

Fostering Inclusivity: ML-Driven Voice to Sign Language Translation

Ananya¹, Bipin Krishna G. A², Manohar B N³, Navaneeth Y⁴ and T M Kiran Kumar⁵

¹⁻⁵Department of Computer Science and Engineering, Siddaganga Institute of Technology, Tumakuru, India
Email: 1si20cs058@sit.ac.in, 1si20cs012@sit.ac.in, 1si20cs136@sit.ac.in, 1si20cs058@sit.ac.in, 1si20cs063@sit.ac.in, tmkiran@sit.ac.in

Abstract—Social inclusivity is essential for everyone, but communication can be challenging for Deaf and Hard of Hearing individuals. Sign language bridges this gap, enabling them to express ideas, thoughts, and experiences without sound. Our project aims to develop a Machine Learning-based Voice to Sign Language Translation system to make communication smoother in various fields like education, healthcare, and customer service. We're using diverse datasets and advanced technology to create a real-time translation app that's easy to use. With a clear interface and support for multiple sign languages, this app will help break down communication barriers and promote inclusivity worldwide. Our goal is to make this technology accessible, scalable, and adaptable for everyone.

Index Terms— Machine Learning, Real-time translation, Datasets, Advanced technology, User-friendly interface, Multi-language support, Scalability, Adaptability.

I. INTRODUCTION

Reducing communication barriers for the Deaf and Hard of Hearing is paramount for fostering accessibility and inclusivity. To tackle this challenge, our ambitious project introduces a voice-to-sign language translation system leveraging machine learning. Our core objective is to enable real-time translation of spoken language into sign language, revolutionizing interactions across social settings, customer service, healthcare, and education. By amalgamating diverse datasets, we train sophisticated machine learning models like sequence-to-sequence and neural machine translation for precise translations.

Prioritizing user-friendly design, we utilize Python and other tools to convert voice input to text, process it through neural machine translation, and match words with corresponding sign language motions, ensuring dynamic communication.

The anticipated outcome is a robust application swiftly and accurately translating spoken words into sign language, facilitating natural conversations with a minimalist interface. Additionally, our system supports multiple sign languages, maximizing accessibility for diverse global audiences.

A. Problem Statement

Develop a Speech to Sign Language translation model to overcome communication barriers within the Deaf and Hard of Hearing community. Prioritize real-time, accurate translations for inclusivity in various domains. Utilize machine learning, focusing on user-friendly integration and global accessibility. Create a cost-effective solution that dynamically enhances communication, ensuring practicality and adaptability for widespread use.

B. Objectives

- Real-Time Sign Language Production from Speech.
- App for Real-World Use.
- Multi-Language Support.
- Minimalist UI Design Showing Pose Signs.

II. PROPOSED METHOD

Using machine learning models, “SignBridge” entails creating a speech-to-sign Flutter application involving an organized process that starts with carefully obtaining requirements, comprehending user needs in their whole, and defining the project’s parameters. A comprehensive research and planning phase follows, exploring current speech-to-sign systems and sequencing models, with an emphasis on efficiently architecting the Flutter frontend and the backend seq2seq model. Before designing the architectural architecture of the seq2seq model, which includes speech recognition and sign language generation components, the user interface (UI) of the Flutter app is designed, supported by wireframes and mockups. Additionally, a seq2seq model for the production of sign language is incorporated.

A. High Level Design

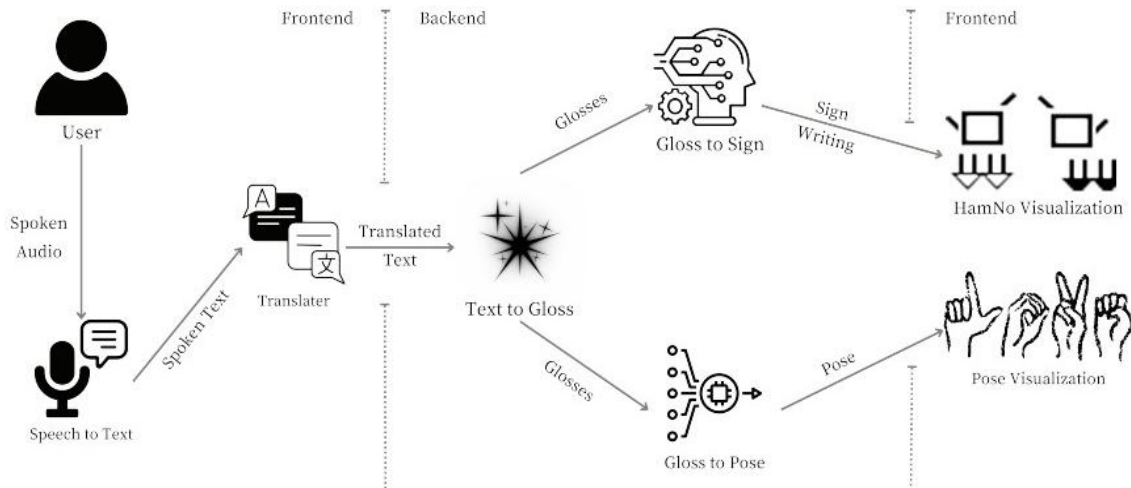


Fig 1. High level Design

At a high level, the system design encompasses modules for voice input processing, language translation, gloss generation, FSW string conversion, Ham notation generation, dictionary lookup, and pose construction. The voice input module supports multiple languages and feeds into the translation module, which converts it into English text. The gloss generation module utilizes a sequence-to-sequence model to derive gloss from the English text. Concurrently, the FSW string is generated from the gloss using neural machine translation techniques. Ham notations are then produced from the FSW string and displayed to the user. Additionally, the gloss facilitates dictionary lookup on sign video datasets, aiding in the construction of poses. These poses, along with the Ham notations, form the final output, completing the high-level design aimed at translating voice input into sign language gestures with support for multiple languages. The dictionary lookup on sign video datasets enhances the accuracy of pose construction by referencing real sign language videos, ensuring that the final poses are realistic and comprehensible. This integrated approach not only translates voice input into sign language but also ensures that the gestures are accurate, fluid, and contextually appropriate, making communication seamless and effective for users.

B. Low Level Design

The low-level design comprises three key parts: text to gloss conversion, gloss to sign conversion, and gloss to pose construction. Firstly, in the text to gloss conversion, the system employs a sequence-to-sequence model trained with datasets to generate gloss from the English text obtained through language translation. Secondly, in the gloss to sign language gesture conversion, the generated gloss is utilized for dictionary lookup on sign video datasets, mapping each gloss to its corresponding sign language gesture. Lastly, in the gloss to pose construction,

the gloss is processed to derive a FSW string using neural machine translation techniques. This FSW string is then transformed into Ham notations, which are displayed as poses to the user. This low-level design ensures the systematic transformation of textual input into sign language gestures and poses, facilitating effective communication for users across diverse linguistic backgrounds.

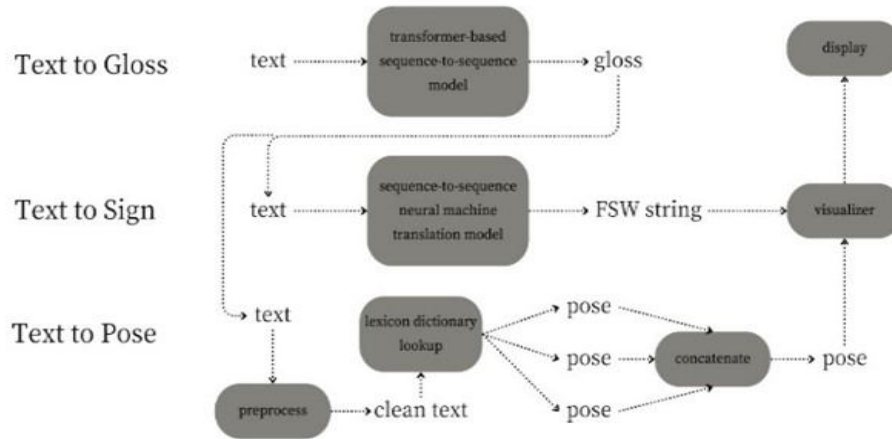


Fig 2. Low Level Design

C. Flowchart

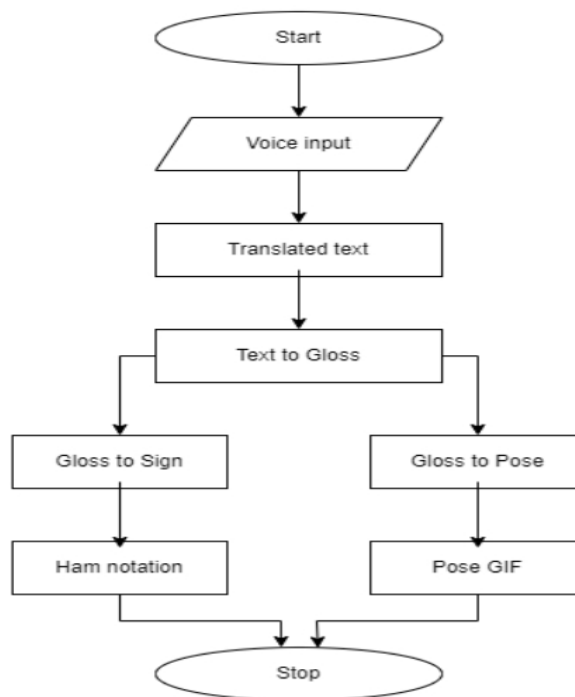


Fig 3. Flowchart

The process initiates by accepting voice input from users, accommodating multiple languages. This input undergoes language translation to English and subsequent conversion into text through a translator package. The English text is then utilized to generate gloss via a sequence-to-sequence model, trained with datasets. From this gloss, a FingerSpelled Word (FSW) string is derived using neural machine translation techniques. The FSW string is further transformed into Ham notations, which are then displayed. Additionally, the gloss facilitates dictionary lookup on sign video datasets, aiding in the construction of poses.

Finally, these poses are presented, thus completing the process of translating voice input into sign language gestures, with support for multiple languages.

III. IMPLEMENTATION

A. Backend server API development

- Utilized Flask or Django framework in Python to develop a backend server API.
- Exposed endpoints specifically for handling text and gloss.
- Responsibilities of these endpoints include gloss generation, FSW string conversion, Ham notation generation, dictionary lookup, and pose construction.

B. Gloss Generation and Language Processing

- Implemented gloss generation using sequence-to-sequence models and neural machine translation.
- Utilized TensorFlow and PyTorch libraries for efficient gloss generation from the provided text data.

C. Integration with Flutter App

- Established seamless communication between the backend API and the Flutter app using HTTP requests.
- The Flutter app handles user interactions and display the translated sign language gestures and poses received from the backend API.

D. User Interface and Interaction

- Designed a user-friendly interface in the Flutter app to facilitate user interactions, such as voice input and viewing translated sign language gestures.
- Intuitive navigation and clear presentation of translated content to enhance user experience.

E. Real-Time Translation and Multiple Language Support

- Real-time translation of voice input into sign language gestures with support for multiple spoken languages.
- * Implemented facilities for sign language production and language translation to accommodate users with varying linguistic backgrounds.

F. Error Handling and Reliability

- Implemented robust error handling mechanisms in both the backend API and Flutter app to handle unexpected scenarios gracefully.
- Meaningful error messages to users to assist them in troubleshooting issues effectively.

G. Deployment

- Deployed the backend API on Hugging face to accommodate varying user loads and ensure high availability.

IV. RESULTS



Fig 4. Home Screen Snapshot

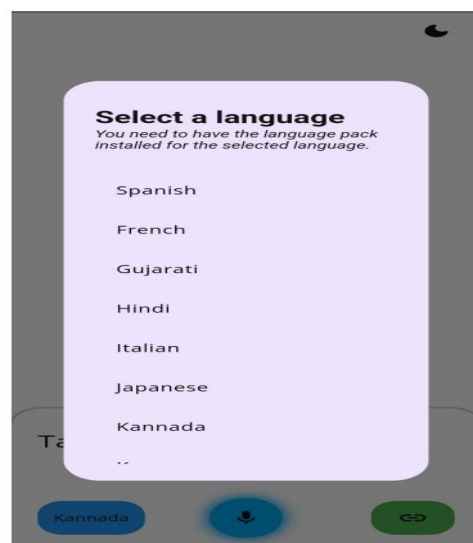


Fig 5. Language Change Option Snapshot



Fig 6. Result Snapshot

V. TESTING

A. Unit Testing

Unit testing for the ML-Driven Voice to Sign Language Translation system involves testing individual components to ensure their proper functionality and behavior.

Here are some examples of unit tests

- Voice Input Processing: Ensure accurate recognition of voice input in multiple languages.
- Text Input Processing: Verify correct interpretation and processing of text input in English.
- Multiple Language Support: Verify that translations from sign language to several spoken languages are generated accurately.
- User Interface Customization: To make sure the display is working properly, try flipping between the light and dark modes.

B. Integration Testing

Integration testing ensures that all components of the system function correctly when integrated together. Here's an outline of integration testing procedures:

- Third-Party Service Integration: Test integration with third-party services like speech recognition and translation API to ensure seamless operation and data exchange.
- Processing and Sign Language Generation: To guarantee accurate translation, verify the integration of the sign language generation modules and voice input processing modules.

VI. CONCLUSION

- Impact on Accessibility: The Voice to Sign Language Translation system addresses critical communication barriers faced by the Deaf and Hard of Hearing community, significantly enhancing accessibility in various aspects of daily life.
- Technological Innovation: Leveraging cutting-edge technologies such as sequence to-sequence models, neural machine translation, and sign video datasets showcases the project's commitment to technological innovation and advancement.

- **User-Centric Design:** The user-friendly interface of the Flutter app, coupled with seamless integration with the backend server API, demonstrates a user-centric approach aimed at enhancing the user experience and ensuring ease of use for all users.
- **Global Reach:** With support for multiple spoken languages and sign languages, the system has the potential for global impact, catering to diverse linguistic backgrounds and fostering inclusivity on a global scale.

In conclusion, the development of the Voice to Sign Language Translation system represents a significant step forward in enhancing accessibility and inclusivity for the Deaf and Hard of Hearing community. By leveraging advanced technologies such as sequence-to-sequence models, neural machine translation, and sign video datasets, the system aims to bridge communication gaps and facilitate natural and fluid conversations in sign language.

SCOPE OF FUTURE WORK

- **Expansion of Language Support:** The project can be expanded to support additional spoken languages and sign languages, catering to a broader and more diverse user base worldwide.
- **Improvement of Translation Accuracy:** Continuous refinement of machine learning algorithms and models can enhance the accuracy and fluency of sign language translation, ensuring more precise and natural communication.
- **Integration of Advanced Features:** Incorporating features such as displaying sign using Avatars, enabling the system to function locally, without relying on internet connectivity etc.
- **Accessibility Enhancements:** Further accessibility enhancements, such as compatibility with assistive devices and compatibility with wearable technology, can empower users with greater independence and accessibility in various environments.
- **Collaboration with Sign Language Communities:** Collaboration with sign language experts, educators, and members of the Deaf and Hard of Hearing community can provide valuable insights and feedback for improving the system's usability, accuracy, and relevance.
- **Deployment in Specific Domains:** Tailoring the system for specific domains such as education, healthcare, customer service, and emergency response can address unique communication challenges and provide targeted solutions for those sectors.
- **Research and Development:** Continued research and development in the field of sign language translation, including exploration of novel techniques, datasets, and applications, can drive innovation and push the boundaries of what is possible in sign language communication technology.
- **Global Adoption and Impact:** Promoting the adoption of the system on a global scale through partnerships, advocacy efforts, and community engagement initiatives can maximize its impact and reach, ensuring that it benefits individuals and communities worldwide.

REFERENCES

- [1] A. Yadav, R. Saxena, B. Saini, V. K. Verma and V. Srivastava, "Audio to Sign Language Translator Web Application," 2021 International Conference on Computational Performance Evaluation (ComPE), Shillong, India, 2021, pp. 321-326, doi: 10.1109/ComPE53109.2021.9751857.
- [2] P. Sonawane, K. Shah, P. Patel, S. Shah and J. Shah, "Speech to Indian Sign Language (ISL) Translation System," 2021 International Conference on Computing, Communication, and Intelligent Systems (ICCCIS), Greater Noida, India, 2021, pp. 92-96, doi: 10.1109/ICCCIS51004.2021.9397097.
- [3] N. J. S. Bhat, R. R. Nair and T. Babu, "Audio to Sign Language conversion using Natural Language Processing," 2022 3rd International Conference on Communication, Computing and Industry 4.0 (C2I4), Bangalore, India, 2022, pp. 1-6, doi: 10.1109/C2I456876.2022.10051414.
- [4] B. D. Patel, H. B. Patel, M. A. Khanvilkar, N. R. Patel and T. Akilan, "ES2ISL: An Advancement in Speech to Sign Language Translation using 3D Avatar Animator," 2020 IEEE Canadian Conference on Electrical and Computer Engineering (CCECE), London, ON, Canada, 2020, pp. 1-5, doi: 10.1109/CCECE47787.2020.9255783.
- [5] M. M. Reda, N. G. Mohammed and R. A. Abdel Azeem Abul Seoud, "SVBiComm: Sign- Voice Bidirectional Communication System for Normal, "Deaf/Dumb" and Blind People based on Machine Learning," 2018 1st International Conference on Computer Applications and Information Security (ICCAIS), Riyadh, Saudi Arabia, 2018, pp.1-8, doi: 10.1109/CAIS.2018.8441985.
- [6] P. Wang, N. Yin and K. Sujatha, "A Web-Based Audio-to-Sign Language Converter for Chinese National Sign Language," 2022 4th International Academic Exchange Conference on Science and Technology Innovation (IAECST), Guangzhou, China, 2022, pp. 1045-1048, doi: 10.1109/IAECST57965.2022.10061950.