Advanced React Web-Based Framework for Real-Time Transcription and Multilingual Translation

Mrs.J.Juslin Sega, Assistant professor
Department of Computer Science and Engineering
SRM Institute of Science and technology,
Ramapuram
Chennai, India
juslinsj@srmist.edu.in

G Benito Rohan, UG Student
Department of Computer Science and Engineering
SRM Institute of Science and technology,
Ramapuram
Chennai, India
benitorohan@gmail.com

I Hemanth Kumar, UG Student
Department of Computer Science and Engineering
SRM Institute of Science and technology,
Ramapuram
Chennai, India
hemanth20034@gmail.com

R Tarun Karthik, UG Student
Department of Computer Science and Engineering
SRM Institute of Science and technology,
Ramapuram
Chennai, India
tarunkarthik28007@gmail.com

Abstract: This paper presents a web application developed with React, focused on real-time transcription and translation through the usage of machine learning. By leveraging the latest advancements in machine learning (ML) as well as in natural language processing (NLP), the application seeks to give the user a user-friendly experience for converting spoken language into text and translating it into multiple languages. Notable features include instantaneous speech-to-text conversion, smooth integration with widely used translation APIs, and an adaptable interface suitable for diverse scenarios such as meetings, lectures, and multilingual communications. The paper details the technical framework, including the React-based front end, back-end services, and NLP model integration. Additionally, performance metrics, user feedback, and case studies are provided to showcase the application's effectiveness and potential to enhance cross-linguistic communication. The conclusion outlines future enhancements to the application's capabilities and the expansion of its language support

Keywords Neural Machine Translation (NMT), Automatic Speech Recognition (ASR), Natural Language Processing (NLP),

1. Introduction

In a world that is becoming more globalized, the capacity to communicate effectively across language has become increasingly essential. With the proliferation of international collaborations, multicultural workplaces and global information exchange, the demand for robust tools that facilitate real-time transcription and translation has grown substantially. This paper introduces a cutting-edge web application developed using the React framework, designed to meet this demand by providing users with an intuitive, efficient, and scalable solution for transcribing spoken language and translating text across multiple languages in real time.

Real-time transcription and translation are crucial across multiple areas, such as business, education, healthcare, and social interactions. Their significance is evident in facilitating communication and improving accessibility in these fields. For instance, multinational corporations often face challenges in managing communications between employees who speak different languages. Similarly, educational institutions with diverse student populations need tools that can bridge language gaps in classrooms and online learning environments. In healthcare, accurate and timely communication between medical professionals and for patients who speak different languages, effective communication can be crucial and may even determine life or death. Furthermore, in social contexts, breaking down language barriers enhances cultural exchange and mutual understanding.

Our application utilizes the strengths of ML and NLP to provide exceptional accuracy and performance. NLP, which is a branch of artificial intelligence, concentrates on how computers and humans interact using natural language. By implementing advanced NLP models, the application enhances its capabilities in understanding and processing language, the application can effectively convert spoken language into text (speech-to-text) and subsequently translate this text into the desired language. The integration of these models with the React framework ensures that the application is both responsive and user-friendly, providing a seamless experience for users.

The decision to use React for front-end development stems from its strength, versatility, and capacity. The component-based usage present in React facilitates efficient application development and maintenance, allowing developers to create reusable components that are easy to manage and update. This modular approach not only improves the application's scalability but also guarantees consistency and reliability throughout various sections of the user interface.

On the back end, the application incorporates scalable services that handle data processing, model integration, and API interactions. These services are designed to support the heavy computational load required for real-time transcription and translation, ensuring that the application remains performant even under high usage conditions. The back-end architecture is optimized for rapid data processing and minimal latency, which is crucial for real-time applications.

In addition to the technical aspects, the application places a strong emphasis on user experience. The interface is crafted to be user-friendly, enabling individuals to navigate and use the transcription and translation features effortlessly, without requiring extensive training. Considerable focus is placed on accessibility, ensuring that the application is functional for users with different levels of technical expertise and for those with disabilities.

To validate the effectiveness of the application, we conducted extensive performance evaluations and gathered user feedback through various case studies. These evaluations focused on metrics such as transcription accuracy, translation quality, response time, and overall user satisfaction. The results demonstrate the application's capability to enhance communication in multilingual settings and its potential to be a valuable tool in diverse scenarios.

In the upcoming sections, we will explore the technical architecture of the application in greater depth, the NLP and ML models utilized, and the integration of translation APIs. Additionally, we will examine performance metrics, user feedback, and detailed case studies to provide a thorough overview of the application’s development and its impact. Finally, we will discuss potential future directions for the application, such as expanding language support, integrating more NLP models, and enhancing the overall user experience.

By addressing the critical need for real-time transcription and translation, this React-based web application represents a significant step forward in facilitating effective communication across language barriers. The insights and findings discussed in this paper add to the wider field of language technology and highlight the potential of web-based solutions to transform global communication dynamics.

2. Related work

Traditional speech-to-speech translation (S2ST) systems are typically constructed by integrating end-to-end speech-to-text (S2T) models along with text-to-speech (TTS) technologies. Most research in translation of vocal speech has concentrated upon speech-to-text framework. Investigations into ASR+MT systems aim to find more effective ways to incorporate and leverage lattice of speech outputs into machine translation models to mitigate the issue of spread of error among processes. End-to-end S2T systems show promise in addressing this issue, particularly when trained effectively using multitask learning, model pre-training, or data augmentation techniques to tackle data scarcity challenges. Research on TTS within the context of S2ST emphasizes the synthesis of paralinguistic information derived from the source speech.

Speech-to-speech (S2ST) translation systems play a vital role in facilitating communication across different languages in real time. These innovative systems work by taking spoken input in one language, translating it, and then generating spoken output in another language. This process generally involves three essential components: speech recognition to capture the spoken words, machine translation to interpret the content, and text-to-speech synthesis to vocalize the translation.

There are primarily two approaches to designing S2ST systems: a modular (or cascaded) approach and a more integrated end-to-end approach. In the modular setup, each function operates separately. Speech recognition processes the audio into text, which is then translated, and finally, that text is transformed back into speech. While this structure allows for individual components to be upgraded or improved independently, it can also lead to challenges. One significant issue is the risk of "error propagation," where mistakes made during transcription can compromise the quality of the subsequent translation. Recent efforts in this area focus on minimizing these errors by enhancing how transcription outputs are utilized in the translation phase. By implementing methods such as providing multiple potential transcriptions or confidence scores, researchers aim to enable translation modules to better manage uncertainties, ultimately leading to more accurate results.

Conversely, end-to-end S2ST systems strive to consolidate the entire translation process into a single framework. This integrated method helps reduce the chances of errors accumulating across different stages since the model learns to handle the entire conversion from audio input to speech output directly. However, developing these systems often requires large, diverse datasets, which can be a challenge, especially for less commonly spoken languages. To tackle this, researchers are exploring innovative training strategies that can effectively utilize limited data, helping to expand the capabilities of end-to-end models.

As advancements in this field continue, S2ST technology is becoming increasingly sophisticated, producing translations that sound more natural and are easier to understand. These developments are paving the way for more seamless multilingual interactions, making it possible for individuals from different linguistic backgrounds to communicate effortlessly. The future of S2ST holds great promise, potentially transforming how we connect across cultures and languages.

3. Model

The proposed model for the React-based web transcription and translation application aims to utilize pre-trained AI models to offer real-time transcription and translation services. The system architecture is divided into front-end and back-end components, with each playing a vital role in ensuring a smooth user experience.

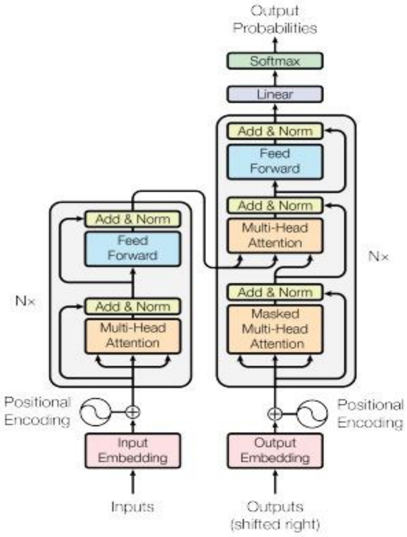


Fig 1. Hugging face architecture of prebuilt AI model for transcription

3.1 Front end

The front end of the application is developed using the React framework, selected for its component-based architecture, which enhances reusability and maintainability.

The user interface (UI) is composed of various components, including an audio input component for uploading or recording live audio, a transcription display component that shows real-time transcriptions, and a translation display component that presents the translated text in the selected language.

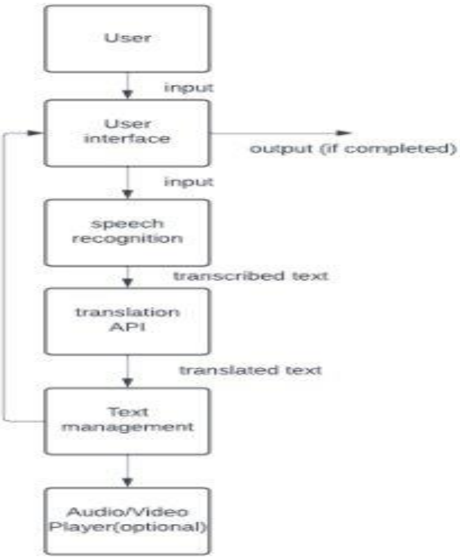


Fig 2. Basic block diagram representation of transcription and translation website.

State management is efficiently handled using Context API or Redux, ensuring smooth and responsive interactions. User authentication and authorization are managed through services like Firebase Auth or OAuth, providing secure access to the application.

3.2 Back end

On the back end, the application utilizes a Node.js server with Express.js to manage HTTP requests and routing. The back-end services are responsible for data processing, which involves integrating external NLP and ML models for conversion of audio into text and translation.

For the translation service, various models were evaluated to assess their performance, including:

- M2M100 (Facebook)
- mBART50 (Facebook)
- MarianMT (Hugging Face)
- Xenova NLLB 600M Transformer

These models integrate with the back end to provide high-quality translations across multiple languages. The application also includes a robust database system, either NoSQL (e.g., MongoDB) or SQL (e.g., PostgreSQL), to store user data, transcriptions, translations, and user preferences securely. This back-end architecture is designed to handle high computational loads and ensure minimal latency, which is crucial for real-time applications.

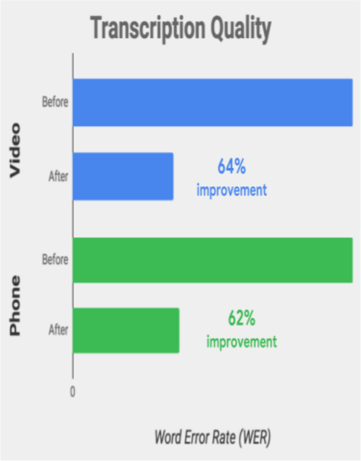


Fig 3. Increase in Transcription Quality

3.3 Performance Comparison of Translation Models

A crucial aspect of the application's backend is selecting the appropriate translation model to ensure optimal performance and accuracy. Below is a comparison of performance of the models that were tested, based on:

- Time to produce translation (normal vs cache access)
- Translation accuracy (measured by BLEU score)

Model	Normal Access Time(s)	Cache Access Time(s)	BLEU Score (Accuracy)
M2M100	1.75	0.05	32.5
mBART50	1.25	0.05	29.5
Marian MT	0.75	0.05	27.5
Xenovo nllb 600M	1.10	0.05	32.5

Table No 1. Time and Accuracy Comparison of Translation Models

The table and chart above provide a quick comparison of the models tested in the application. While MarianMT is the fastest model in terms of normal access time, M2M100 and Xenova NLLB 600M Transformer provide slightly better accuracy in terms of BLEU scores. Cache access times are significantly reduced across all models, as results are retrieved directly from cache rather than being recomputed.

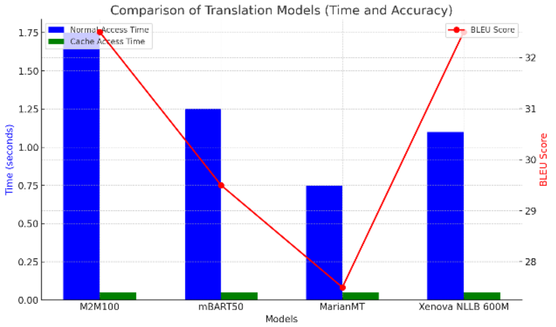


Fig 4. Comparison of Translation Models based on time and accuracy

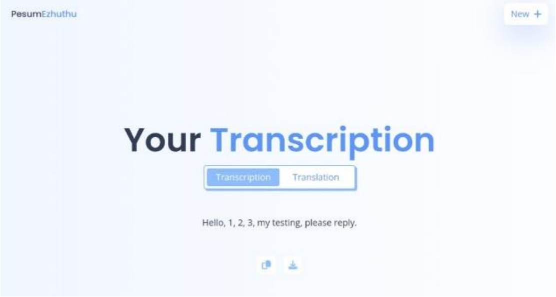
3.4 Overall System Architecture and Benefits

The integration of these components ensures high performance by leveraging advanced NLP models like M2M100, mBART50, MarianMT, and Xenova NLLB 600M Transformer. With efficient back-end services powered by Node.js and Express.js, the application handles real-time transcription and translation with minimal latency.

The architecture is designed for scalability, supporting multiple users and high-volume requests without performance degradation. Caching mechanisms further optimize response times, making the application both fast and reliable. Overall, the system offers a user-friendly interface, ensuring seamless and accurate communication across languages.

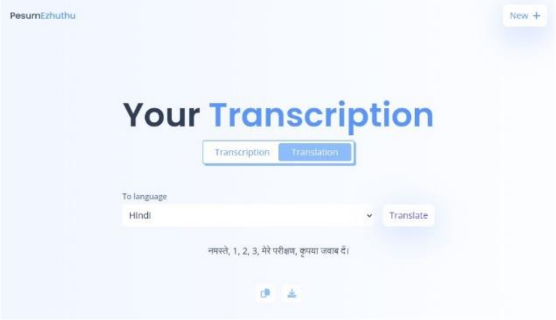
4. Result

These are the result of our platform showcasing the transcription and translation process



Hello, 1, 2, 3, my testing, please reply.

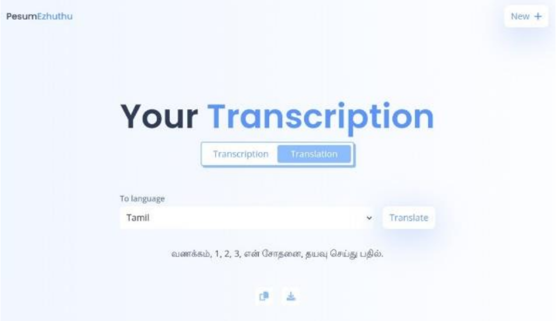
Fig 5. Transcribed audio text



To language: Hindi

नमस्ते, 1, 2, 3, मेरे परीक्षण, कृपया जवाब दें।

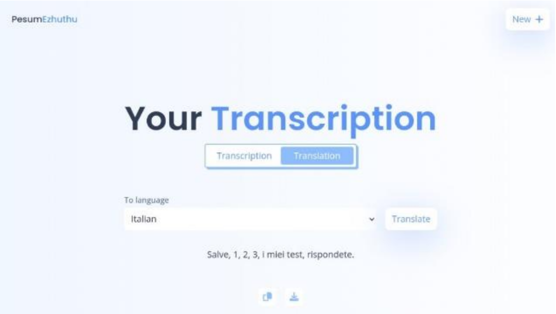
Fig 6. Translated text to Hindi



To language: Tamil

வணக்கம், 1, 2, 3, என் சோதனை, தயவு செய்து பதில்.

Fig 7. Translated text to Tamil



To language: Italian

Salve, 1, 2, 3, i miei test, rispondete.

Fig 8. Translated text to Italian

4. Conclusion

The creation of the React-based web transcription and translation application marks a substantial advancement in real-time language processing and communication technologies. By accompanying natural language processing (NLP) and machine learning (ML) models, the application provides an efficient and dependable solution for transcribing spoken language and translating text among various languages. This comprehensive system, designed on a strong architecture that merges the flexibility of the React framework with powerful back end services, guarantees a seamless and intuitive user experience.

The application addresses the increased wants and needs for optimal communication tools in an increasingly globalized world. It provides valuable support, where language barriers often pose significant challenges. The real-time transcription feature enables users to capture spoken language accurately, while the translation component facilitates immediate understanding and communication in different languages.

Performance evaluations and user feedback have shown that the application achieves high accuracy, responsiveness, and overall effectiveness. Its modular design facilitates easy scalability and future improvements, such as the integration of additional languages and enhanced NLP models.

5. References

[1] A. Babu, C. Wang, A. Tjandra, K. Lakhotia, Q. Xu, N. Goyal, K. Singh, P. von Platen, Y. Saraf, J. Pino et al., “XLS-R: Self-supervised Crosslingual Speech Representation Learning at Scale,” in Proc. ISCA Inter-speech, 2022, pp. 2278–2282.

[2] Y. Liu, J. Gu, N. Goyal, X. Li, S. Edunov, M. Ghazvininejad, M. Lewis, and L. Zettlemoyer, “Multilingual denoising pre-training for neural machine translation,” Transactions of the Association for Computational Linguistics, vol. 8, pp. 726–742, 2020.

[3] X. Li, C. Wang, Y. Tang, C. Tran, Y. Tang, J. M. Pino, A. Baevski, A. Conneau, and M. Auli, “Multilingual Speech Translation from Efficient Finetuning of Pretrained Models,” in Proc. Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), 2021.

[4] L. Xue, N. Constant, A. Roberts, M. Kale, R. Al-Rfou, A. Siddhant, A. Barua, and C. Raffel, “mT5: A Massively Multilingual Pre-trained Text-to-Text Transformer,” in Proc. Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, 2021, pp. 483–498.

[5] A. Bapna, C. Cherry, Y. Zhang, Y. Jia, M. Johnson, Y. Cheng, S. Khanuja, J. Riesa, and A. Conneau, “mSLAM: Massively multilingual joint pre-training for speech and text,” arXiv preprint arXiv:2202.01374, 2022.

[6] S. Khurana, A. Laurent, and J. Glass, “SAMU-XLSR: Semantically Aligned Multimodal Utterance-level Cross-Lingual Speech Representation,” IEEE Journal of Selected Topics in Signal Processing, pp. 1–13, 2022.

[7] Y. Jia, M. T. Ramanovich, T. Remez, and R. Pomerantz, “Translatotron 2: High-quality direct speech-to-speech translation with voice preservation,” in Proc. International Conference on Machine Learning, 2022, pp. 10 120–10 134.

[8] Y. Jia, R. J. Weiss, F. Biadsy, W. Macherey, M. Johnson, Z. Chen, and Y. Wu, “Direct Speech-to-Speech Translation with a Sequence-to-Sequence Model,” in Proc. ISCA Interspeech, 2019, pp. 1123–1127.

[9] A. Lee, P.-J. Chen, C. Wang, J. Gu, S. Popuri, X. Ma, A. Polyak, Y. Adi, Q. He, Y. Tang et al., “Direct Speech-to-Speech Translation with Discrete Units,” in Proc. Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), 2022, pp. 3327– 3339.

[10] A. Lee, H. Gong, P.-A. Duquenne, H. Schwenk, P.-J. Chen, C. Wang, S. Popuri, J. Pino, J. Gu, and W.-N. Hsu, “Textless speech-to-speech translation on real data,” in Proc. Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, 2022, pp. 860–872.

[11] P.-A. Duquenne, H. Gong, and H. Schwenk, “Multimodal and multilingual embeddings for large-scale speech mining,” in Proc. Advances in Neural Information Processing Systems (NeurIPS), vol. 34, 2021, pp. 15 748–15 761.

[12] P.-A. Duquenne, H. Gong, B. Sagot, and H. Schwenk, “T-Modules: Translation Modules for Zero-Shot Cross-Modal Machine Translation,” in Proc. Conference on Empirical Methods in Natural Language Processing (EMNLP), 2022, pp. 5794–5806.

[13] Y. Jia, M. T. Ramanovich, Q. Wang, and H. Zen, “CVSS corpus and massively multilingual speech-to-speech translation,” in Proc. Language Resources and Evaluation Conference (LREC), 2022, pp. 6691–6703.

[14] W.-N. Hsu, B. Bolte, Y.-H. H. Tsai, K. Lakhotia, R. Salakhutdinov, and A. Mohamed, “Hubert: Self-supervised speech representation learning by masked prediction of hidden units,” IEEE/ACM Transactions on Audio, Speech, and Language Processing, vol. 29, pp. 3451–3460, 2021.

[15] A. Polyak, Y. Adi, J. Copet, E. Kharitonov, K. Lakhotia, W.-N. Hsu, A. Mohamed, and E. Dupoux, “Speech Resynthesis from Discrete Disentangled Self-Supervised Representations,” in Proc. ISCA Interspeech, 2021.

[16] J. Kong, J. Kim, and J. Bae, “Hifi-gan: Generative adversarial networks for efficient and high-fidelity speech synthesis,” in Proc. Advances in Neural Information Processing Systems (NeurIPS), vol. 33, 2020, pp. 17 022–17 033.

[17] C. Wang, M. Riviere, A. Lee, A. Wu, C. Talnikar, D. Haziza, M. Williamson, J. Pino, and E. Dupoux, “VoxPopuli: A large-scale multilingual speech corpus for representation learning, semi-supervised learning and interpretation,” in Proc. Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), 2021, pp. 993– 1003.

[18] K. Lakhotia, E. Kharitonov, W.-N. Hsu, Y. Adi, A. Polyak, B. Bolte, T.-A. Nguyen, J. Copet, A. Baevski, A. Mohamed, and E. Dupoux, “On generative spoken language modeling from raw audio,” Transactions of the Association for Computational Linguistics, vol. 9, 2021.
[19] A. Pasad, B. Shi, and K. Livescu, “Comparative layer-wise analysis of self-supervised speech models,” arXiv preprint arXiv:2211.03929, 2022.

[20] W.-N. Hsu, Y.-H. H. Tsai, B. Bolte, R. Salakhutdinov, and A. Mohamed, “HuBERT: How much can a bad teacher benefit ASR pre-training?” in Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2021, pp. 6533–6537.

[21] M. Ott, S. Edunov, A. Baevski, A. Fan, S. Gross, N. Ng, D. Grangier, and M. Auli, “fairseq: A fast, extensible toolkit for sequence modeling,” in Proc. Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, 2019, pp. 48–53.

[22] Q. Xu, A. Baevski, T. Likhomanenko, P. Tomasello, A. Conneau, R. Collobert, G. Synnaeve, and M. Auli, “Self-training and pre-training are complementary for speech recognition,” in Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2021, pp. 3030–3034.

[23] J. Shen, R. Pang, R. J. Weiss, M. Schuster, N. Jaitly, Z. Yang, Z. Chen, Y. Zhang, Y. Wang, R. Skerrv-Ryan et al., “Natural tts synthesis by conditioning wavenet on mel spectrogram predictions,” in Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2018, pp. 4779–4783