Next-Gen Communication: LSTM-Powered Speech-to-Sign Language Translation

Vinaya Kulkarni[1], Pranoti Kale[2], Sanika Chaudhari[3], Shruti Bhumkar[4], Manasi Deshmukh[5], Samruddhi Deshmukh[6]

[1]*Assistant Professor, Computer Engineering Department, Bharati Vidyapeeth's College of Engineering for Women, Pune, Maharashtra, India*

[2]*Associate Professor, Computer Engineering Department, Bharati Vidyapeeth's College of Engineering for Women, Pune, Maharashtra, India*

[3,4,5,6]*Student, Computer Engineering Department, Bharati Vidyapeeth's College of Engineering for Women, Pune, Maharashtra, India*

Abstract*—Deaf and Hard of Hearing (DHH) individuals experience a lot of difficulties and are deprived of access to video material. Due to the increasing demand for audio-visual communication, this makes it difficult for the DHH community to access the material. Owing to the high use of audio-visual content delivery in contemporary society hampers the DHH community's accessibility to the content being passed across. Addressing the above issues, our system overcomes the Communication barrier by applying highly advanced AI-based video translation methods that translate spoken content to Sign Language through GIFs, making it a more natural and accessible experience to the users. Our framework integrates diverse algorithms, including Sequence-to-sequence (Seq2seq), and Long Short-Term Memory (LSTM) – a type of Recurrent Neural Network. This approach aims to capitalize on the strengths of individual models, optimizing the overall performance.

Index Terms*— Sign Language Translation, Video Processing, Audio Extraction, Audio-to-Text Conversion, Text Preprocessing, Contextual Analysis with BERT, Gesture Synchronization, LSTM Networks, Speech Recognition, Natural Language Processing (NLP).

I. INTRODUCTION

In this fast pace of technology, where all the things are opting to go digitally. Digital content is in multimedia form such as text, audio, images, videos, and animation. As video content has become a major part of communication, entertainment and education, various platforms like YouTube rely more on audio-visual formats to convey information that engages the audiences effectively. This leads to communication barriers between hearing and non-hearing individuals bringing challenges in day-to-day life. Often, with a gap present between the languages of communication, it makes it hard to interact with each other.

According to the World Health Organization (WHO) over 5% of the global population, approximately 430 million people experience loss of hearing and hard-to-hear problems indicating the growth of this number could rise higher in future. One way of reducing the gap between hearing and non-hearing is the use of Sign language, such as ISL (Indian Sign Language), and ASL (American Sign Language) which were properly developed according to their syntax and grammar for communication. It is a visual means of communication using hand gestures, body movements and facial expressions. This created a significant barrier, especially in consuming digital content like lectures, educational videos, conferences, and news articles where sign language is a must. Despite the growing awareness of inclusivity, very few platforms offer solutions that translate spoken language into videos based on sign language. The few existing solutions either rely on manual interpretation, which is not scalable, or are limited to very specific applications or languages, often using pre-recorded videos of sign language interpreters. These solutions are inadequate in a world that demands real-time, automated, and scalable sign language interpretation.

To address these challenges, this project employs an AI-driven platform that inherently translates video content into sign language. This incorporates extracting audio from the video, further converting it to text using various advanced methods such as speech-to-text engines or APIs such as IBM Watson [9], Librosa and other Python libraries, and translating this into Sign language gestures.[1]

These gestures are synchronised with the video content to ensure they follow the correct sequence. It employs deep learning architectures such as CNN for visual recognition and LSTM for sequencing gestures in the real-time translation of spoken language into sign language.[2] The solution is built upon using advanced tools in NLP (Natural Language Processing), speech recognition, machine learning, and deep learning, making use of custom ISL datasets to train the machine learning models.[3] By making use of these advanced technologies, the system offers an automated and scalable solution to sign language translation, thus bridging the communication gap in video content and fostering inclusivity for the deaf community.

By addressing the limitations of current methodologies, this project aims to contribute to more inclusive communication tools, ultimately enhancing interactions for individuals with hearing impairments.

II. LITERATURE REVIEW

Shailesh Kumar, Kushank Saraswat, and Sumit Prasad present how to evaluate the audio of YouTube videos by segmenting it into manageable pieces to improve processing and using specialized libraries like Librosa, PyDub, and Kaldi in optimizing audio for accuracy and scalability [1]. A paper that converts speech-text conversion using Deep Learning Neural Network Methods [2] presents a study that increases efficiency for applications like voice-based email systems by investigating several methods for turning spoken language into text for precise speech-to-text (STT) conversions. CNN is also employed for its capacity to handle high-dimensional speech recognition features, utilizing Hidden Markov Models (HMM) in conjunction with Artificial Neural Networks (ANN) and Convolutional Neural Networks (CNN).

In "Audio to Sign Language Using NLTK" [3], the paper presents a 3D avatar-based sign language learning system that effectively translates spoken language into ISL and enhances communication accessibility, allowing users to interact with the system via voice input. This input is then processed into sign language animations using NLP, Blender software for 3D Avatar Animation, and Python's Speech Recognition library. However, its scope is limited in supporting multiple languages and expanding its vocabulary.

Extracting audio from video using Python [4] details a simple mechanism for extracting audio in MP3 or WAV format by efficiently segregating audio. The paper focuses on user-friendly methods and tools like the MoviePy library for MP3 audio format. A study of a system that uses different sign languages [5] achieved an 80.3% recognition rate. It shows promising results in translating spoken English to sign language using speech and image processing. The system employs Speech Recognition and Vector Quantization for speech recognition, aiming to improve communication between normal people and those with hearing or speech disabilities, especially within the Malaysian Deaf Community.

A study [6] presents a rule-based translation mechanism that identifies the structure of English sentences and converts them into ISL glosses. The translation process involves three main phases: Pre-processing, Grammar transfer rules, and Post-processing. Techniques such as Multi-Word Expression (MWE) detection and synonym substitution enhance translation accuracy. The tools and techniques used include Stanza for linguistic analysis, NLTK for tokenization and processing multi-word expressions, and WordNet. Neural Sign Language Translation [7] emphasizes the need to consider the grammatical structures unique to sign languages and also created the first publicly available continuous SLT dataset, which includes over 0.95 million frames and more than 67,000 signs. Techniques used include CNN for Seq-to-Seq encoding and decoding networks, tokenization for mapping sign video to spoken language, and the Neural Machine Translation Framework for joint learning of spatial representations and language models.

U. Shiva Prasad, DR K. Anuradha, E. Siddartha, S. Sahith Reddy, and N. Prashanth Reddy [8] present a system that successfully converts English audio or text input into ISL using speech recognition and displays ISL as images or videos. This tool bridges communication gaps for the hearing-impaired. The system uses Speech recognition via Google Speech API, NLP for text pre-processing, and image processing for ISL conversion.

Speech to Indian Sign Language Translator [9] is a system that converts English audio to text according to ISL grammar rules. It generates ISL gloss and provides output in the form of video representation of ISL signs, improving user engagement using PyAudio for speech-to-text conversion, Google Cloud Speech API for audio processing, Hamburg

Sign Language Notation System, and IFrame API for video display.

The video-based sign translation model [10] presents a real-time system that utilizes a combination of deep learning algorithms to automate the recognition and translation of sign language into text and speech. The system achieves an overall accuracy rate of 85.46% when tested on datasets such as Sign Language MNIST, ASL Alphabet, and Sign Language Digits Dataset. It uses CNN for feature extraction from depth images of hand gestures and RNN to recognize temporal patterns in dynamic gestures, along with Google Text-To-Speech API.

A study of another system [11] presents a real-time sign language detection mechanism that can identify different hand gestures from a normal camera feed. For the interpretation of sign language without specialized equipment, an interactive system using SSD MobileNet V2 (a deep learning algorithm) is employed.

Automated Speech to Sign Language Conversion [12] presents a system that takes spoken words and converts them into American Sign Language (ASL) by processing the audio and finding a matching video from the sign language video library. It uses Google's Speech-to-Text API and NLP for processing.

Another study presents a 3D Avatar Approach [13], developing a system to convert speech/text into Indian Sign Language (ISL) using a 3D avatar. It demonstrated effective translation for both isolated words and complete sentences. It uses NLP, IBM Watson for speech-to-text conversion, Blender for 3D avatar creation and sign movement animation, and BLEU score and SER for evaluation.

Sign Language Translation with 3D Avatars [14] creates a gloss-video dictionary using existing datasets for sign language recognition, then converts sign videos into 3D representations using a specialized model, and finally converts spoken text into gloss sequences, retrieving corresponding 3D signs and dynamically connecting them to produce coherent translations. Tools used include the SMPLify-X model for estimating 3D signs, mBART for translating text into gloss sequences, and the TwoStream-SLR Model for segmenting continuous sign language videos into isolated signs.

Further study by Nayan Mehta, Suraj Pai, and Sanjay Singh [15] provides a standardized ISL teaching tool, reducing teacher workload by automating captions for online videos and using a 3D avatar to enhance learning, especially for young children. It uses Natural Language Processing, 3D animated avatars to map signs to video, Speech processing for subtitle conversion, Microsoft Kinect v2 for gesture recognition, and iClone software for 3D modelling.

III. PROPOSED MODEL

Our proposed model employs an advanced technique of converting video content translated to sign language by uploading the link for the video to generate sign language as the final output. We described each stage in the pipeline here, detailing the preprocessing and conversion steps needed to ensure that representation is smooth and well-crafted enough to be representative of sign language.

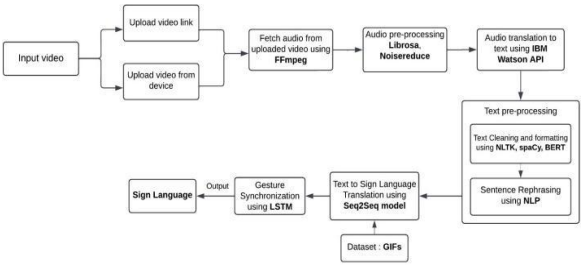

Fig. 1 Architecture of Proposed Model

A. *Video Upload and Audio Extraction:*

The process starts with a user uploading video content on our platform. Users can upload videos directly from their devices or share the URL link. It allows the platform to accommodate and account for the different needs and sources of the users. Supported formats include MP4, AVI, and MOV, among which are some of the popular video files that make it easier to use on other devices and platforms.

Once the user completes uploading, the system checks the format and quality of the video before audio extraction from the video. This check ensures good performance because poorly formatted or low-quality videos might cause some problems while trying to extract audio and then process it. So, videos not satisfying the quality parameters are tagged for action by the users to be sure that only the right content moves forward.

With the aid of video processing libraries like FFmpeg, we can extract the audio track of the uploaded video. The extracted audio is then saved into a suitable standard format, such as WAV or FLAC, to be used in subsequent processes. These formats are selected considering that the audio

processing libraries used in further stages could be of high fidelity and suitable. Audio files obtained from here are stored securely in a temporary location to adhere to the data-protection policies. This way, the audio content remains private and secure as it is processed within the platform. The extracted audio data is then directly passed on to the pre-audio processing stage

B. Pre - Process Audio :

Audio preprocessing is deemed crucial because its quality and uniformity directly affect the accuracy of the translation model and, of course, the audio file being translated. Other processes involved, such as normalization, noise reduction, segmentation, and feature extraction, can also be regarded as part of this phase.

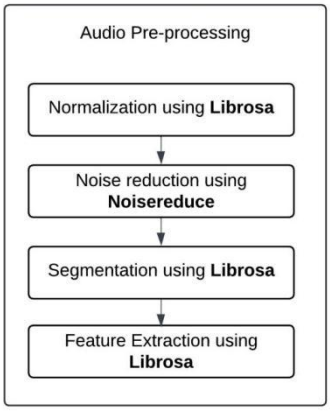

Fig. 2 Audio Pre-processing steps

Normalization: This is adjusting the volume across the audio track to a uniform level. In this manner, normalization ensures uniform loudness. Standardization of audio levels helps the model to process the audio without bias from volume fluctuations.

Noise Reduction: The next step involves noise reduction where all the background noise from the audio is filtered out. This improves the model's performance and efficiency.

Segmentation: Segmentation implies breaking down of the audio into smaller chunks rather than using it as a single block of audio. This simplifies working with the audio and makes audio processing much easier as every phrase is isolated, thus making transcription and analysis much more precise.

Feature Extraction: Some of the essential audio feature extraction used are MFCC, Chroma, and spectral contrast and we make use of the Librosa Python library. We put numerical representations to them to create input data for our machine learning models, thus ensuring reliable analysis and training for an accurate sign language translation.

C. Audio-to-Text Conversion :

The process for the conversion of audio to sign language text would be a major process in the text conversion. This may be done using various APIs that make the task of audio-to-text conversion faster and more efficient. Every API offers unique features and capabilities that suit different applications.

The proposed model uses the IBM Watson Speech to Text API which ensures the audio undergoes all the pre-requisites required by the API. Using an HTTP POST request, the processed audio is first sent to the API. Following this, IBM Watson handles the audio processing and accurate transcription into text.

D. Text Pre-Processing

The preprocessing stage ensures the transcribed text will be clean, well-structured, and properly formatted. That's the groundwork for doing the entire process of text-to-sign language translation accurately. Thus, it implies cleaning, standardization, and contextual language processing.

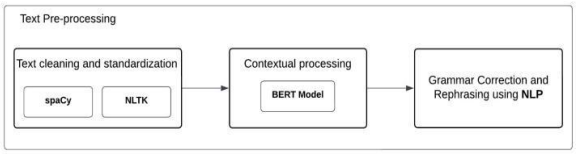

Fig. 3 Text Pre-processing steps

Text Cleaning and Standardization: It makes sure that E2E text preprocessing to the transcribed text is good, clean and formatted properly, creating a base for Text to Sign Language translation. It cleans, normalizes, and performs contextual language processing on them. We eliminate excess characters, punctuations and other irregularities in the text by using NLTK and spaCy. This cleaning part of raw text reduces complexity and the challenging nature of the task for the model.

Contextual Processing with BERT: Using the BERT model, we obtain the contextual meaning of words (which is very important for correctly translating phrases). BERT is particularly important to make sure that every single word or phrase can be correctly mapped onto its corresponding sign language gesture, allowing both languages to keep their meaning.

Grammar Correction and Rephrasing: Natural language grammar is different from sign language

grammar, and this difference leads us to apply NLP-based grammar correction to paraphrase and rearrange sentences.

E. Text to Sign Language Translation :

After pre-processing the text it is fed into the translation model which follows the sequence-to-sequence (seq2seq) paradigm. The task of this model is to map different words and phrases with gestures of sign language used contextually. Each such segment of text is then linked with appropriate GIFs illustrating the sign gestures, enabling the creation of a logical flow. The final output provides a smooth and clear translation of the text in sign language using an orderly series of signs in the form of GIFs that reproduce the content of the source language in a visual sign language.

F. Gesture Synchronization with LSTM :

Hereafter, it becomes highly crucial to follow the synchronization of gestures with smooth content flow. This means that all the gestures have to be sequenced appropriately in time as well concerning when the audio playback happens. For efficient analysis of time and proper synchronization, the proposed model uses a Long Short-Term Memory (LSTM) algorithm to ensure accuracy. This is particularly helpful when specific phrases have widely accepted, context-appropriate signs which allows the LSTMs to identify those phrases and automatically predict the correct signs in sequence.This makes the platform make smooth transitions making it naturally fluid as perceived by the deaf people on the platform. As LSTMs learn from sequences, we can directly map words, phrases or sentences to the corresponding gesture sequences. Using the LSTM networks in our model, we allow it to recognize context-appropriate signs, and predict gesture sequences that fit into familiar sign language phrases. This synchronization will give rise to a fluent, comfortably-paced sign language output that is perceptually natural.

IV. CONCLUSION AND FUTURE SCOPE

The project aims to bridge the communication gap between individuals who are deaf or hard of hearing and the broader population by converting video content into sign language using deep learning techniques. The system integrates multiple machine learning methods, such as IBM Watson's Speech-to-Text API, Librosa for audio preprocessing, and BERT for text processing, enabling the efficient conversion of speech into text and translating that into sign language. By incorporating gesture synchronization using LSTM models, the system ensures fluid, natural sign language translation, making video content more accessible. Future developments could include expanding the gesture database to cover more regional and international sign languages, integrating facial expressions to improve the expressiveness of translations, and enabling real-time translation for live content such as video calls or conferences. Additionally, adding offline functionality, supporting multi-language sign translations would significantly broaden the scope of the system. User feedback and customization features, along with integration with wearables and AR technologies, could further enhance the user experience, making sign language translation more natural and versatile in various real-world contexts.

VI. REFERENCES

[1] S. Kumar, K. Saraswat, and S. Prasad, "Audio Extraction from Video," International Journal of Research in Engineering and Science (IJRES), vol. 11, no. 3, pp. 45–49, May 2023.

[2] B. Pandipati and R. Praveen Sam, "Speech to Text Conversion Using Deep Learning Neural Net Methods," Turkish J. Computer and Mathematics Education, vol. 12, no. 5, pp. 2037–2042, 2021.

[3] H. Kotha, S. D. Ponugoti, and V. Krishnan, "Audio to Sign Language Using NLTK," vol. 10, no. 6, pp. 1–5, Jun. 2023.

[4] U. S. Prasad, K. A. Anuradha, E. Siddartha, S. S. Reddy, and N. P. Reddy, "Audio/Text to Sign Language," Intl. J. Creative Research Thoughts, vol. 11, no. 5, pp. 1450–1456, May 2023.

[5] H. Monga, J. Bhutani, M. Ahuja, N. Maida, and H. Pande, "Speech to Indian Sign Language Translator," in Recent Trends in Intensive Computing, M. Rajesh, Ed., pp. 55–60.

[6] Ruthvik M, Sandhya AR, Shristi, Skandan PS, Dr. Manjunatha Kumar BH "Video based sign translation model"

[7] A. Pathak, A. Kumar, Priyam, P. Gupta, and G. Chugh, "Real-Time Sign Language Detection," Intl. J. for Modern Trends in Science and Technology, vol. 8, no. 1, pp. 32–37, 2022.

[8] Ritika Bharti, Sarthak Yadav, Sourav Gupta, Rajitha Bakthula"Automated Speech to Sign Language Conversion"
[9] D. D. Chakladar, P. Kumar, S. Mandal, P. P. Roy, M. Iwamura, and B. G. Kim, "3D Avatar Approach for Continuous Sign Movement Using Speech/Text," Basel, Switzerland, 2021.
[10] N. Mehta, S. Pai, and S. Singh, "Automated 3D Sign Language Caption Generation for Video," Springer-Verlag, Germany, 2019.
[11] Lan Thao Nguyen Florian Schicktanz Aeneas Stankowski Eleftherios Avramidis "Automatic generation of a 3D sign language avatar on ARglasses given 2D videos of human signers"
[12] O. M. Foong, T. J. Low, and W. W. La, "V2S: Voice to Sign Language Translation System for Malaysian Deaf People," presented at the Intl. Conf. on Modern Trends in Intensive Computing, Nov. 2009.
[13] R. Zuo, F. Wei, Z. Chen, B. Mak, J. Yang, and X. Tong, "A Simple Baseline for Spoken Language to Sign Language Translation with 3D Avatars," 3 July 2024
[14] R. M., S. A. R., S. Shristi, S. P. S., and M. Kumar B. H., "Video-Based Sign Translation Model,"
[15] S. Stoll, N. C. Camgoz, S. Hadfield, and R. Bowden, "Text2Sign: Towards Sign Language Production Using Neural Machine Translation and Generative Adversarial Networks," International Journal of Computer Vision, vol. 128, pp. 891–908, 2020.
[16] Abhigyan Ghosh,Radhika Mamidi "English To Indian Sign Language: Rule-Based Translation System Along With Multi-Word Expressions and Synonym Substitution"
[17] S. Sasavade, T. Sutar, K. Baral, and D. Kambale, "Extract the Audio from Video by Using Python," Intl. Research J. Engineering and Technology, vol. 10, no. 6, pp. 120–123, Jun. 2023.
[18] S. Sasavade, T. Sutar, K. Baral, and D. Kambale, "Extract the Audio from Video by Using Python," Intl. Research J. Engineering and Technology, vol. 10, no. 6, pp. 120–123, Jun. 2023.
[19] L. Goyal and V. Goyal, "Text to Sign Language Translation System: A Review of Literature," Intl. J. Synthetic Emotions, vol. 7, no. 2, pp. 20–28, Jul.–Dec. 2016.
[20] Z. Yu, S. Huang, Y. Cheng, and T. Birdal, "Sign Avatars: A Large-Scale 3D Sign Language Holistic Motion Dataset and Benchmark," in Proc. English to Indian Sign Language Rule-Based Translation System, May 2023.