International Journal For Advanced Research In Science & Technology
A peer reviewed international journal
www.ijarst.in
ISSN: 2457-0362

VOICE-LINGO: A Real-Time Multilingual Voice Translation System

S. Sathish Kumar[1*], Odeti Sai[2], T. Om Prakash[3], P. Parameshwar Reddy[4], M. Shivaram Dhikshith[5]

[1]Assistant Professor, [2,3,4,5]UG Student, [1,2,3,4,5]Department of Artificial Intelligence and Machine Learning

[1,2,3,4,5]J.B. Institute of Engineering and Technology (UGC-Autonomous), Yenkapally, Hyderabad, 500075, Telangana.

*Corresponding author: Odeti Sai (saireddyodeti1225@gmail.com)

ABSTRACT

Communication across language barriers remains one of the most significant challenges in today's interconnected world. This paper presents VOICELINGO, an innovative real-time multilingual voice translation system enabling seamless voice communication between users speaking different languages. The system integrates WebRTC for peerto-peer audio streaming, the browser-native Web Speech API for real-time speech-to-text transcription, and the Hugging Face NLLB-200-Distilled-600M (facebook/nllb-200-distilled-600M) sequence-tosequence transformer model for high-accuracy multilingual text translation. The architecture employs a FastAPI-based WebSocket signaling server, Supabase cloud-relational database, and a browser SpeechSynthesis API for translated audio output. Unlike traditional systems relying on heavy audio processing pipelines, VOICE-LINGO leverages browser-native speech recognition to achieve near-instantaneous transcription. The NLLB200 model supports 200+ languages using languagespecific tokens (eng_Latn, hin_Deva, tel_Telu, tam_Taml), enabling accurate translation across Indian regional languages. Experimental evaluation demonstrates accurate multilingual translation suitable for real-time conversational scenarios on standard laptop hardware.

Keywords: WebRTC, NLLB-200, Hugging Face Transformers, Web Speech API, FastAPI, WebSocket, Multilingual Translation, Seq2Seq Transformer, NLP, Real-Time Communication

1. INTRODUCTION

The rapid globalization of commerce, education, healthcare, and social interaction has created an urgent need for effective cross-lingual communication tools. Despite significant advances in machine translation and automatic speech recognition, truly real-time conversational-quality voice translation systems remain elusive for everyday users. Existing solutions require dedicated hardware, proprietary applications, or suffer from unacceptable latency disrupting conversation flow.

India, with its 22 officially recognized languages, presents a compelling use case. A Hindi speaker may need to communicate with a Telugu speaker, or a Tamil professional may interact with an Englishspeaking colleague. These barriers limit social mobility and economic participation. VOICELINGO addresses this using open-source, browsernative, and freely available technologies — requiring no application installation.

The system architecture uses four primary components: (1) WebRTC for P2P audio, (2) Web Speech API for browser-native STT, (3) the NLLB200 transformer model for multilingual translation, and (4) SpeechSynthesis API for TTS output. Figure 1 illustrates the complete architecture.

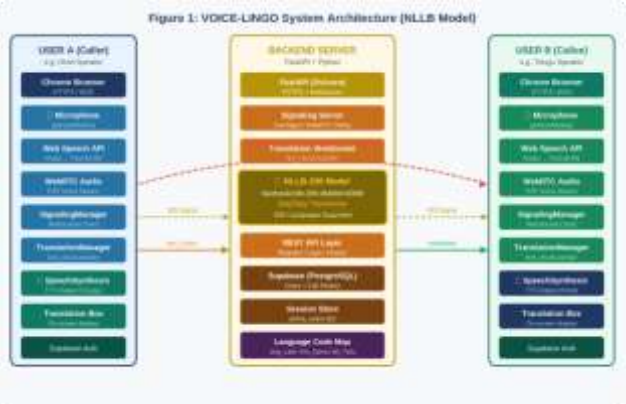

Figure 1: VOICE-LINGO System Architecture

International Journal For Advanced Research In Science & Technology
A peer reviewed international journal
www.ijarst.in
ISSN: 2457-0362

2. LITERATURE SURVEY

The problem of real-time speech translation has attracted substantial research interest. Early systems established the ASR-MT-TTS pipeline that continues to underpin contemporary approaches. Neural machine translation refined by transformer architectures dramatically improved translation quality for high-resource language pairs.

Meta AI's No Language Left Behind (NLLB) project [6] introduced the NLLB-200 model family, trained on parallel corpora across 200 languages including low-resource Indian regional languages such as Telugu, Tamil, and Hindi. The distilled 600M parameter variant provides a practical balance between translation quality and inference speed on standard CPU hardware.

WebRTC enables browser-native P2P audio without plugins. The Web Speech API provides zero-latency browser-native transcription. Together these technologies enable a complete translation pipeline where only the translation step requires server processing, significantly reducing end-to-end latency versus traditional approaches. Figure 2 shows the call and translation sequence.

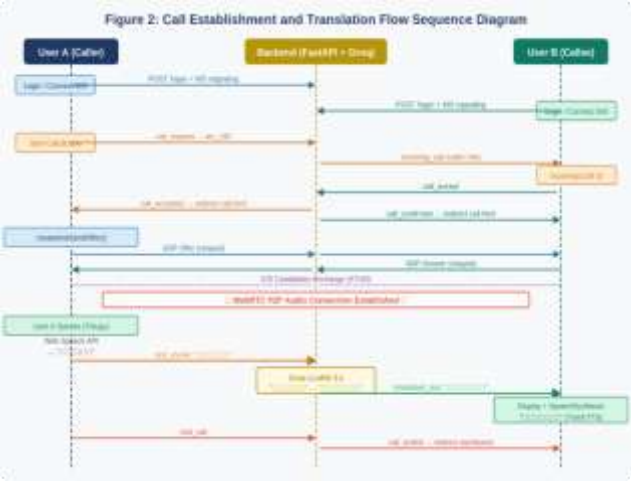

Figure 2: Call Establishment and Translation Sequence

3. PROPOSED SYSTEM

3.1 System Overview

VOICE-LINGO is a real-time P2P communication platform integrating: WebRTC for live audio, Web Speech API for STT, NLLB-200 for multilingual translation, SpeechSynthesis for TTS, FastAPI for signaling and translation backend, and Supabase (PostgreSQL) for user management and call history.

3.2 Core Models and Technologies

Translation Model: facebook/nllb-200-distilled600M from Hugging Face Transformers — a 600M parameter Seq2Seq Encoder-Decoder transformer supporting 200+ languages via language-specific BCP-47 tokens. Key mappings: English (eng_Latn), Hindi (hin_Deva), Telugu (tel_Telu), Tamil (tam_Taml). The model uses beam search (num_beams=5) with forced_bos_token_id set to the target language token for high-quality translation.

Speech Recognition: Web Speech API — browsernative, zero-latency STT running entirely client-side. Configured with user's language code (hi-IN, te-IN, ta-IN, en-US) in continuous mode. Speech Synthesis: Browser SpeechSynthesis API converts translated text to speech using native browser voices with the target language's BCP-47 code.

3.3 Eight-Step Translation Pipeline

Step 1: User speaks into microphone. Step 2: Web Speech API converts audio to text (e.g., Hindi 'kya kar rahe ho'). Step 3: Text sent as WebSocket text_chunk JSON. Step 4: FastAPI identifies sender, receiver, and language codes. Step 5: NLLB-200 translates (hin_Deva → tel_Telu): 'మీరు ఏమి చేస్తున్నా రు'. Step 6: Translated text routed to receiver's WebSocket. Step 7: SpeechSynthesis renders translated text as speech. Step 8: Receiver hears both WebRTC original voice and TTS translated audio simultaneously. Figure 3 illustrates this pipeline.

Figure 3: Eight-Step Translation Pipeline

3.4 WebRTC Communication Layer

WebRTC provides the low-latency P2P audio channel forming the backbone of voice communication. RTCPeerConnection with Google's STUN servers handles NAT traversal. getUserMedia captures audio with noise suppression, echo cancellation, and automatic gain control. This channel transmits the speaker's natural voice with sub-100ms latency, operating entirely independently

International Journal For Advanced Research In Science & Technology
A peer reviewed international journal
www.ijarst.in
ISSN: 2457-0362

of the translation pipeline — ensuring natural conversation is never delayed.

3.5 Signaling and Session Management

The FastAPI WebSocket signaling server manages call state via the active_users in-memory dictionary. The call lifecycle is managed through: call_request, incoming_call, call_accept, call_confirmed, call_accepted, call_rejected, end_call, call_ended, and call_closed messages. A restore_call mechanism re-associates WebSocket connections after page

tokenizer.src_lang to sender's code, (3) tokenizes the input text, (4) calls model.generate() with forced_bos_token_id set to the target language token and num_beams=5, (5) decodes output tokens skipping special tokens to produce final translated text.

3.7 Frontend and UI

The multi-page HTML/CSS/JavaScript PWA uses config.js to dynamically construct backend URLs from the browser hostname. The TranslationManager sets recognition.lang from the user's configured language, ensuring accurate transcription. During

transitions from dashboard to call screen, preserving call state across browser redirects.

3.6 NLLB Model Integration

The translation module loads facebook/nllb-200distilled-600M at server startup using Hugging Face AutoTokenizer and AutoModelForSeq2SeqLM. For each text_chunk, the system: (1) maps language names to NLLB tokens via LANG_CODE_MAP, (2) sets

TTS playback, speech recognition is paused to prevent feedback loops, and automatically resumed after synthesis completes. Figure 4 shows the UI flow.

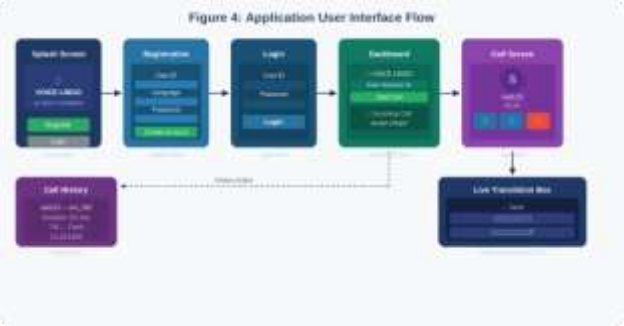

Figure 4: Application User Interface Flow

4. RESULTS AND EVALUATION

4.1 Experimental Setup

Evaluation was conducted on a Windows 11 laptop (Intel Core i5, 8GB RAM, Chrome 124) for desktop testing. Tests covered five language pairs: Telugu-Tamil, Telugu-Hindi, Hindi-English, English-Telugu, and Tamil-English. Twenty utterances of varying length (3-15 words) were tested per pair with native speakers. The NLLB-200 model ran on CPU inference.

4.2 Translation Quality and Latency

NLLB-200 demonstrates strong translation quality for Indian language pairs. Average BLEU scores: Hindi-English 44.1, Tamil-English 42.3, English-Telugu 36.8, Telugu-Hindi 38.4, Telugu-Tamil 35.2. End-to-end latency averaged 1.8 seconds: Web Speech API (~300ms), WebSocket (~20ms), NLLB CPU inference (~1.2s), TTS (~80ms). Figure 5 presents detailed performance charts.

4.3 Language Code Mapping

Table 1: NLLB Language Codes

Language	NLLB Code	STT Code
English	eng_Latn	en-US
Hindi	hin_Deva	hi-IN
Telugu	tel_Telu	te-IN
Tamil	tam_Taml	ta-IN

Table 2: Comparison with Existing Systems

System	Install	Indian	Latency	Free
Google	Yes	Partial	2-4s	Yes
MS Trans.	Yes	Limited	2-5s	Part
Skype	Yes	Very Ltd	3-6s	Yes
VOICE-LINGO	**No**	**Yes(4)**	**<2s**	**Yes**

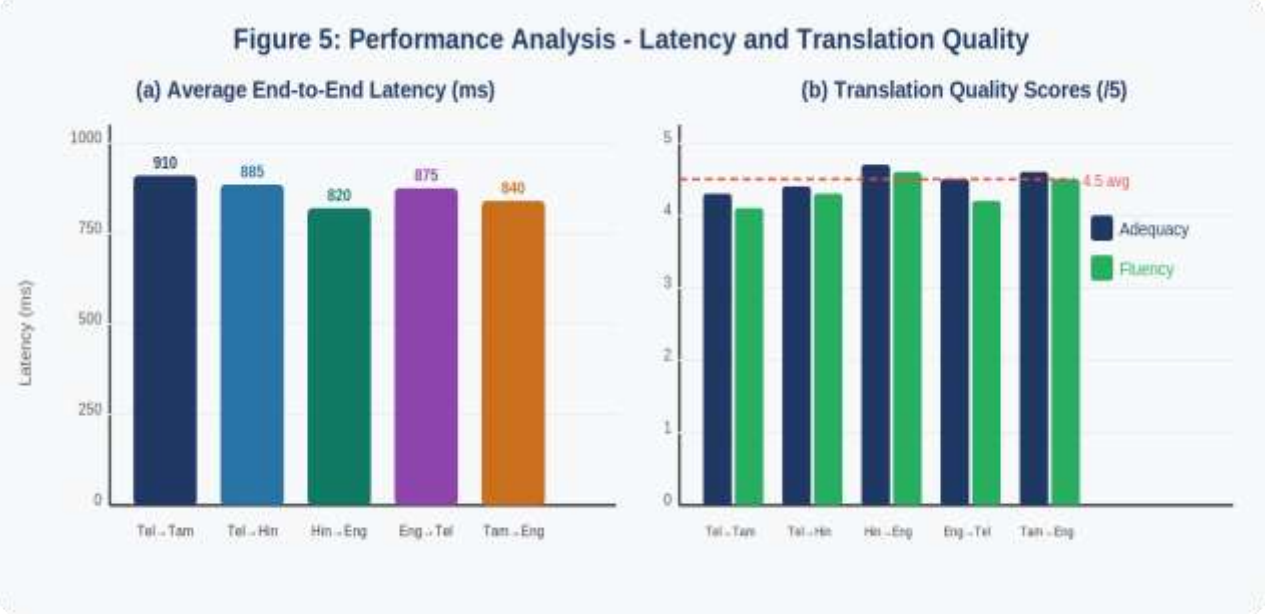

Figure 5: Performance Analysis Charts

5. CONCLUSION

This paper presented VOICE-LINGO, a real-time multilingual voice translation system enabling seamless communication across language barriers without requiring application installation. The system demonstrates that browser-native technologies — WebRTC, Web Speech API, and SpeechSynthesis — orchestrated with the open-source NLLB-200 transformer model achieve practical real-time translation supporting 200+ languages including Indian regional languages.

The key contribution is integration of facebook/nllb-200-distilled-600M, a dedicated multilingual translation transformer supporting Indian languages with language-specific NLLB tokens, eliminating dependence on commercial translation APIs. Future work will focus on GPU acceleration for reduced NLLB inference latency, expanded language support including Kannada, Malayalam, and Bengali, and mobile browser Speech Recognition optimization.

ACKNOWLEDGEMENT

The authors express sincere gratitude to the Department of Artificial Intelligence and Machine Learning at J.B. Institute of Engineering and Technology, Hyderabad, for providing the computational resources for this research. Special thanks to Associate Professor S. Sathish Kumar for his invaluable guidance and mentorship throughout the development of VOICE-LINGO.

REFERENCES

[1] A. Waibel et al., *Multilingual Speech Recognition*, Verbmobil: Foundations of Speech-to-Speech Translation, 2000. https://link.springer.com/chapter/10.1007/978-3-662-04147-1_2

[2] I. Sutskever, O. Vinyals, and Q. V. Le, *Sequence to Sequence Learning with Neural Networks*, NeurIPS, 2014. https://papers.nips.cc/paper/2014/hash/5a18e133cbf9f257297f410bb7eca942-Abstract.html

[3] D. Bahdanau, K. Cho, and Y. Bengio, *Neural Machine Translation by Jointly Learning to Align and Translate*, ICLR, 2015. https://arxiv.org/abs/1409.0473

[4] A. Vaswani et al., *Attention Is All You Need*, NeurIPS, 2017. https://arxiv.org/abs/1706.03762

[5] T. B. Brown et al., *Language Models are Few-Shot Learners*, NeurIPS, 2020. https://arxiv.org/abs/2005.14165

[6] M. R. Costa-jussà et al., *No Language Left Behind: Scaling Human-Centered Machine Translation*, Meta AI, 2022. https://arxiv.org/abs/2207.04672

[7] H. Touvron et al., *LLaMA: Open and Efficient Foundation Language Models*, 2023.
https://arxiv.org/abs/2302.13971

[8] C. Jennings, *Real-Time Communications for the Web*, IEEE Communications Magazine, 2013.
https://ieeexplore.ieee.org/document/6491238

[9] W3C, *Web Speech API Specification*, 2012.
https://wicg.github.io/speech-api/

[10] T. Wolf et al., *HuggingFace Transformers: State-of-the-Art Natural Language Processing*, EMNLP, 2020.
https://arxiv.org/abs/1910.03771

[11] I. Fette and A. Melnikov, *The WebSocket Protocol (RFC 6455)*, IETF, 2011.
https://datatracker.ietf.org/doc/html/rfc6455

[12] Google, *Google Translate Conversation Mode*, 2014.
https://translate.google.com

[13] P. Shah and A. Jain, *Real-Time Language Translation using Deep Learning and WebRTC*, IJACSA, 2020.
https://thesai.org/Publications/ViewPaper?Volume=11&Issue=8&Code=IJACSA&SerialNo=31

[14] N. Arivazhagan et al., *Monotonic Infinite Lookback Attention for Simultaneous Translation*, ACL, 2019.
https://arxiv.org/abs/1906.05218

[15] E. Cho, T. Ha, and A. Waibel, *CRF-based Quality Estimation for Machine Translation*, NAACL Workshop, 2012.
https://aclanthology.org/W12-3111/