VAANI-SETU: AN AI-POWERED REAL-TIME MULTILINGUAL VOICE COMMUNICATION SYSTEM USING LARGE LANGUAGE MODELS

Hemavathi Patil[1], Megha[2], Sakshi[3], Shreya[4], Prarthana[5]

[1,2,3,4,5]*Department of Computer Science and Engineering (Data Science), Guru Nanak Dev Engineering College, Bidar – 585403, Karnataka, India*

hbpatilbidar@gmail.com , ramthirthmegha@gmail.com, sakshihippalgaonkar@gmail.com, shreyareddy306@gmail.com,prarthanabhushetty@gmail.com

Abstract

The exponential diversity of regional languages in India presents a significant challenge to seamless communication across the subcontinent. Despite digital advancements, the lack of robust, real-time, and context-aware speech translation hinders interactions in critical sectors like healthcare, education, and public administration.This research introduces Vaani-Setu, an AI-powered multilingual system designed to bridge these barriers. The platform integrates Automatic Speech Recognition (ASR), Large Language Models (LLMs) for semantic translation, and Text-to-Speech (TTS) for natural audio output. Specifically engineered for Indian languages like Hindi and Kannada, Vaani-Setu effectively handles regional accents, code-mixed speech (e.g., Hinglish), and natural conversational patterns.Unlike conventional tools that translate isolated phrases, Vaani-Setu utilizes a rolling dialogue buffer to preserve the speaker's intent across conversational turns. By leveraging pre-trained transformer models and optimized pipelines, the system enables two-way, low-latency interaction. This addresses a vital research gap, providing a context-aware, real-time communication platform tailored to the unique linguistic needs of the Indian population.

Keywords- Speech-to-Text, Large Language Models, Text-to-Speech, Multilingual Communication, Real-Time Translation, Indian Languages, Context-Aware NLP, Vaani-Setu.

1. Introduction

India's diversity of 1,600 dialects and 22 official languages create significant communication barriers in healthcare, law, and education, often leading to socio-economic exclusion. As digital adoption grows, the need for scalable, intelligent multilingual systems has outpaced traditional human interpretation. **Vaani-Setu** ("Voice Bridge") addresses this by using AI and Natural Language Processing to enable real-time spoken interaction. By integrating Speech-to-Text (STT), context-aware Large Language Models (LLMs), and Text-to-Speech (TTS), it provides a robust foundation for bridging regional linguistic gaps accurately and inclusively.

1.1 Technical Components Overview

The **STT module** converts audio into text, utilizing transformer-based architectures like Whisper or Indic ASR to handle regional accents and code-mixed speech (e.g., Hinglish). Unlike rule-based tools, the **LLM module** (using models like IndicTrans2 or GPT-4) ensures semantic coherence and understands idiomatic expressions across conversational turns. Finally, the **TTS system** employs neural models like FastSpeech 2, fine-tuned on Indian datasets, to synthesize natural, intelligible audio in languages such as Hindi and Kannada.

1.2 Importance of the Study

This research addresses high-impact communication gaps in critical domains; for instance, facilitating life-saving dialogue between a Kannada-speaking patient and a Hindi-speaking doctor. While global AI research often neglects low-resource Indian languages in favor of English or French, Vaani-Setu

contributes specifically to the Indian AI ecosystem.

2. Related Work

The development of Vaani-Setu is informed by a comprehensive review of principal research contributions across six thematic streams: Speech-to-Speech Translation, Automatic Speech Recognition, Neural Machine Translation, Large Language Models for Translation, Indian Language Processing, and Real-Time Communication Systems.

2.1 Speech-to-Speech Translation Systems

In the domain of direct speech-to-speech translation, Jia et al. (2019) proposed Translatotron, the first model to bypass intermediate text representation using a sequence-to-sequence architecture. Although it established a benchmark for end-to-end approaches, it showed lower accuracy compared to cascaded systems and was not evaluated on Indian languages, motivating the cascaded (STT → LLM → TTS) architecture used in Vaani-Setu. Meta AI recently introduced SeamlessM4T, a unified model supporting speech and text translation across 100 languages, including several Indian languages. However, performance on low-resource Indian languages such as Kannada remains suboptimal compared to dedicated Indian-language models, and real-time deployment on resource-constrained devices remains challenging. Additional toolkits like ESP net-ST facilitate rapid prototyping but lack built-in support for Indian languages and do not address real-time conversational use cases. Ma et al. (2020) explored simultaneous speech translation using Monotonic Multi head Attention (MMA) to achieve low latency, a methodology that informs the streaming inference strategy adopted in Vaani-Setu's STT module.

2.2 Automatic Speech Recognition (ASR)

OpenAI's Whisper is a transformer-based ASR model trained on 680,000 hours of multilingual audio, achieving near-human transcription accuracy across 99 languages including Hindi. While its large model size poses challenges for real-time mobile deployment, the medium variant offers an effective balance of accuracy and inference speed for Vaani-Setu. The Indic NLP Suite provides monolingual corpora and evaluation benchmarks for 11 Indian languages, directly supporting the evaluation of Vaani-Setu's components. Furthermore, Wav2Vec 2.0 introduced a self-supervised framework for speech representation learning, which has been successfully fine-tuned for low-resource languages like Kannada. The Conformer architecture, combining convolutional neural networks with transformer attention, has set new benchmarks on Libri Speech and has been adapted for Indian language recognition, demonstrating strong performance on continuous speech.

2.3 Neural Machine Translation and LLMs

The foundational work by Vaswani et al. (2017) introduced the Transformer architecture and its self-attention mechanism, which is essential for capturing long-range linguistic dependencies in modern NMT systems. Multilingual models like

mBART have been pre-trained on 25 languages including Hindi using denoising objectives, though their model size can create latency challenges. IndicTrans2, an open-source NMT model designed for all 22 scheduled Indian languages, significantly outperforms general-purpose systems on Indian language pairs and serves as the primary translation engine in Vaani-Setu. Tiedemann and Thottingal (2020) provided Opus-MT, which offers computationally efficient models that nonetheless lack the conversational context awareness required for high-fidelity dialogue.

The integration of Large Language Models has further refined translation capabilities. GPT-4 demonstrates strong multilingual translation performance and conversational context maintenance, making it an excellent candidate for context-enriched translation. LLaMA 2, an open-source LLM family, can be fine-tuned for multilingual translation with lower computational overhead, suitable for privacy-preserving on-premise deployment. Lightweight models like Gemma from Google DeepMind enable edge device deployment, supporting real-time requirements for mobile

versions of the system. Research by Shi et al. (2023) demonstrated that LLMs could perform multilingual reasoning through chain-of-thought prompting, informing the prompting strategy used in Vaani-Setu's dialogue management.

2.4 Indian Language Processing and Latency

The AI4Bharat-IndicNLP corpus provides foundational training data for models like Indic Trans. MuRIL, a BERT-based model pre-trained on 17 Indian languages, serves as an initialization for fine-tuning language understanding components. Samanantar, the largest publicly available parallel corpus for Indian languages, directly enables high-quality Hindi-Kannada translation. For speech synthesis, the Indic TTS collection from IIT Madras forms the core training corpus used to fine-tune Fast Speech 2 and VITS models.

Real-time communication requires minimizing latency. Liu et al. (2020) proposed partial hypothesis selection to achieve a 30% reduction in average latency. Streaming ASR using CTC and RNN-Transducer architectures developed at Google enables real-time mobile transcription. Fast Speech 2 generates mel-spectrograms 3 to 5 times faster than autoregressive models while maintaining high audio quality. VITS provides a single-stage end-to-end alternative with reduced inference latency. Huber et al. (2023) presented a framework for low-latency simultaneous speech translation, informing the evaluation strategy for Vaani-Setu's end-to-end pipeline.

2.5 Summary Comparison of Core Models

The following table provides a comparison of the primary models considered and integrated into the Vaani-Setu architecture for Indian language processing.

3. Materials and Methods

The materials and methods section explains the overall framework and methodology used in this study. The Vaani-Setu system follows a modular, pipeline-based architecture with five core processing stages organized as a streaming dataflow, supporting bidirectional communication between Hindi and Kannada speakers.

3.1 System Architecture

The architecture is designed to process audio from both User A (Hindi speaker) and User B (Kannada speaker) simultaneously through parallel pipeline instances. The streaming nature of the dataflow ensures that processing stages are overlapped to minimize end-to-end latency.

3.2 Module-wise Description

The system is partitioned into seven distinct modules, each responsible for a specific stage of the communication pipeline.

3.2.1 Audio Input Module

The audio input module captures real-time speech from the user's microphone at a sampling rate of 16,000 Hz in mono format. Voice Activity Detection (VAD) is applied to identify speech onset and offset, eliminating silent segments and reducing unnecessary processing load. A noise reduction pre-filter improves the signal-to-noise ratio in noisy environments such as public spaces, ensuring robust downstream ASR performance.

3.2.2 Speech-to-Text (STT) Module

The STT module transcribes the incoming audio stream into UTF-8 encoded text in the source language. The primary model used is OpenAI Whisper (medium variant) for Hindi, which supports transcription with strong accent tolerance. For Kannada input, AI4Bharat's Indic ASR or a Wav2Vec 2.0 model fine-tuned on Kannada data is employed. Streaming transcription using CTC-based decoding processes audio incrementally, producing partial transcriptions that are progressively refined, achieving low-latency output.

3.2.3 Context Manager

The context manager maintains a rolling dialogue buffer storing the N most recent conversational turns, including both utterances and their translations. This context window is dynamically managed; older turns are evicted as the conversation progresses to keep the total

context within the LLM's input token limit. The context is serialized and prepended to each new translation request, enabling the LLM to resolve pronouns, maintain topic continuity, and understand references to earlier parts of the conversation.

3.2.4 LLM Context-Aware Translation Module

The translation module receives the transcribed text and the dialogue context to perform semantically faithful translation into the target language. IndicTrans2 serves as the primary translation engine for Hindi-Kannada and Kannada-Hindi translation, selected for its state-of-the-art performance on Indian language pairs. For scenarios requiring deeper contextual understanding, the module optionally invokes LLaMA 2 or GPT-4 through a structured prompt. Code-mixed inputs (e.g., Hinglish) are normalized by the context manager before being passed to the translation engine.

3.2.5 Text-to-Speech (TTS) Module

The TTS module converts translated text into natural-sounding speech audio. Fast Speech 2, fine-tuned on the Indic TTS Kannada and Hindi corpora, serves as the primary synthesis model, generating mel-spectrograms that are vocoded into audio waveforms using HiFi-GAN. The output audio is produced at 22,050 Hz. Alternatively, VITS provides single-stage synthesis with comparable quality at slightly higher computational cost.

3.2.6 Audio Output Module

The audio output module plays the synthesized speech through the device's speaker. In the web-based interface, audio is delivered via the Web Audio API, buffering and playing synthesized chunks as they arrive to minimize perceptible latency. The total end-to-end latency target—from speech onset to audio playback—is under 2 seconds for typical conversational utterances of 5 to 10 words.

4. Results and Discussion

This section presents the experimental results obtained using the proposed method. Performance metrics are analyzed and compared with existing approaches to interpret the results and highlight the advantages of the Vaani-Setu solution.

4.1 Evaluation Metrics

The system's performance is evaluated using a multi-dimensional framework comprising objective and subjective metrics.

4.1.1 Word Error Rate (WER)

WER is the primary metric for evaluating the STT module. It is calculated by comparing the transcribed text against a ground-truth reference, considering substitutions (***S***), deletions (***D***), and insertions

(***I***):

$$WER = \frac{S + D + I}{N}$$

where N is the total number of words in the reference.

4.1.2 BLEU Score

The Bilingual Evaluation Understudy (BLEU) score measures the quality of text that has been machine-translated from one natural language to another. It computes the precision of n-grams of the machine translation against the reference translation.

4.1.3 Mean Opinion Score (MOS)

MOS is a subjective measure of the synthesized speech quality in the TTS module. Human evaluators rate the naturalness and intelligibility of the audio on a scale from 1 (poor) to 5 (excellent).

4.2 Comparative Performance Analysis

The proposed cascaded architecture was evaluated against general-purpose translation tools. The results indicate that while general-purpose models perform well on high-resource language pairs, Vaani-Setu's integration of IndicTrans2 and a context manager provides superior results for Indian languages.

The discussion interprets these results as evidence of the effectiveness of context-aware LLMs in maintaining semantic intent across conversational turns. The reduction in WER for code-mixed speech is particularly significant for the Indian demographic, where "Hinglish" is pervasive. By normalizing these inputs before translation, Vaani-Setu avoids the catastrophic failures typical of monolingual statistical models.

5. Conclusion

The conclusion summarizes the key findings and contributions of the study. This research demonstrates the effectiveness of the Vaani-Setu system as a real-time, context-aware communication bridge for Indian languages. By integrating STT, LLMs, and TTS into a unified streaming pipeline, the system successfully addresses critical communication gaps in healthcare, education, and public administration. The system's ability to handle regional accents, code-mixed speech, and conversational context distinguishes it from conventional translation tools.

Future research directions will include extending the language scope to all 22 scheduled Indian languages, investigating further latency optimization through model quantization and hardware acceleration, and exploring federated deployment strategies to enhance privacy in sensitive domains. The project holds significant promise for enabling linguistic inclusivity across India's diverse population and contributing meaningfully to the vision of a digitally accessible nation.

References

[1] Y. Jia et al., “Direct speech-to-speech translation with a sequence-to-sequence model,” Interspeech, 2019.
[2] L. Barrault et al., “SeamlessM4T: Massively multilingual & multimodal machine translation,” arXiv, 2023.
[3] H. Inaguma et al., “ESPnet ST: All-in-One Speech Translation Toolkit,” ACL, 2020.
[4] X. Ma et al., “Monotonic Multihead Attention,” ICLR, 2020.
[5] A. Radford et al., “Robust speech recognition via large-scale weak supervision,” ICML, 2023.
[6] D. Kakwani et al., “IndicNLPSuite for Indian languages,” EMNLP Findings, 2020.
[7] A. Baevski et al., “Wav2Vec 2.0 for speech representation learning,” NeurIPS, 2020.
[8] A. Gulati et al., “Conformer: Transformer for speech recognition,” Interspeech, 2020.
[9] A. Vaswani et al., “Attention is all you need,” NeurIPS, 2017.
[10] Y. Liu et al., “Multilingual denoising pre-training for translation,” TACL, 2020.
[11] J. Gala et al., “IndicTrans2 for Indian language translation,” arXiv, 2023.
[12] J. Tiedemann and S. Thottingal, “OPUS-MT translation models,” EAMT, 2020.
[13] OpenAI, “GPT-4 technical report,” arXiv, 2023.
[14] H. Touvron et al., “LLaMA 2 foundation models,” arXiv, 2023.
[15] Gemma Team, “Gemma models by Google DeepMind,” arXiv, 2024.
[16] F. Shi et al., “Language models as multilingual reasoners,” ICLR, 2023.
[17] A. Kunchukuttan et al., “IndicNLP corpus for Indian languages,” arXiv, 2020.
[18] S. Khanuja et al., “MuRIL for Indian language representation,” arXiv, 2021.
[19] G. Ramesh et al., “Samanantar parallel corpus for Indic languages,” TACL, 2022.
[20] A. Baby et al., “Resources for Indian languages,” Springer, 2016.
[21] D. Liu et al., “Low-latency speech translation,” Interspeech, 2020.
[22] Y. He et al., “Streaming speech recognition for mobile devices,” ICASSP, 2019.
[23] Y. Ren et al., “FastSpeech 2 for text-to-speech,” ICLR, 2021.
[24] J. Kim et al., “VITS for end-to-end text-to-speech,” ICML, 2021.
[25] C. Huber et al., “Low-latency speech translation evaluation,” EMNLP, 2023.