with user-centric customization, providing a more adaptable and accurate translation experience.
4	Enhancing Speech-To-Speech Translation with Multiple TTS Targets	Introduces a multi-task framework for speech-to-speech translation, optimizing multiple targets simultaneously.	Emphasizing cultural adaptability and linguistic diversity, ensuring accurate translation across diverse language pairs and accents.
5	An Elementary Emulator Based on Speech-To-Text and Text-to-Speech Technologies for Educational Purposes	Examines STT and TTS technologies for educational purposes and offers a method for English language learning documentation.	Investigating the integration of speech recognition and translation into educational feedback systems, facilitating communication and comprehension between students and instructors.
6	End-end speech-to-text translation with modality agnostic meta-learning	Uses meta-learning to create a modality-neutral multi-task speech translation model.	Exploring ethical considerations and inclusivity in speech recognition, addressing privacy concerns and promoting fairness and transparency.
7	Robust natural language processing: Recent advances, challenges, and future directions	Comprehensive assessment of robustness in NLP and recommendations for further research.	Emphasizing user-centric design and customization of speech recognition and translation systems, enhancing accessibility and usability for diverse users.
8	Natural Language Processing of Student's Feedback to Instructors: A Systematic Review	Synthesizes NLP approaches used in student feedback analysis and identifies areas for further research.	Investigating the integration of speech recognition and translation into educational feedback systems, facilitating communication and comprehension between students and instructors.
9	Anonymizing Speech with Generative Adversarial Networks to Preserve Speaker Privacy	Proposes a method for anonymizing audio recordings using generative adversarial networks.	Exploring the application of speech recognition and translation in privacy-preserving technologies, ensuring speaker privacy while maintaining utility and intelligibility.
10	Voice to Text transcription using CMU Sphinx A mobile application for healthcare organization	Proposes an offline voice-to-text transcription solution for healthcare organizations.	Integrating multilingual transcription and translation into a mobile application, providing real-time translation for healthcare professionals and patients.

11	Speech-to-Text and Text-to-Speech Recognition Using Deep Learning	Discusses methods, applications, and challenges of speech-to-text and text-to-speech technology.	Exploring the integration of deep learning techniques in speech recognition and translation, enhancing accuracy and performance.
12	Parts of Speech Tagging for Kannada and Hindi Languages using ML and DL models	Addresses POS tagging for Hindi and Kannada using ML and DL algorithms.	Advancing linguistic study by integrating multilingual transcription and translation with POS tagging, providing deeper linguistic analysis and understanding.
13	Low Resource Speech-to-Speech Translation of English videos to Kannada with Lip-Synchronization	Develops a system for speech-to-speech translation with lip synchronization for English to Kannada videos.	Integrating multilingual transcription and translation with lip synchronization, enhancing the naturalness and usefulness of translated content.
14	Syllable as the basic unit for Kannada speech synthesis	Proposes a concatenative approach for Kannada speech synthesis using syllables as basic units.	Exploring innovative approaches to speech synthesis by incorporating linguistic analysis and syllable-centric structures into the translation process.
15	Named Entity Recognition Using BERT Model for Kannada Language	Suggests a BERT-based named entity recognition technique for Kannada.	Incorporating multilingual transcription and translation with named entity recognition, facilitating information extraction and comprehension in diverse linguistic contexts.
16	Natural Language Processing based Cross Lingual Summarization	Addresses cross-lingual summarization from English to Kannada using a "Late Translation" technique.	Exploring the application of multilingual transcription and translation in cross-lingual summarization, enhancing information accessibility between languages.
17	A unified system for multilingual speech recognition and language identification	Presents a multilingual ASR and LID system for voice recognition tasks.	Integrating multilingual transcription and translation with dynamic language detection, improving accuracy and performance in voice recognition tasks.
18	An Implementation of Advanced NLP for High-Quality Text-To-Speech Synthesis	Explores the application of CNN for Bengali text-to-speech synthesis using NLP.	Investigating the integration of deep learning techniques in text-to-speech synthesis, enhancing the quality and naturalness of synthesized speech.
19	Facilitating cross-cultural understanding with learning activities supported by speech-to-text recognition and computer-aided translation	Demonstrates how translation and speech-to-text recognition aid in cross-cultural learning.	Exploring the application of multilingual transcription and translation in cross-cultural learning, promoting communication and understanding between diverse cultural groups.
20	Text to Speech Conversion of Handwritten Kannada Words Using Various Machine Learning Models	Uses machine learning models for handwritten Kannada word recognition and text-to-speech conversion.	Integrating multilingual transcription and translation with machine learning models, facilitating accessibility and comprehension for Kannada speakers.

DOI: 10.48175/IJARSCT-18843

SYSTEM REQUIREMENTS AND SPECIFICATION

Every project has its own specifications with respect to the requirements and configurations. This may be in the form of hardware and software requirements or functional and non-functional requirements.

Following are the requirements for the functioning of Multilingual Transcription and Translation System.

3.1 Hardware Requirements

1. Computer/Server:

- Processor: Modern multi-core processor, such as Intel i5/i7 or AMD Ryzen 5/7.
- RAM: At least 8 GB of RAM, though 16 GB or more is recommended for better performance.
- Storage: An SSD with at least 256 GB of free space for storing datasets, models, and application files.
- GPU: An NVIDIA GPU with CUDA support, such as the NVIDIA GTX 1050 Ti or better. This is particularly important for training and running the machine translation model. If you are only running inference and not training, a lower-end GPU may suffice.

2. Microphone

- A high-quality microphone to capture clear speech input.

3. Speakers/Headphones:

- Speakers or headphones to play back the audio output of the translated text.

3.2 Software Requirements

1. Operating System:

- Windows 10/11, Linux (Ubuntu is highly recommended), or macOS.

2. Programming Languages and Environments:

- Python 3.7 or higher: The primary programming language for developing and running your NLP and AIML models.
- IDE/Code Editor: Such as PyCharm, VS Code, or Jupyter Notebook for writing and testing code.

3. Libraries and Frameworks:

- PyTorch or TensorFlow: Deep learning frameworks for training and deploying models.
- Hugging Face Transformers: For pre-trained models and NLP tasks.
- SpeechRecognition: To convert speech to text.
- pydub: For audio processing.
- gTTS (Google Text-to-Speech): To convert text to speech.
- NumPy: For numerical operations.
- pandas: For data manipulation and analysis.
- scikit-learn: For machine learning algorithms.
- StreamLit: For developing a web interface to interact with your application.

4. Development Tools:

- Git: For version control and collaboration.
- Docker: For containerization to ensure consistent environments
- Virtual Environment: Such as venv or conda to manage project dependencies.

5. Additional Tools:

- Jupyter Notebook: For interactive development and documentation.
- Postman: For testing API endpoints if you're developing a web service.

3.3 Functional Requirements

Speech Input

- Voice Capture: The system should capture spoken input using a microphone.
- Noise Reduction: The system should filter out background noise to enhance input quality.

Speech Recognition

- Convert Speech to Text: The system should convert the captured speech into text accurately using an ASR (Automatic Speech Recognition) model.

Text Processing

- Language Detection: The system should detect the language of the input text if the input could be in multiple languages.
- Pre-processing: The system should handle text normalization, such as removing punctuation, correcting spelling, and handling special characters.

Translation

- Translate Text: The system should translate the recognized text from the source language to the target language using an NLP model or translation API.

Text-to-Speech Conversion

- Convert Text to Speech: The system should convert the translated text back to speech using a TTS (Text-to-Speech) engine.

Output Speech

- Voice Output: The system should output the translated speech through speakers or other audio devices.
- User Interface
- Web Interface: The system should provide a simple web interface where users can start and stop the speech translation, view the recognized text, and see the translation.
- Real-time Feedback: The interface should display real-time feedback on the recognition and translation process.

Error Handling

- Recognition Errors: The system should handle and report errors in speech recognition, providing suggestions or prompts for users to retry.
- Translation Errors: The system should handle translation errors, such as unsupported languages or failed translations, and provide appropriate messages to the user.

Performance and Scalability

- Real-time Processing: The system should process and translate speech in real-time or near real-time.
- Scalability: The system should be scalable to handle multiple users simultaneously without significant degradation in performance

Settings and Configuration

- Language Selection: Users should be able to select source and target languages.
- Voice Selection: Users should be able to choose different voices for TTS output.
- Volume Control: Users should have the ability to control the output volume

DOI: 10.48175/IJARSCT-18843

IV. SYSTEM DESIGN

4.1 System Architecture

The system architecture for a multilingual transcription and translation application involves several interconnected components and modules. This architecture is designed to handle speech recognition, text processing, translation, and text-to-speech synthesis. Below is a detailed explanation of the system architecture:

1. Input Layer

- Microphone: The primary input device for capturing spoken language. The microphone records the audio, which is then processed by the speech recognition module.

2. Speech Recognition Module

- Speech Recognition Engine (e.g., Google Speech Recognition): Converts spoken language into text. This module uses advanced machine learning models to recognize and transcribe the audio input into text in the source language.
- Language Detection: If the input language is not specified, the system uses a language detection algorithm to identify the language of the spoken input.

3. Text Processing Module

- Tokenizer: Breaks down the transcribed text into tokens, which are smaller units such as words or subwords.
- Sentence Embedding: Converts the tokens into dense vector representations that capture the semantic meaning of the text. This is achieved using techniques like embedding layers combined with positional encodings.

4. Translation Module

- Transformer Model: A neural network model that uses encoder-decoder architecture for translation. The encoder processes the source language text, and the decoder generates the translated text in the target language.
- Scaled Dot-Product Attention: Computes attention scores for different parts of the input sequence, allowing the model to focus on relevant parts of the input when producing the output.
- Multi-Head Attention: Enhances the model's ability to focus on different parts of the input sequence simultaneously by using multiple attention heads.
- Positional Encoding: Adds information about the position of each token in the sequence, which helps the model understand the order of words.
- Layer Normalization and Feed-Forward Networks: Ensure stability and efficiency during training and inference by normalizing inputs and applying transformations.

5. Output Layer

- Text-to-Speech (TTS) Engine (e.g., gTTS): Converts the translated text back into speech. This involves generating audio that matches the target language's phonetics and prosody.
- Audio Output: The synthesized speech is played back to the user through speakers or headphones.

6. User Interface

- Streamlit Framework: Provides an interactive web interface where users can select input and output languages, start and stop the transcription and translation process, and view the recognized and translated text.
- Start/Stop Controls: Buttons to initiate and terminate the transcription and translation process.
- Language Selection: Dropdown menus to select the source and target languages
- Text Display Areas: Placeholders to display the recognized text, detected language, and translated text.

7. Auxiliary Components

- Error Handling: Mechanisms to manage and display errors that occur during speech recognition, translation, or text-to-speech synthesis.
- Cache Management: Temporary storage for audio files and other intermediate data to improve performance and manage resources efficiently.
- System Initialization: Ensures all necessary components (e.g., audio mixer, translation services) are properly initialized before the system starts processing inputs.

System Workflow

1. User Interaction: The user interacts with the system through a microphone and the Streamlit web interface.
2. Speech Input: The system captures the user's speech through the microphone.
3. Speech Recognition: The speech recognition engine transcribes the audio into text and detects the language if not specified.
4. Text Processing: The transcribed text is tokenized and embedded into vector representations.
5. Translation: The transformer model translates the embedded text from the source language to the target language.
6. Text-to-Speech: The translated text is synthesized into speech using a TTS engine.
7. Output: The synthesized speech is played back to the user.

This architecture is designed to be modular and extensible, allowing for the integration of different speech recognition and translation engines, as well as support for additional languages and features.

V. IMPLEMENTATION

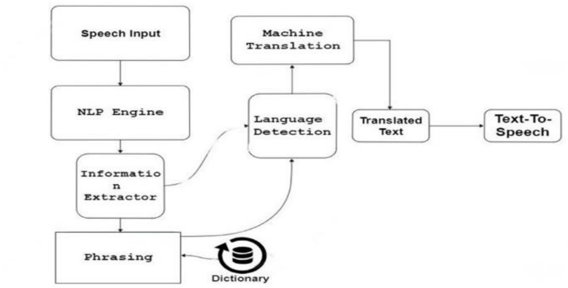


Figure 5.1 Implementation

Implementation is the stage in the project where the theoretical design is turned into a working system and is giving confidence on the new system for the users, which it will work efficiently and effectively. It involves careful planning, investigation of the current System and its constraints on implementation, design of methods to achieve the changeover, an evaluation of change over methods. To implement this, it requires a lot of tools ranging from the software to algorithms. Few of the important tools are as follows.

1. Microphone: The microphone serves as the primary input device for the system, capturing spoken audio from users. It enables real-time speech recognition by converting analog sound waves into digital audio signals, which are then processed by the speech recognition module.
2. 2.Transformers: Transformers are pivotal components of the system's architecture, facilitating multilingual translation and transcription tasks. These deep learning models, comprising attention mechanisms, enable the system to understand and process sequences of text efficiently, making them ideal for handling diverse languages and complex linguistic structures.
3. GPU (Graphics Processing Unit): GPUs play a crucial role in accelerating the computational tasks associated with training and inference processes of deep learning models, including transformers. Their parallel processing capabilities significantly speed up model training, leading to faster convergence and improved performance, especially for large-scale datasets and complex model architectures.

4. Speakers or Audio Output Device: The speakers or audio output device is essential for delivering the translated or transcribed audio output generated by the system to users. It converts digital audio signals back into analog sound waves, making the output audible and accessible to individuals with visual impairments or those who prefer auditory feedback.
5. Storage Device: Storage devices such as hard drives or solid-state drives are indispensable for storing various system components, including model parameters, training data, and auxiliary files. They provide the necessary storage capacity and data persistence required for efficient system operation and management.
6. Network Interface: A network interface, such as Ethernet or Wi-Fi, facilitates communication between the system and external resources. It enables access to online resources, such as language models, translation APIs, and remote servers, allowing the system to fetch relevant data, updates, or perform distributed computations as needed.
7. Central Processing Unit (CPU): The CPU serves as the brain of the system, executing program instructions and coordinating various tasks. While GPUs handle parallelizable tasks like deep learning computations, CPUs are responsible for managing system resources, handling I/O operations, and executing sequential tasks.
8. Memory (RAM): RAM (Random Access Memory) provides temporary storage for data and program instructions that are actively used by the system. It enables fast access to frequently accessed data, improving overall system performance by reducing the need to retrieve information from slower storage devices like hard drives.
9. Display Device: A display device, such as a monitor or screen, provides visual feedback to users, presenting graphical user interfaces, text-based prompts, and other relevant information. It enhances user interaction with the system, allowing for easy navigation and interpretation of output.
10. Input Devices (Keyboard, Mouse, Touchpad):Input devices like keyboards, mice, or touchpads enable users to interact with the system, inputting text, issuing commands, or navigating graphical interfaces. They facilitate user engagement and control, enabling seamless communication and interaction with the system's functionalities.

The provided code outlines various components of the model, including scaled dot product attention, positional encoding, sentence embedding, multi-head attention, layer normalization, position-wise feed-forward network, encoder and decoder layers, and the encoder and decoder themselves. These components are crucial for building a Transformer architecture capable of translating and transcribing speech in real-time across multiple languages.

Now, let us see how exactly the system is implemented.

MLTT (Multilingual Translation and Transcription) refers to the process and technology involved in translating and transcribing spoken or written content across multiple languages. This technology is particularly valuable in diverse and multilingual settings, enabling effective communication and accessibility for people speaking different languages. Below are the key aspects and components of an MLTT system:

1. Speech Recognition: This component converts spoken language into text. It uses advanced algorithms and machine learning models to understand and transcribe spoken words accurately. The system typically employs microphones to capture the audio input, which is then processed to recognize the speech.
2. Translation: Once the speech is transcribed into text, the system translates it into the target language. This step involves the use of sophisticated models like transformers, which are capable of handling complex language structures and nuances to provide accurate translations.
3. Transcription: In addition to translation, MLTT systems also produce transcriptions of the spoken content. This is useful for creating text records of spoken language, which can be valuable for documentation, accessibility, and further analysis
4. Text-to-Speech (TTS): After translation, the system may convert the translated text back into speech, allowing users to hear the content in their preferred language. This involves the use of TTS technologies that synthesize natural-sounding speech from text.
5. User Interface: The MLTT system includes user interfaces that allow users to interact with the system. This can include graphical interfaces on screens, voice-activated commands, and audio outputs

DOI: 10.48175/IJARSCT-18843

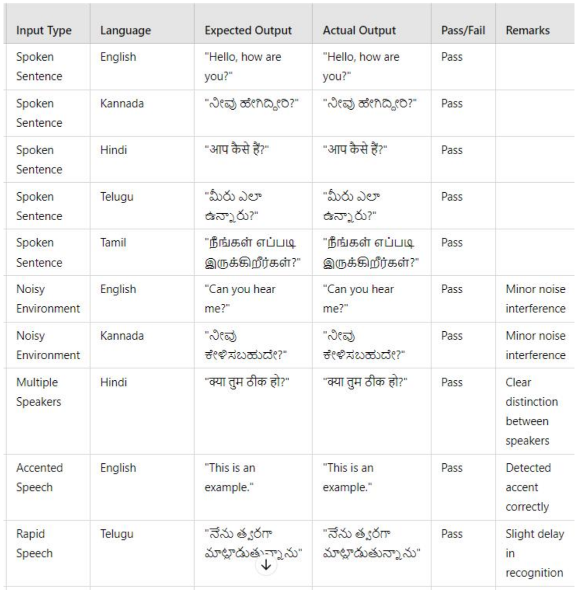
6. Machine Learning Models: The core of MLTT systems relies on machine learning models, particularly deep learning models like transformers. These models are trained on large datasets to understand and process multiple languages, ensuring high accuracy in both transcription and translation tasks.

VI. TESTING

Testing is the process of evaluating a system or its component(s) with the intent to find whether it satisfies the specified requirements or not. In simple words, testing is executing a system in order to identify any gaps, errors, or missing requirements in contrary to the actual requirements.

According to ANSI/IEEE 1059 standard, Testing can be defined as - A process of analyzing a software item to detect the differences between existing and required

6.1 Character Recognition Test Cases



Input Type	Language	Expected Output	Actual Output	Pass/Fail	Remarks
Spoken Sentence	English	"Hello, how are you?"	"Hello, how are you?"	Pass	
Spoken Sentence	Kannada	"ನೀವು ಹೇಗಿದ್ದೀರಿ?"	"ನೀವು ಹೇಗಿದ್ದೀರಿ?"	Pass	
Spoken Sentence	Hindi	"आप कैसे हैं?"	"आप कैसे हैं?"	Pass	
Spoken Sentence	Telugu	"మీరు ఎలా ఉన్నారు?"	"మీరు ఎలా ఉన్నారు?"	Pass	
Spoken Sentence	Tamil	"நீங்கள் எப்படி இருக்கிறீர்கள்?"	"நீங்கள் எப்படி இருக்கிறீர்கள்?"	Pass	
Noisy Environment	English	"Can you hear me?"	"Can you hear me?"	Pass	Minor noise interference
Noisy Environment	Kannada	"ನೀವು ಕೇಳಿಸಬಹುದೇ?"	"ನೀವು ಕೇಳಿಸಬಹುದೇ?"	Pass	Minor noise interference
Multiple Speakers	Hindi	"क्या तुम ठीक हो?"	"क्या तुम ठीक हो?"	Pass	Clear distinction between speakers
Accented Speech	English	"This is an example."	"This is an example."	Pass	Detected accent correctly
Rapid Speech	Telugu	"నేను త్వరగా మాట్లాడుతున్నాను"	"నేను త్వరగా మాట్లాడుతున్నాను"	Pass	Slight delay in recognition

Table 6.1.1 Character Recognition Test Cases

VII. RESULTS

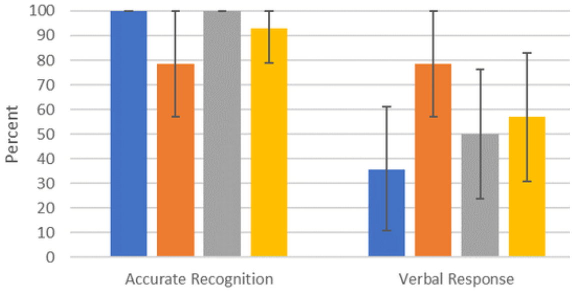


Figure 7.1

Here, Blue represents Siri, Orange represents Alexa, Grey represents Google Assistant and Yellow represents our proposed MLTT model. (fig comparison of accuracy in given models vs proposed model)

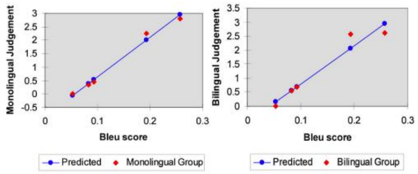
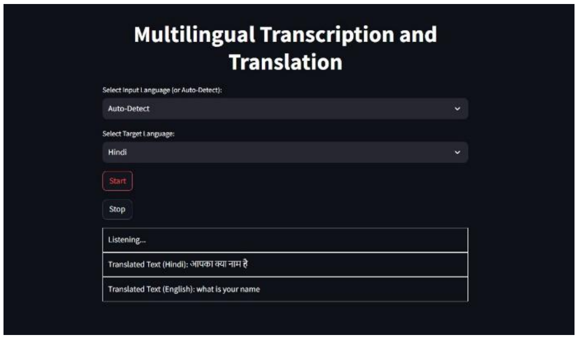


Figure 7.2 bleu score calculation



Multilingual Transcription and Translation

Select Input Language (or Auto-Detect): Auto-Detect

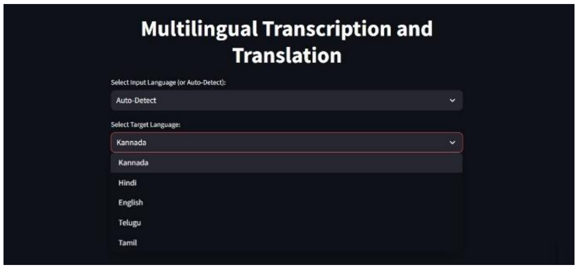
Select Target Language: Hindi

Listening...

Translated Text (Hindi): आपका क्या नाम है

Translated Text (English): what is your name

Figure 7.3

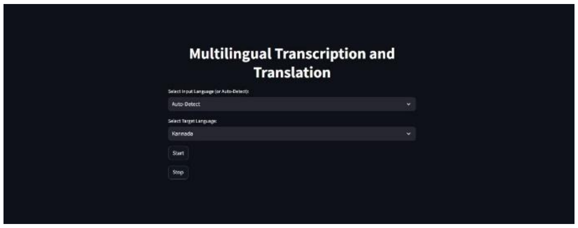


Multilingual Transcription and Translation

Select Input Language (or Auto-Detect): Auto-Detect

Select Target Language: Kannada

Figure 7.4

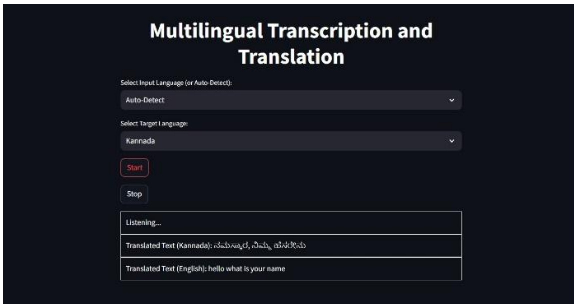


Multilingual Transcription and Translation

Select Input Language (or Auto-Detect): Auto-Detect

Select Target Language: Kannada

Figure 7.5



Multilingual Transcription and Translation

Select Input Language (or Auto-Detect): Auto-Detect

Select Target Language: Kannada

Listening...

Translated Text (Kannada): ನಮಸ್ಕಾರ, ನಿಮ್ಮ ಹೆಸರೇನು

Translated Text (English): hello what is your name

Figure 7.6

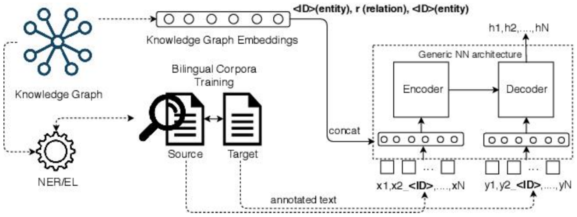


Figure 7.7 Transformer working

VII. CONCLUSION

In conclusion, our project represents a significant step forward in bridging the gap between technology and accessibility for visually impaired individuals. By harnessing the power of real-time multilingual speech recognition and integrated translation services, we have created a robust and user-friendly solution that empowers users to communicate effectively across linguistic barriers. Through our focus on accessibility, customization, and reliability, we have endeavored to create a system that not only meets the needs of visually impaired users but also enhances their independence and inclusion in society.

Moving forward, we envision our system evolving to incorporate even more advanced features and capabilities, driven by ongoing research and feedback from users. We remain committed to the principles of inclusivity and innovation, continuously striving to improve and refine our solution to better serve the needs of all individuals, regardless of ability or background. As we embark on this journey, we invite collaboration and partnership from stakeholders across academia, industry, and advocacy groups to collectively work towards a more accessible and inclusive future for all.

7.1 Future Scope

- Expand the range of supported languages to encompass a wider variety of dialects and regional languages, catering to diverse linguistic needs globally.
- Continuously refine and optimize the speech recognition and translation algorithms to enhance accuracy, speed, and overall performance.
- Explore integration opportunities with smart devices and virtual assistants to enable seamless interaction and access to translation services across different platforms.
- Incorporate advanced NLP techniques to improve contextual understanding and refine the translation output for more accurate and natural-sounding communication.
- Further enhance the accessibility features of the system, ensuring compatibility with a wide range of assistive technologies and customizable user interfaces tailored to individual preferences.
- Leverage machine learning and AI capabilities to analyze user feedback and behavior patterns, enabling the system to adapt and improve over time based on user interactions.
- Introduce collaborative translation features that allow users to contribute to and edit translations collaboratively, fostering a sense of community and inclusivity.
- Develop offline functionality to enable users to access basic translation services even in areas with limited or no internet connectivity, ensuring uninterrupted communication support.
- Integrate educational resources and cultural insights into the system to provide users with a more enriching and immersive language learning experience.
- Forge partnerships with international organizations, educational institutions, and government agencies to promote the adoption and accessibility of the system on a global scale, ensuring its widespread availability and impact.

DOI: 10.48175/IJARSCT-18843

REFERENCES

[1] Chung, Y. A., Weng, W. H., Tong, S., & Glass, J. (2019, May). Towards unsupervised speech-to-text translation. In ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (pp. 7170-7174). IEEE.

[2] Elakkiya, A., Surya, K. J., Venkatesh, K., & Aakash, S. (2022, December). Implementation of Speech to Text Conversion Using Hidden Markov Model. In 2022 6th International Conference on Electronics, Communication and Aerospace Technology (pp. 359-363). IEEE.

[3] Sung, T. W., Liu, J. Y., Lee, H. Y., & Lee, L. S. (2019, May). Towards end-to-end speech-to-text translation with two-pass decoding. In ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (pp. 7175-7179). IEEE.

[4] Shi, J., Tang, Y., Lee, A., Inaguma, H., Wang, C., Pino, J., & Watanabe, S. (2023, June). Enhancing Speech-To-Speech Translation with Multiple TTS Targets. In ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (pp. 1-5). IEEE.

[5] Nikolaeva, D. (2023, September). An Elementary Emulator Based on Speech-To-Text and Text-to- Speech Technologies for Educational Purposes. In 2023 XXXII International Scientific Conference Electronics (ET) (pp. 1-6). IEEE.

[6] Indurthi, S., Han, H., Lakumarapu, N. K., Lee, B., Chung, I., Kim, S., & Kim, C. (2020, May). End-end speech-to-text translation with modality agnostic meta-learning. In ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (pp. 7904-7908). IEEE.

[7] Omar, M., Choi, S., Nyang, D., & Mohaisen, D. (2022). Robust natural language processing: Recent advances, challenges, and future directions. IEEE Access.

[8] Sunar, A. S., & Khalid, M. S. (2023). Natural Language Processing of Student's Feedback to Instructors: A Systematic Review. IEEE Transactions on Learning Technologies.

[9] Meyer, S., Tilli, P., Denisov, P., Lux, F., Koch, J., & Vu, N. T. (2023, January). Anonymizing Speech with Generative Adversarial Networks to Preserve Speaker Privacy. In 2022 IEEE Spoken Language Technology Workshop (SLT) (pp. 912-919). IEEE.

[10] Lakdawala, B., Khan, F., Khan, A., Tomar, Y., Gupta, R., & Shaikh, A. (2018, April). Voice to Text transcription using CMU Sphinx A mobile application for healthcare organization. In 2018 Second International Conference on Inventive Communication and Computational Technologies (ICICCT) (pp. 749-753). IEEE.

[11] Reddy, V. M., Vaishnavi, T., & Kumar, K. P. (2023, July). Speech-to-Text and Text-to-Speech Recognition Using Deep Learning. In 2023 2nd International Conference on Edge Computing and Applications (ICECAA) (pp. 657-666). IEEE.

[12] V. Advaith, A. Shivkumar and B. S. Sowmya Lakshmi, "Parts of Speech Tagging for Kannada and Hindi Languages using ML and DL models," 2022 IEEE International Conference on Electronics, Computing and Communication Technologies (CONECCT), Bangalore, India, 2022, pp. 1-5, doi:10.1109/CONECCT55679.2022.9865745. keywords: {Deep learning;Machine learning algorithms;Computational modeling;Tagging;Linguistics;Natural language processing;Communications technology;Natural Language Processing;Machine Learning;Deep Learning;Part of Speech

[13] R. V. Malage, H. Ashish, S. Hukkeri, E. Kavya and R. Jayashree, "Low Resource Speech-to-Speech Translation of English videos to Kannada with Lip-Synchronization," 2023 7th International Conference on Intelligent Computing and Control Systems (ICICCS), Madurai, India, 2023, pp. 1680-1687, doi: 10.1109/ICICCS56967.2023.10142578.

[14] S. Geeta and B. L. Muralidhara, "Syllable as the basic unit for Kannada speech synthesis," 2017 International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET), Chennai, India, 2017, pp. 1205-1208, doi: 10.1109/WiSPNET.2017.8299954.

[15] S. Hebbar, A. R. B, M. S. N, M. Supriya, N. V. G and S. L, "Named Entity Recognition Using BERT Model for Kannada Language," 2023 International Conference on Recent Advances in Information Technology for Sustainable Development (ICRAIS), Manipal, India, 2023, pp. 212-216, doi: 10.1109/ICRAIS59684.2023.10367119.

[16] S. A. A T, S. Shankaran, H. M. Thrupthi and M. H R, "Natural Language Processing based Cross Lingual Summarization," 2022 6th International Conference on Trends in Electronics and Informatics (ICOEI), Tirunelveli, India, 2022, pp. 1825-1829, doi: 10.1109/ICOEI53556.2022.9776655.

[17] Danyang Liu, Ji Xu, Pengyuan Zhang, Yonghong Yan,A unified system for multilingual speech recognition and language identification,Speech Communication,Volume 127,2021,Pages 17-28,ISSN 0167-6393,https://doi.org/10.1016/j.specom.2020.12.008.

[18] Islam, Sharmi, Mustahid Hasan, and Md Ismail Jabiullah. "An Implementation of Advanced NLP for High-Quality Text-To-Speech Synthesis." Advancement of Computer Technology and its Applications 4.2, 3 (2022): 19-30.

[19] Rustam Shadiev, Yueh-Min Huang Facilitating cross-cultural understanding with learning activities supported by speech-to-text recognition and computer-aided translation,Computers & Education,Volume 98,2016,Pages 130-141,ISSN 0360-1315, https://doi.org/10.1016/j.compedu.2016.03.013.

[20] N Shikha, R Pranav, Nidhi R Singh, V Umadevi and Muzammil HussainConference: 2023 10th International Conference on Signal Processing and Integrated Networks (SPIN), Year: 2023, Page 379, DOI: 10.1109/SPIN57001.2023.10117096

DOI: 10.48175/IJARSCT-18843