Review

Multilingual Speech and Text Translation for Indian Regional Languages

Navya Sharma , Uma Chauhan * , Sandeep Kumar

Posted Date: 7 March 2025

doi: 10.20944/preprints202503.0478.v1

Keywords: machine learning; multilingual; speech-to-text; Indian languages; IndicTrans2; text-to-speech

 doi:10.20944/preprints202503.0478.v1

Review

Multilingual Speech and Text Translation for Indian Regional Languages

Navya Sharma, Uma Chauhan * and Sandeep Kumar

Department of Computer Science and Engineering, Sharda School of Engineering and Technology, Sharda University, Greater Noida, India

* Correspondence: uc1005uma@gmail.com

Abstract: In a multilingual country like India Language barrier between communities present significant challenges for effective cross-lingual communication. The development of robust translation systems that handle both text and speech across these languages is essential for fostering inclusivity and accessibility. In order to address these challenges this study aims to the exploration of multilingual speech and text translation using the IndicTrans2 model, a state-of-the-art transformer-based model tailored for Indian languages. The proposed application leverages IndicTrans2's capabilities to seamlessly translate text and spoken content between 22 Indian languages, enhancing accessibility and inclusivity. This solution is specifically designed to bridge communication barriers for speakers of different Indian languages, offering real-time translations that support both speech-to-text and text-to-speech functionalities and make our code available on GitHub. The system architecture, underlying model mechanics, training methodologies, and evaluation metrics are thoroughly discussed in paper. Future work aims to expand the model's capabilities by incorporating additional dialects and optimizing real-time speech translation performance.

Keywords: machine learning; multilingual; speech-to-text; Indian languages; IndicTrans2; text-to-speech

1. Introduction

India is rich in languages, with 22 officially recognized languages Figure 1 and hundreds of languages spread across different regions. While creating culture, this diversity in language also creates serious problems in terms of communication, especially where communication between different speakers is required. Crossing over these languages is important to improve relationships, improve access to services, and promote economic and educational development [1].

Existing multilingual translation tools have made strides in eliminates language barriers, but many solutions fall short when it comes to accurately and effectively handling the complexities of Indian languages. Problems such as varied scripts, context-based meanings, dialectal differences, and low-resource languages present challenges that conventional models struggle to address [2]. To solve these problems, robust, adaptable, and context-aware translation systems are crucial [3].

This research focuses on leveraging the IndicTrans2 model, a state-of-the-art transformer-based model tailored for Indian languages, to develop a multilingual speech and text translation system. IndicTrans2 is designed to understand and translate between 22 Indian languages with high accuracy, making it an optimal choice for this endeavor. By incorporating IndicTrans2 into an application that supports both speech-to-text and text-to-speech translation, this research aims to provide an innovative solution that facilitates real-time communication between speakers of different Indian languages.

The objective of this research is to outline the development, implementation, and performance evaluation of the proposed multilingual translation application. The paper will detail the data sources and pre-processing methods, the integration of IndicTrans2, and the overall architecture of the system. Additionally, we will present empirical results that demonstrate the system's effectiveness and discuss potential use cases in education, healthcare, and public administration.

 doi:10.20944/preprints202503.0478.v1

Figure 1. Different Languages in Different Indian Regions [4].

2. Related Work

In India with over 1 billion people speaking 22 languages, India's linguistic diversity creates unique challenges and opportunities for multilingualism and translation. Recent research has focused on developing flexible, easy-to-use machine translation (MT) and speech-to-speech machine translation (SSMT) systems to bridge the gap and facilitate communication across Indian languages.

2.1. Challenges in Multilingual Translation for Indian Languages

- **Data Scarcity and Quality:** One of the biggest problems in developing effective MT systems for Indian languages is the lack of large, high-quality parallel corpora. This scarcity hampers the ability to train robust models that can handle the nuances of each language [5].
- **Transliteration and Code-Mixing:** Indian languages often appear in transliterated forms or code-mixed with English, especially in informal settings like social media. Current state-of-the-art multilingual language models (LMs) struggle to handle these variations effectively [6].
- **Diverse Linguistic Landscape:** The linguistic diversity in India, with languages from four major language families, requires MT systems to be highly adaptable and capable of handling multiple scripts and dialects [7].

2.2. MuRIL: Multilingual Representations for Indian Languages

MuRIL is a multilingual language model specifically built for Indian languages. It is trained on large amounts of Indian text corpora, including both translated and transliterated document pairs,

 doi:10.20944/preprints202503.0478.v1

which serve as supervised cross-lingual signals. MuRIL significantly outperforms multilingual BERT (mBERT) on various tasks in the challenging cross-lingual XTREME benchmark, demonstrating its efficacy in handling transliterated data [8].

2.3. Hybrid Models and Neural Networks

Systems like KHiTE utilize a combination of Hidden-Markov-Models, Artificial Neural Networks, Deep Neural Networks, and Convolutional Neural Networks to process cross-lingual speech and translate it into monolingual text for languages such as Kannada, Hindi, Telugu, and English. This approach involves pre-processing steps like noise removal and speech splitting to obtain phonemes, which are then mapped to text using trained corpora [9].

Neural Machine Translation (NMT) models, particularly those using sequence-to-sequence models with encoder-decoder attention mechanisms, have shown significant improvements in translating languages like Telugu to English and vice versa [10].

2.4. VAKTA-SETU: Speech-to-Speech Machine Translation

VAKTA-SETU is a deployment-ready SSMT system for English-Hindi, English-Marathi, and Hindi-Marathi language pairs. It integrates Automatic Speech Recognition (ASR), Disfluency Correction (DC), Machine Translation (MT), and Text-to-Speech Synthesis (TTS) models. The system is designed for various use cases, including government initiatives, tourism, the judiciary, and agriculture, demonstrating its practical utility in real-world scenarios [5,11].

3. Proposed Work

The proposed work focuses on developing a robust multilingual speech and text translation system tailored specifically for 22 Indian languages using the IndicTrans2 model. The system aims to facilitate seamless communication across diverse linguistic groups by providing real-time translation of both spoken and written language.

3.1. System Architecture

The system architecture Figure 2 of the IndicTrans2-based translation application, built with Streamlit, is designed to facilitate seamless language translation and text-to-speech capabilities. The architecture is structured into multiple interconnected components, each fulfilling a specific function in the workflow.

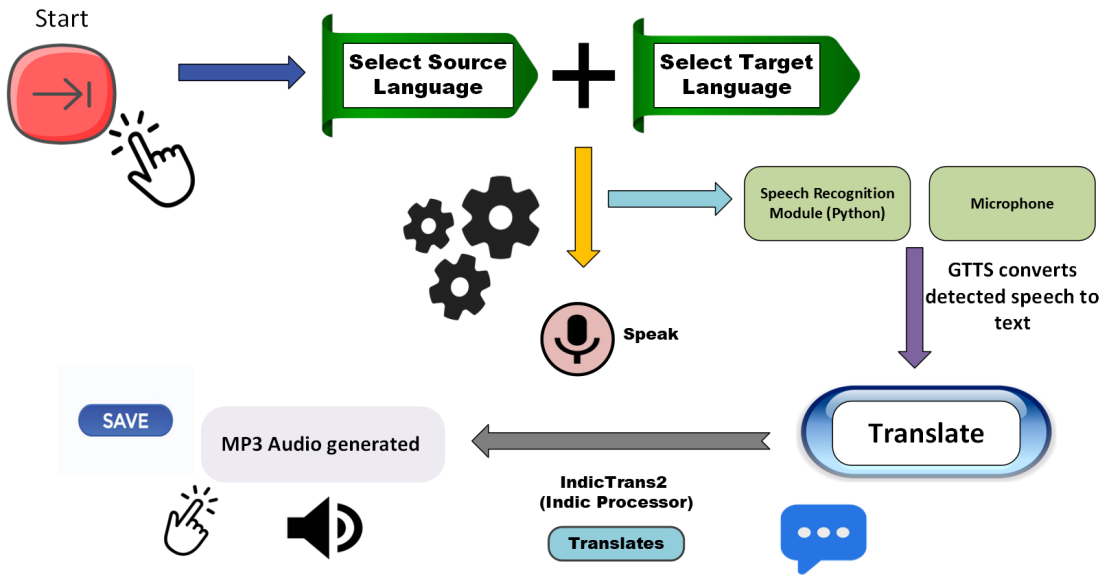

Figure 2. System Architecture Workflow.

 doi:10.20944/preprints202503.0478.v1

At the core of the system the User Interface Layer is implemented using Streamlit. This layer allows users to input either text or speech, choose source and target languages, and trigger translation or audio playback. Then The speech Recognition Module plays a crucial role by enabling voice input from users.

The Processing Layer integrates the Translation Engine, which utilizes the AI4Bharat IndicTrans2 model. This model, loaded with torch and transformers libraries, processes user input text and generates translations.The IndicProcessor is used for preprocessing and postprocessing tasks to ensure that input and output formats are optimized for translation accuracy.

Once the translation is complete, the Audio Generation Module steps in for text-to-speech conversion. It uses the gTTS (Google Text-to-Speech) library to generate an audio file from the translated text.

3.2. Components of System

3.2.1. **Text-to-Speech - TTS**

The TTS process involves converting a given input text sequence into a corresponding speech waveform.This feature is critical for providing audio feedback, allowing users who prefer auditory output or have visual impairments to understand the translation. Mathematically, this can be expressed as:

$$y = \text{TTS}(x)$$

where:

- x represents the input text sequence.
- y represents the generated speech waveform (audio output).

The TTS function $\text{TTS}(\cdot)$ takes the input text and applies a sequence of transformations including text normalization, phoneme conversion, prosody modeling, and waveform synthesis.

3.2.2. **Speech Recognition Module**

The Speech Recognition Module's task is to map an audio signal to a corresponding text output by finding the most probable sequence of words. This module plays a critical role in ensuring that spoken language inputs are accurately understood, even in complex linguistic environments with varied accents and background noise. This can be modeled as:

$$\hat{x} = \arg\max_{x} P(x|y)$$

where:

- $\hat{x}$ represents the most likely transcribed text.
- $P(x|y)$ is the conditional probability of the text x given the audio input y.

The speech recognition process typically follows this equation, employing algorithms such as the Hidden Markov Model (HMM) or deep neural networks to maximize the likelihood $P(x|y)$ using a combination of acoustic and language models.

3.2.3. **Speech-to-Text - STT**

The STT process involves converting an audio signal into a text sequence. This module is essential for capturing spoken input from the user and translating it into text that can be further processed by the translation engine. This can be expressed as:

$$x = \text{STT}(y)$$

where:

- y represents the input speech waveform (audio input).
- x represents the transcribed text output.

 doi:10.20944/preprints202503.0478.v1

The function STT$(\cdot)$ utilizes acoustic models, feature extraction (e.g., MFCCs), and language models to decode the audio signal and generate a text transcription.

3.3. User FLow Diagram

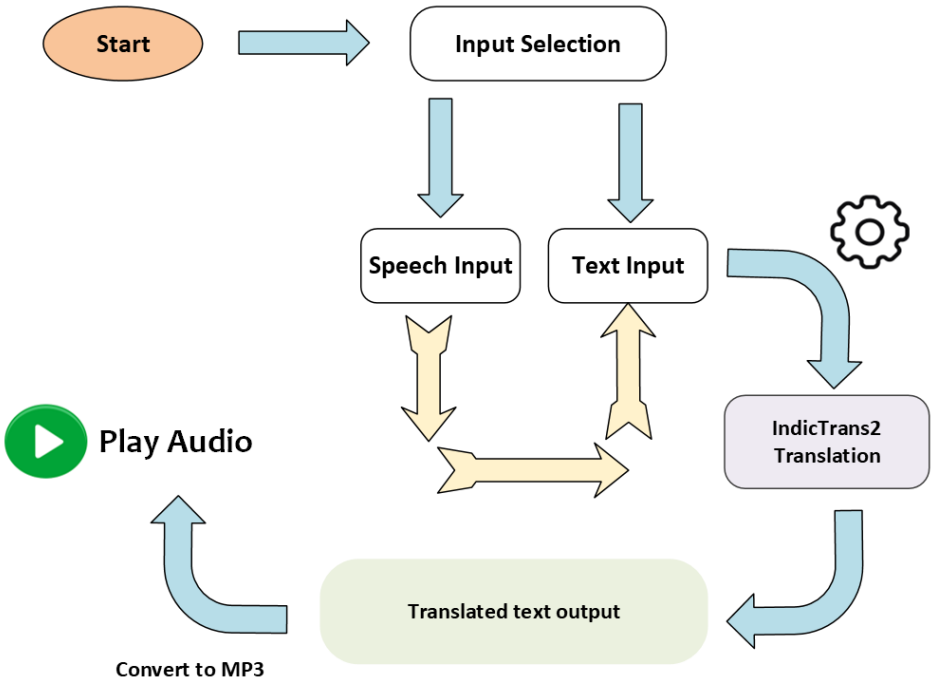

Figure 3. User Flow Diagram.

4. IndicTrans2 Model Overview

4.1. Model Architecture

The IndicTrans2 model is built upon a transformer-based architecture optimized for multilingual translation. It incorporates various modules such as encoder-decoder stacks, attention mechanisms, and positional encoding to effectively learn and translate between 22 Indian languages. The encoder captures the semantic and syntactic structure of the source language, while the decoder generates the translation in the target language with attention guiding the alignment between input and output sequences [12].

4.2. Training Methods

The model is trained using a large, curated dataset comprising parallel corpora of Indian languages. Training employs techniques like data augmentation, subword tokenization (e.g., SentencePiece), and back-translation to enhance model robustness. The optimization process uses Adam or AdamW optimizers with a custom learning rate schedule. Multi-task learning is employed to improve cross-lingual transfer capabilities, where related language pairs are trained together to boost performance [12].To enhance the model's generalization and translation accuracy, several key techniques are employed:

- **Data Augmentation:** To improve model robustness, various data augmentation methods are applied. This includes synthetically generating additional training data by introducing controlled variations in the existing text to expand the dataset and expose the model to a broader set of linguistic patterns.
- **Subword Tokenization (e.g., SentencePiece):** The training pipeline incorporates subword tokenization techniques like SentencePiece to break down words into subword units. This approach

 doi:10.20944/preprints202503.0478.v1

helps handle out-of-vocabulary words effectively, improves language modeling, and ensures efficient handling of morphologically rich languages.

- **Back-Translation:** Back-translation is used to augment training data by translating sentences from the target language back into the source language and then re-translating them into the target language. This technique boosts the training dataset, particularly for low-resource language pairs, and enhances the model's capability to produce fluent and accurate translations.
- **Optimization Strategy:** The model training process uses state-of-the-art optimization algorithms such as Adam or AdamW, tailored with a custom learning rate schedule. This ensures that the model converges efficiently and avoids common pitfalls like vanishing or exploding gradients during training.
- **Multi-Task Learning:** To further improve cross-lingual transfer capabilities, the training process employs multi-task learning. Related language pairs are trained together, allowing shared learning representations across linguistically similar languages. This joint training approach enhances performance on both high-resource and low-resource languages by transferring knowledge between them.

4.3. Evaluation Metrics

The performance of the IndicTrans2 model is evaluated using established translation metrics such as BLEU (Bilingual Evaluation Understudy) and TER (Translation Edit Rate). These metrics measure the quality of the translations by comparing the generated outputs to human translations. Additionally, human evaluation is conducted for qualitative analysis, taking into account fluency, adequacy, and linguistic accuracy across different Indian languages [12].

- **BLEU (Bilingual Evaluation Understudy Score):** BLEU is a widely used metric for evaluating the quality of machine-translated text by comparing it to one or more human-generated reference translations. It calculates the precision of n-grams between the model's output and the reference, penalizing overly short translations [13].
- **TER (Translation Edit Rate):** TER measures the number of edits (insertions, deletions, substitutions, and shifts) required to transform the model's translation into the reference translation. It quantifies how much post-editing is needed for the output [14].
- **ChrF (Character F-score):** ChrF is a character-level metric that measures the precision and recall of character n-grams. It is particularly useful for evaluating translations in morphologically rich languages where word-level matching might not capture nuances effectively [15].

5. Requirements and Environment Setup

5.1. Hardware and Software Requirements

5.1.1. Hardware Requirements

The hardware requirements for this project are designed to ensure optimal performance for training, testing, and real-time inference. The following hardware resources are essential:

- Processor (CPU)
- Graphics Processing Units (GPU)
- RAM - 16GB
- Storage - 500GB
- System - I5 Processor

5.1.2. Software Requirements

The software stack used for implementing the multilingual speech and text translation system is chosen for its efficiency in processing speech data, training deep learning models, and managing language-specific translations. Key software requirements include:

- Operating System
- Langauges and Libraries:

 doi:10.20944/preprints202503.0478.v1

  - Python
  - Streamlit
- Deep Learning Frameworks:
  - TensorFlow/PyTorch
  - Hugging Face Transformers
- Text-to-Speech (TTS) and Speech-to-Text (STT) Libraries:
  - Google Text-to-Speech (gTTS) API
  - SpeechRecognition

5.2. Development Environment and Tools

The development environment includes tools that facilitate the coding, testing, and deployment of the system. Tools used in this project include:

5.2.1. Integrated Development Environment (IDE)

- **VSCode** for code editing and debugging.
- **Jupyter Notebooks** for testing and experimenting with machine learning models in Python.

5.2.2. Version Control

- **Git/GitHub** for version control and collaborative development.

5.2.3. **Deployment:** The final application will be deployed as a web application, accessible via standard web browsers.

6. Results and Experiments

This section outlines the various modules implemented in the Multilingual Speech and Text Translation system, their functionalities, and the corresponding user interface (UI) features. The goal of this system is to provide seamless, real-time translation and transcription services for Indian regional languages through an intuitive user interface.

6.1. Project Modules

6.1.1. HomeScreen

A user's Home Screen shows the information about IndicTrans2 Translator Application with Speech recognition.

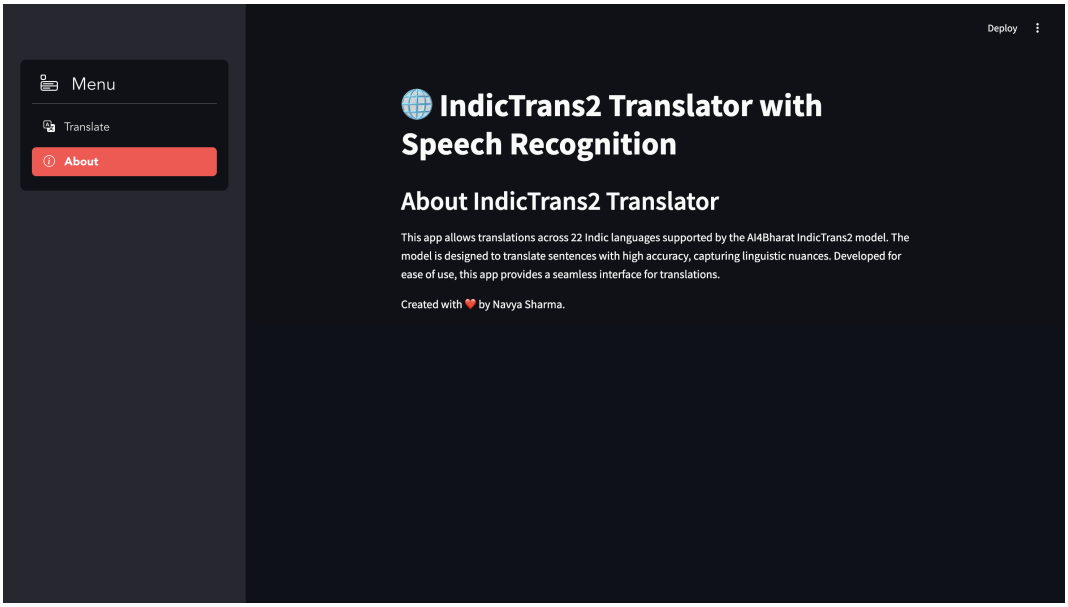

Figure 4. HomeScreen.

 doi:10.20944/preprints202503.0478.v1

6.1.2. Source and Target Language Selection

The user will select the Source and the target language from the languages shown in the dropdown.

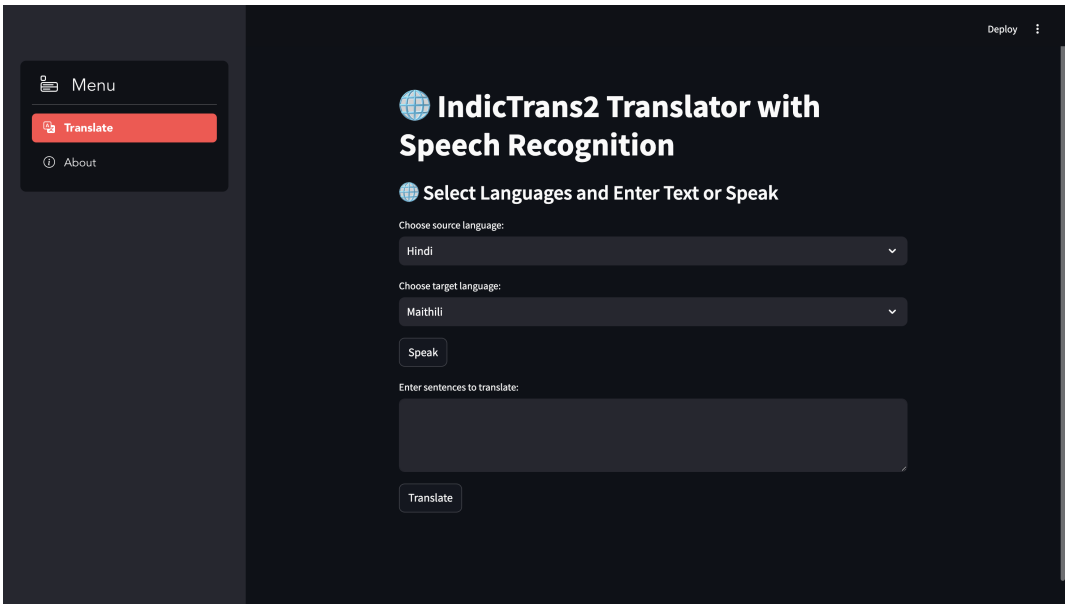

Figure 5. Source and Target Language Select.

6.1.3. Audio Input

The user will speak the sentence with is recognized by speech recognition module and converted into text format.

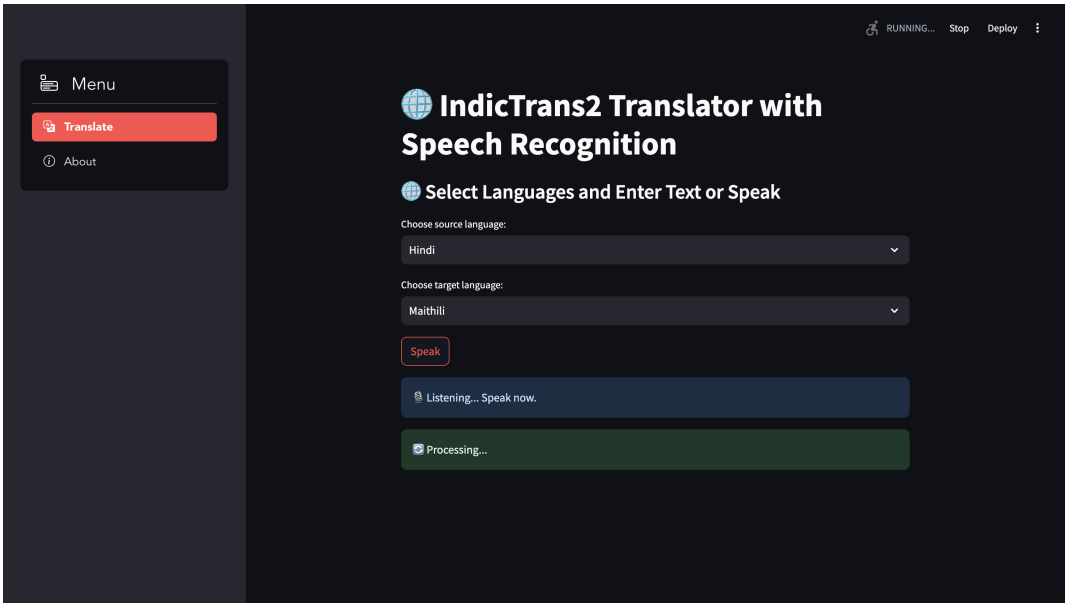

Figure 6. Audio Input.

6.1.4. Output Screen

The desired output will be displayed in text and audio format in Output Screen for the User.

 doi:10.20944/preprints202503.0478.v1

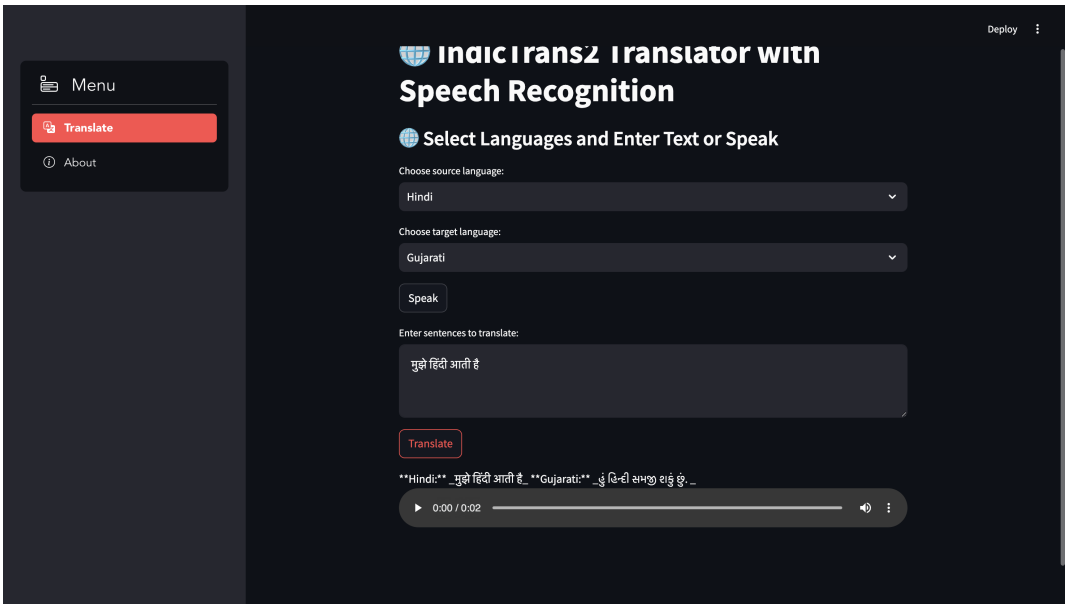

Figure 7. Output Screen.

Conclusion

In this research, we have developed a Multilingual Speech and Text Translation system designed to address the linguistic diversity of India. By integrating state-of-the-art models like IndicTrans2 with Speech-to-Text (STT) and Text-to-Speech (TTS) technologies, the system facilitates seamless translation and speech conversion for a wide range of Indian regional languages. This work aims to bridge language barriers by providing a comprehensive solution for real-time cross-lingual communication, ensuring that individuals can interact effortlessly despite speaking different languages.

The system enhances accessibility by supporting both speech and text inputs and outputs, which allows individuals with varying literacy levels or those with speech and hearing impairments to communicate effectively. Additionally, the real-time translation feature enables bilingual conversations, making it easier for users to engage in dialogue without language hindrances. The user interface has been designed to be intuitive and simple, catering to a wide audience, including those with limited technical expertise.

Furthermore, the system's adaptability and scalability make it suitable for expanding language support and catering to different applications across industries such as healthcare, education, and tourism. The results indicate the significant potential of AI-driven translation tools in improving communication and inclusivity in a multilingual society like India. By breaking down language barriers, the system can enhance interactions and help create more accessible environments in various sectors.

Overall, this work lays the foundation for future advancements in multilingual communication technologies. Moving forward, the focus will be on improving model accuracy, increasing language coverage, and exploring new applications, such as in e-commerce, governance, and emergency management. The Multilingual Speech and Text Translation system holds great promise in fostering better communication and understanding in multilingual regions [16].

References

1. Inaguma, H.; Duh, K.; Kawahara, T.; Watanabe, S. Multilingual End-to-End Speech Translation. In Proceedings of the 2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU), 2019, pp. 570–577. https://doi.org/10.1109/ASRU46091.2019.9003832.
2. Patil, P.S.; Dhande, M.S.; More, M.M.; Dalvi, M.S. Text to Speech and Language Conversion in Hindi and English Using CNN. *International Journal for Research in Applied Science and Engineering Technology* **2022**, *10*. https://doi.org/10.22214/ijraset.2022.40923.

 doi:10.20944/preprints202503.0478.v1

3. Al-Bakhrani, A.; Amran, G.A.; Al-Hejri, A.; Chavan, S.; Manza, R.; Nimbhore, S., Development of Multilingual Speech Recognition and Translation Technologies for Communication and Interaction; 2023; pp. 711–723. https://doi.org/10.2991/978-94-6463-196-8_54.
4. Wikipedia contributors. Languages of India, 2024. Accessed: 2024-11-09.
5. Mhaskar, S.; Bhat, V.; Batheja, A.; Deoghare, S.; Choudhary, P.; Bhattacharyya, P. VAKTA-SETU: A Speech-to-Speech Machine Translation Service in Select Indic Languages. *ArXiv* **2023**, *abs/2305.12518*. https://doi.org/10.48550/arXiv.2305.12518.
6. Sannigrahi, S.; Bawden, R. Investigating Lexical Sharing in Multilingual Machine Translation for Indian Languages **2023**. pp. 181–192. https://doi.org/10.48550/arXiv.2305.03207.
7. Padmane, P.; Pakhale, A.; Agrel, S.; Patel, A.; Pimparkar, S.; Bagde, P. Multilingual Speech and Text Recognition and Translation. *International Journal of Innovations in Engineering and Science* **2022**. https://doi.org/10.46335/ijies.2022.7.8.15.
8. Khanuja, S.; Bansal, D.; Mehtani, S.; Khosla, S.; Dey, A.; Gopalan, B.; Margam, D.; Aggarwal, P.; Nagipogu, R.; Dave, S.; et al. MuRIL: Multilingual Representations for Indian Languages. *ArXiv* **2021**, *abs/2103.10730*.
9. Rudrappa, N.T.; Reddy, M.V.; Hanumanthappa, M. KHiTE: Multilingual Speech Acquisition to Monolingual Text Translation. *Indian Journal Of Science And Technology* **2023**. https://doi.org/10.17485/ijst/v16i21.727.
10. Jayanthi, N.; Lakshmi, A.; Raju, C.S.K.; Swathi, B. Dual Translation of International and Indian Regional Language using Recent Machine Translation. *2020 3rd International Conference on Intelligent Sustainable Systems (ICISS)* **2020**, pp. 682–686. https://doi.org/10.1109/ICISS49785.2020.9316016.
11. Mujadia, V.; Sharma, D. Towards Speech to Speech Machine Translation focusing on Indian Languages **2023**. pp. 161–168. https://doi.org/10.18653/v1/2023.eacl-demo.19.
12. Gala, J.; Chitale, P.A.; Raghavan, A.K.; Gumma, V.; Doddapaneni, S.; M, A.K.; Nawale, J.A.; Sujatha, A.; Puduppully, R.; Raghavan, V.; et al. IndicTrans2: Towards High-Quality and Accessible Machine Translation Models for all 22 Scheduled Indian Languages. *Transactions on Machine Learning Research* **2023**.
13. Papineni, K.; Roukos, S.; Ward, T.; Zhu, W.J. Bleu: a Method for Automatic Evaluation of Machine Translation. In Proceedings of the Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics; Isabelle, P.; Charniak, E.; Lin, D., Eds., Philadelphia, Pennsylvania, USA, 2002; pp. 311–318. https://doi.org/10.3115/1073083.1073135.
14. Snover, M.; Dorr, B.; Schwartz, R.; Micciulla, L.; Makhoul, J. A Study of Translation Edit Rate with Targeted Human Annotation. In Proceedings of the Proceedings of the 7th Conference of the Association for Machine Translation in the Americas: Technical Papers, Cambridge, Massachusetts, USA, 8-12 2006; pp. 223–231.
15. Popović, M. chrF: character n-gram F-score for automatic MT evaluation. In Proceedings of the Proceedings of the Tenth Workshop on Statistical Machine Translation; Bojar, O.; Chatterjee, R.; Federmann, C.; Haddow, B.; Hokamp, C.; Huck, M.; Logacheva, V.; Pecina, P., Eds., Lisbon, Portugal, 2015; pp. 392–395. https://doi.org/10.18653/v1/W15-3049.
16. Proksch, S.O.; Wratil, C.; Wäckerle, J. Testing the Validity of Automatic Speech Recognition for Political Text Analysis. *Political Analysis* **2019**, *27*, 339–359. https://doi.org/10.1017/PAN.2018.62.