iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 14 **Issue:** IV **Month of publication:** April 2026

DOI: https://doi.org/10.22214/ijraset.2026.81529

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue IV Apr 2026- Available at www.ijraset.com

Real-Time Translation and Pronunciation Assistance Using NLP

Mrs. P. Neelima[1], M. Sai Kusuma[2], M. Preethi[3], S. Neelima[4]
[1]*Assistant Professor, Department of Computer Science & Engineering Bapatla Women's Engineering College Bapatla, AP, India*
[2, 3, 4]*B.Tech Computer Science & Engineering , Bapatla Women's Engineering College Bapatla, AP, India*

Abstract: ***In a study of multilingual communication challenges,language barriers are identified as a major issue,especially in regions with diverse languages. This project presents a web-based speech translation and pronounciation assistance system.It uses Natural Language Processing(NLP) and cloud services to convert speech into text,translate it into different languages like English and Telugu, and provide audio output. The system supports both online and offline modes, making it useful even with low internet connectivity.It also includes regional and tribal languages to improve accessibility. This approach helps users understand pronounciation and meaning easily, improving communication and reducing language barriers.***
Index Terms: Real-Time Speech Translation, Pronounciation Assistance System, Natural Languge Processing(NLP), Multilingual Communication, Speech-to-Text Conversion, Text-to-Speech Synthesis, Language Translation System.

I. INTRODUCTION

The Speech translation and Pronounciation Assistance System was developed to address the growing challenges of communication across diverse languages. With increasing linguistic diversity, especially in multilingual regions and rural areas, individuals often face difficulties in understanding and conveying information due to language barriers. This project aims to overcome these challenges by providing an intelligent eb-based platform that enables real-time speech translation and improves accessibility to multilingual communication. The integration of advanced Natural Language Processing (NLP) and speech technologies represents a significant step toward enhancing communication efficiency and inclusivity.

The development of intelligent translation systems capable of processing speech, audio files, and text input has wide-ranging applications in education, healthcare, and public services. These systems not only translate languages but also provide pronounciation support through audio playback, helping users better understand meaning and usage. Traditional translation methods often depend on manual

interpretation or internet-based tools with limited flexibility, which can be time-consuming and less reliable in low-connectivity environments. By utilizing NLP techniques, Azure-based speech and translation services, and context-aware translation. This enhances user experience, reduces communication gaps, and supports effective multilingual interaction in real-world scenarios.

With increasing linguistic diversity and communication challenges across different regions, especially in rural and

multilingual communities, language barriers often lead to misunderstandings and limited access to essential information. Traditional translation methods rely heavily on internet connectivity and manual interpretation, which can be inefficient and unreliable in real-time scenarios. The primary motivation of this project is to develop an intelligent speech translation system that enables users to communicate effectively by providing accurate, real-time translation and pronounciation support, thereby improving accessibility and reducing communication gaps. Users face challenges in communication due to language barriers and lack of accessible translation tools. Traditional methods depend on manual translation or internet-based systems, which may be inaccurate and fail in low-connectivity environments. Existing solutions are not always user-friendly or support multiple input formats effectively. Hence, there is a need for a simple and reliable system that provides accurate real-time tranlation with offline support to improve communication.

II. LITERATURE SURVEY

In order to address the challenge of real-time multilingual communication, several research works have been studied and analyzed.

A study by Wu and Chen (2016) proposed a neural machine translation (NMT) model for real-time spoken language processing, which significantly improved translation accuracy by integrating speech recognition with translation modules. The system utilized deep learning techniques to capture contextual meaning in sentences, enabling more natural translations compared to traditional methods. However, the model required high computational resources and stable internet connectivity, limiting its usability in low-resource environments[1].

Recent research by Xiang Sun (2024) introduced an online traanslation recognition system based on Natural Language Processing (NLP), focusing on improving real-time processing efficiency and user interaction through web-based platforms. The system emphasized faster response times using optimized algorithms and cloud-based services. Despite its efficiency, the system was highly dependent on internet availability and faced challenges in handling low-resource or regional languages[2].

Recent work by BWEC students and staff (2025) proposed an NLP-powered offline speech-to-speech translation system, addressing the limitation of internet dependency. The system enabled translation without requiring continuous connectivity, making it suitable for rural and remote areas. It also emphasized accessibility for regional and tribal language users. However, the system had limitations in terms of the number of supported languages and showed reduced accuracy when handling complex sentences or diverse dialects[3].

From the above studies, it is evident that while existing systems provide significant advancements in translation accuracy and real-time processing, challenges such as internet dependency, limited language support, and computational complexity still exist. The proposed system aims to overcome these limitations by providing an efficient, user-friendly, and scalable solution that supports real-time translation and pronounciation asistance in both online and offline modes[4].

III. PROBLEM STATEMENT

Effective communication across multiple languages remains a significant challenge, particularly in multilingual and low-resource environments. Language barriers, coupled with the lack of accessible and reliable translation tools, hinder the exchange of information in domains such as education, healthcare, and public services. Conventional approaches rely heavily on manual interpretation or internet-based translation systems, which are often time-consuming, prone to inaccuracies, and unreliable in low-connectivity scenarios. Moreover, existing solutions provide limited support for diverse input formats such as speech, audio, and text, and frequently lack adequate pronounciation assistance.

These limitations highlight the need for an intelligent, user-friendly system capable of delivering accurate real-time translation along with pronounciation support. Additionally, the system should operate efficienctly in both online and offline modes and support regional and tribal languages to ensure inclusivity and improved accessibility.

IV. PROPOSED SYSTEM

The proposed Real-Time Translation And Pronounciation Assistance Using NLP.

A. System Components

The proposed system consists of the following components:

- Input Processing Module: Handles user input such as speech, audio files, and text.
- Speech Recognition Module(ASR): Converts speech input into text.
- Text Processing Module(NLP): Processes and refines the text for better accuracy.
- Language Translation Module: Translates Text into the target language.
- Text-to-Speech Module(TTS): Converts translate text into audio ouput.
- User Interface Module: Enables user interaction and displays results.

B. Main Features

- Real-time speech and text translatiom
- Pronounciation assistance using audio output
- Multiple input methods (speech, audio, text)
- Multilingual support including regional/tribal languages
- Online and offline functionality
- User-friendly interface
- Fast and accurate processing

C. Technologies Used

- Natural Language Processing(NLP)
- Automatic Speech Recognition(ASR)
- Text-to-Speech(TTS)

- Azure Speech Services / Translation APIs
- Python (Backend Development)
- Flask / Node.js (Web Framework)
- HTML, CSS, JavaScript (Frontend)
- MongoDB / Database (Data Storage)

V. SYSTEM ARCHITECTURE

The System architecture is composed of the following major components.

A. User Interface

A web-based interface that allows users to input speech, audio, or text and view the translated output easily. It ensures simple and user-friendly interaction.

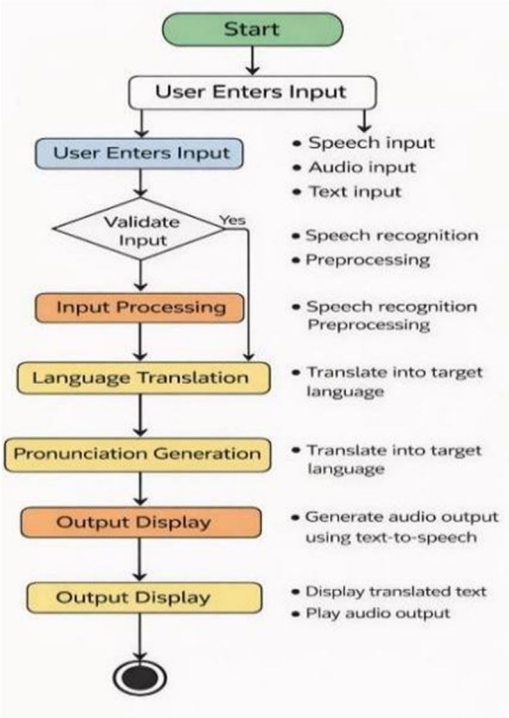


Fig. 1. System Architecture of the Real-Time Translation And Pronounciation Assistance Using NLP

B. Backend Processing

The backend is implemented using technologies such as python(Flask) or Node.js to handle request processing, API integration, and communication between different modules efficiently.

C. Database(Language Dataset)

Stores structured data including language models, vocabulary, and user input records. It supports multilingual processing, including regional and tribal languages. It helps in faster processing by storing required data. It also enables offline functionality.

D. Input Module

Handles both text and voice input from users. It captures user data and forwards it to the processing modules for further operations.It supports speech, audio files, and text input.

E. Input Processing Module

Performs validation and preprocessing of the input data such as noise removal and text cleaning to improve accuracy. This step improves overall system performance.

F. Speech Recognition Module (ASR)

Converts spoken input into text format using speech recognition techniques. It supports voice-based interaction. It works even with different accents and noise levels.

G. Text Processing Module (NLP)

Processes and analyzes the text using NLP techniques like tokenization and normalization for better translation. It helps in understanding sentence structure. It improves translation accuracy.

H. Language Translation Module

Translates the processed text into the target language using translation models or APIs. It ensures accurate and meaningful translation. It supports multiple languages.

I. Text-to-Speech Module (TTS)

Converts translated text into speech output, providing pronounciation assistance. It generates clear and natural speech. It improves user experience.

VI. METHODOLOGY

The Real-Time Translation and Pronounciation Assistance System is developed by following a structured workflow to ensure efficient and accurate multilingual communication.

The methodology includes the following steps:

1) Data Acquisition: The system utilizes speech datasets, text corpora, and pre-trained language models obtained from cloud platforms and open-source NLP resources. These datasets contain multilingual text, speech samples, and language mappings required for speech recognition and translation. The data is stored in structured formats such as JSON or text files for efficient processing.

2) Model Processing and Integration: The system integrates speech recognition and translation models using APIs such as Azure Speech Services and Translator. Speech input is converted into text using speech-to-text models, and the extracted text is processed using Natural Language Processing (NLP) techniques. These models are capable of handling multiple languages and provide accurate translation outputs.

3) Features Extraction and Text Processing: Input text is processed using NLP techniques such as tokenization, normalization, and noise removal. This step improves the quality of input data and ensures better translation accuracy. Linguistic features such as words, phrases, and sentence structure are extracted to generate meaningful and context-aware translations.

4) System Integration: A web-based interface is developed using frameworks such as Flask or Express.js to integrate the backend processing with the user interface. The system supports multiple input methods including speech, audio files, and text. The processed output id displayed on the interface along with audio playback for pronounciation assistance.

5) Testing and Validation: The system is tested using different types of inputs such as live speech, recorded audio, and text data. Performance is evaluated based on accuracy, response time, and usability. The results show that the system provides real-time translation with minimal delay and acceptable accuracy across multiple languages.

VII. IMPLEMENTATION

The implementation of the speech Translation and pronounciation Assistance System involved key steps to ensure efficient functionality and smooth integration of components:

1) System Development Environment: The system id developed using python as the primary programming language for backend processing. Web frameworks such as Flask/Express.js are used to build the application and manage communication between frontend and backend. The frontend interface is designed using HTML, CSS and JavaScript tomprovide a user-friendly experience.

2) Speech Recognition Module: The speech recognition functionality is implemented using APIs such as Azure Speech Services or offline models like Vosk. This module captures voice input from the user and converts it into text format. Audio preprocessing techniques such as noise reduction and format conversion are applied to improve accuracy.

3) Text Processing Module: Once the speech is converted into text, it is processed

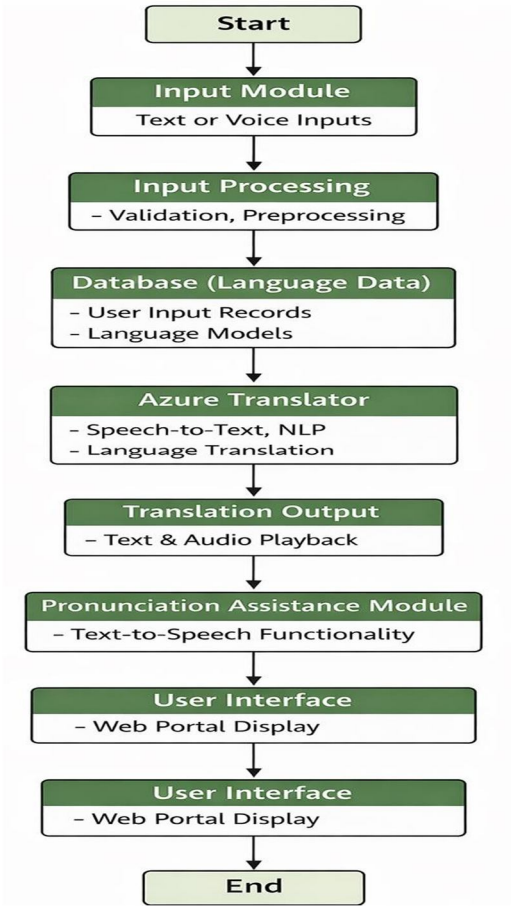


Fig. 2. Implementation

using NLP techniques. Libraries such as NLTK are used for tokenization, text cleaning, input text is structured properly

4) Language Translation Module: The processed text is translated into the target language using translation APIs like Azure Translator. The system supports multiple languages including English and Telugu. The translation module ensures context-aware and meaningful output.

5) Text-to-Speech Module: To assist users in pronounciation, the translated text is converted into speech using text-to-speech (TTS) technology. The module generates audio output, allowing users to hear the correct pronounciation of translated content.

6) Web Framework & User Interface: A web-based framework such as Flask or Express.js was used to develop a responsive and user-friendly interface, allowing users to input speech, audio, or text and view translated results easily.

7) Backend Processing & API Integration: Python / Node.js was used to process input data, perform speech recognition, translation, and text-to-speech using Azure APIs, along with database integration for managing language data efficiently.

8) Testing and Execution: The system is tested with different types of inputs such as live speech, recorded audio, and text. The implementation ensures real-time processing with minimal delay. The results confirm that the system works efficiently in both online and offline modes.

VIII. RESULT AND ANALYSIS

The proposed system, Real-Time Translation and Pronounciation Assistance using NLP, was successfully implemented and tested with multiple input formats such as speech, audio files, and text. The system demonstrated effective performance in real-time speech recognition, laguage translation, and pronounciation assistance.

A. Signup and Login Screens

The signup screen allows new users to create an account easily by entering basic details, while the login page provides secure access to existing users. These screens ensure authentication and personalized usage of the system. The simple and clean design improves user experience and makes the system accessible even for non-technical users.

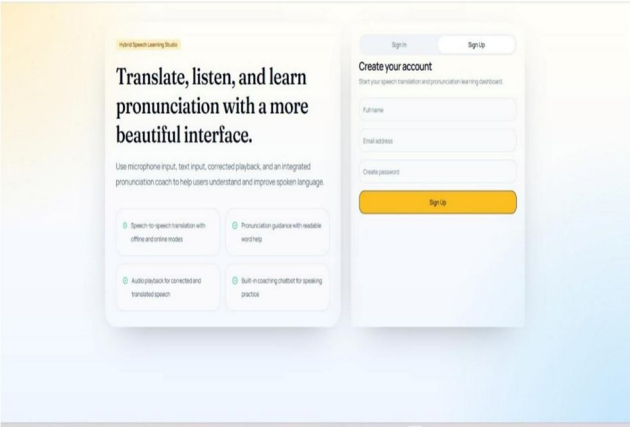


Fig.3. Signup and Login Page

B. Dashboard

The dashboard acts as the central interface of thesystem where users can access all features such as speech input, text translation, and pronounciation assistance. It provides a well-organized layout that allows users to choose different input methods (speech, audio, or text). This screen reflects the flexibility and user-friendly nature of the system.

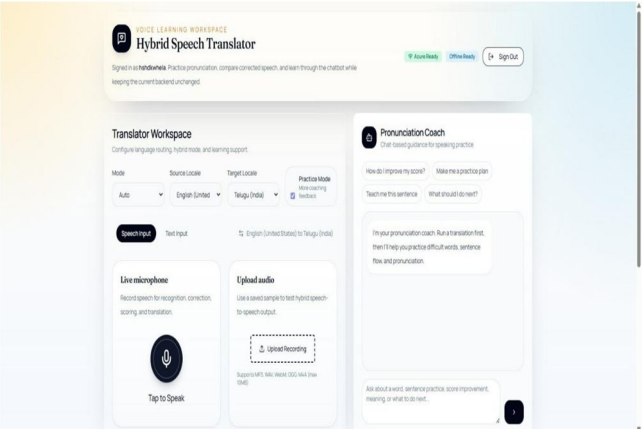


Fig.4. Dashboard

C. Output Screen

This screen displays the translated text , corrected content, and speech output results of the system. It also shows accuracy metrics and audio playback options for better understanding and analysis.

This screen displays the translated text , corrected content, and speech output results of the system. It also shows accuracy metrics and audio playback options for better understanding and analysis.

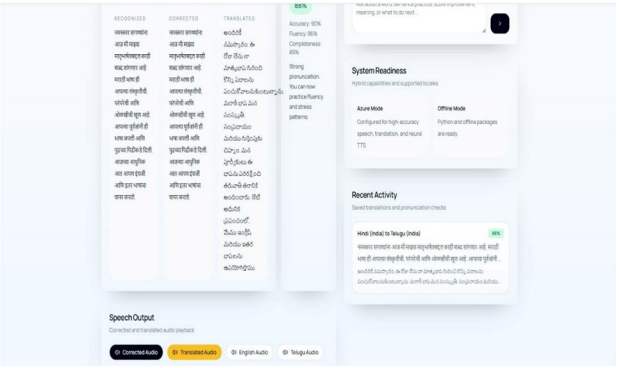


Fig. 5. Output Screen

D. Coaching Panel

This screen shows the pronounciation coaching feature that guidesusers to improve their speaking skills.

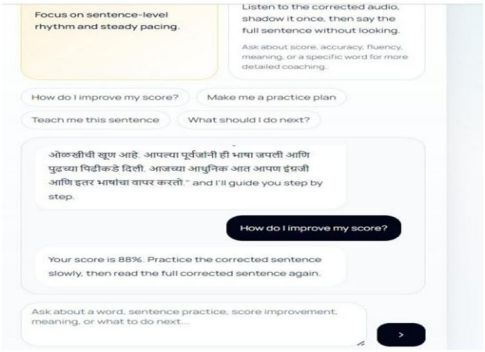


Fig.6. Coaching panel

E. Text Translation Screen

This screen shows the text input translation feature of the system with real-time results.It allows users to enter text, view translated output, and access pronounciation feedback.

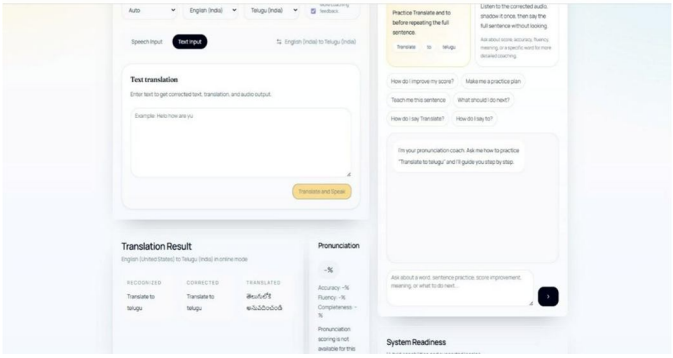


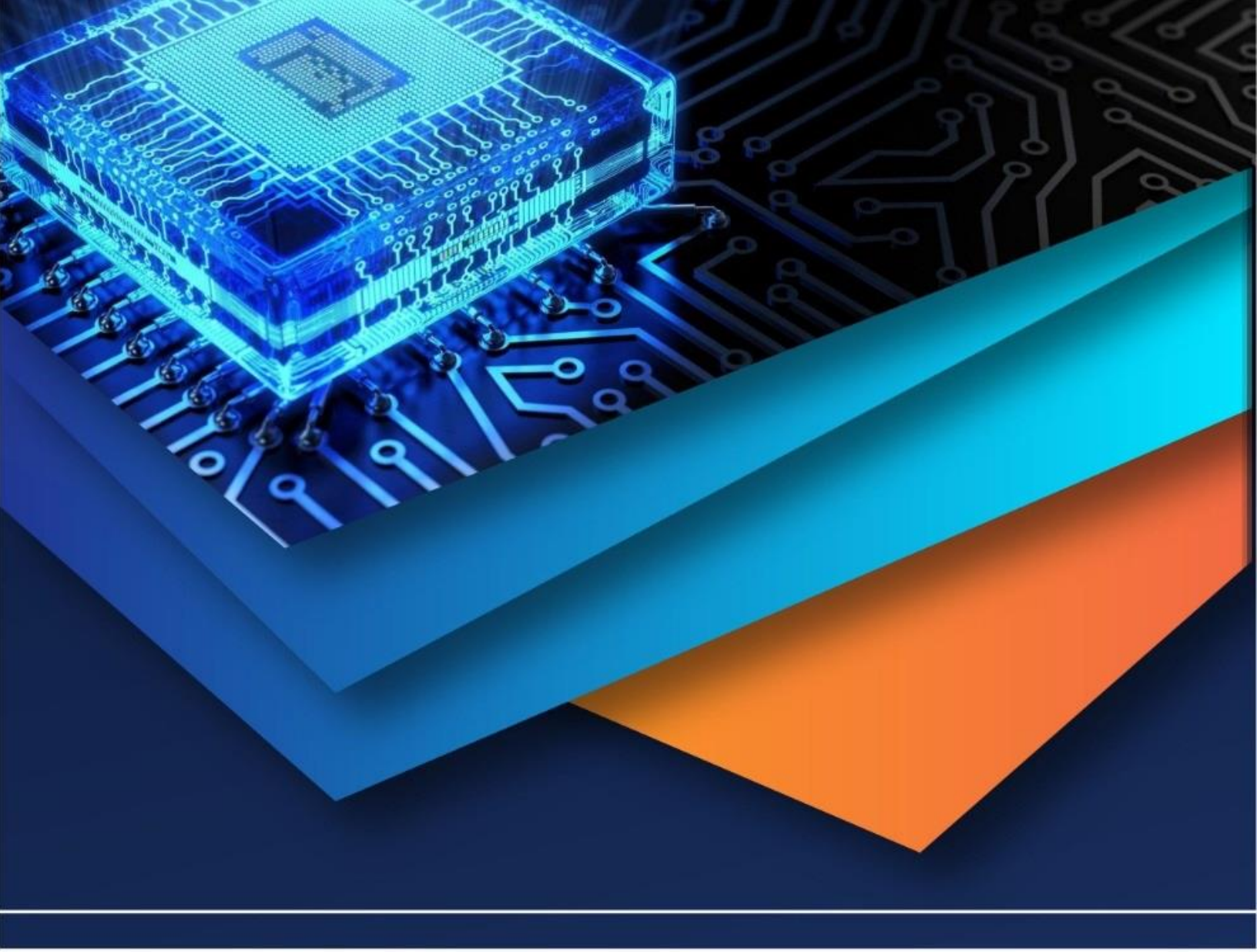
Fig. 7. Text translation

IX. CONCLUSION

The Speech Translation and Pronounciation Assistance System has successfully made significant strides in improvving multilingual communication and accessibility across various domains. By integrating advanced AI technologies, the system delivers real-time speech recognition, accurate language translation, and clear pronounciation support. This capability enables users to understand and communicate effectively across different languages, making it highly useful in areas such as education, healthcare, and public services. Overall, the project demonstrates the effectiveness of AI-driven solutions in simplifying multilingual communication and future advancements in speech and language technologies.

REFERENCES

[1] [1] D. Bahdanau, K. Cho, and Y. Bengio, “Neural machine translation by jointly learning to align and translate,” arXiv preprint arXiv: 1409.0473, 2014.
[2] A. Vaswani et al., “Attention is all you need,” arXiv preprint arXiv:1706.03762, 2017.
[3] A. Graves, A. Mohamed, and G. Hinton, “Speech recognition with deep recurrent neural networks,” in Proc. ICASSP, 2013, pp. 6645-6649.
[4] W. Xiaong et al., “The Microsoft 2017 conversational speech recognition system,” in Proc. ICASSP, 2018,pp. 5934-5938.
[5] T. Brown et al., “Language models are few-shot learners,” arXiv preprint arXiv:2005.14165, 2020.
[6] OpenAI, “Whisper: Robust speech recognition via large-scale weak supervision,” arXiv preprint arXiv:2212.04356, 2022.
[7] J. Devlin, M. Chang, K. Lee, and K. toutanova, “BERT: Pre-training of deep bidirectional transformers for language understanding,” in Proc. NAACL, 2019.
[8] M. Schuster and K. Nakajima, “Japanese and Korean voice search,” in Proc. ICASSP, 2012.
[9] S. Yun and Y. Lee, “Multiligual speech-to-speech translation system for mobile devices,” International Journal of Engineering Research & Technology, vol. 8, no. 6, 2019.
[10] D. Y. Fujita et al., “End-to-end neural speaker diarization with self=attention,” arXiv preprint arXiv:1909,06247, 2019.
[11] A. Radford et al., “Improving language understanding by generative pre-training,” OpenAI, 2018.
[12] I. Sutskever, O. Vinyals, and Q. V. Le, “Sequence to sequence learning with neural networks,” in Proc. NIPS, 2014.
[13] K. Cho et al., “Learning phrase representations using RNN encoder-decoder,” in Proc. EMNLP, 2014.
[14] Microsoft, “Azure Speech Services Documentation,” Microsoft Docs, 2024.
[15] Google, “Google Speech-to-Text API Documentation,” Google Cloud Docs, 2024.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY