AUDIO TO SIGN LANGUAGE TRANSLATOR USING PYTHON

Ms.Arunasree Bethapudi[1],Gurram Thapasvi[2],Hanmandla Hanishka[3],Kedari Jahnavi[4]

[1] Professor, School of CSE ,Malla Reddy Engineering College For Women(Autonomous Institution), Maisammaguda, Dhulapally,Secunderabad,Telangana-500100

[234]UG Student, Department of CSE,Malla Reddy Engineering College for Women, (Autonomous Institution), Maisammaguda,Dhulapally,Secunderabad,Telangana-500100

Mail Id:arunasrimrecw@gmail.com

ABSTRACT

Speech-to-text technology makes audio signals transformable into text form. This can be employed on small, medium and even large vocabulary systems. In these systems, voices inputted are processed through various algorithms and apis that convert those inputs into text. The paper hereby outlines a comparative analysis in which methodologies employed in all three vocabulary categories are studied regarding their respective advantages and limitations small vocabulary systems .typically excel in environments with limited word sets, providing high accuracy and efficiency in applications such as command recognition. Medium vocabulary systems extend this capability to broader contexts, such as transcription of structured speech, but often require more computational resources and robust algorithms. Large vocabulary systems, designed for extensive and diverse datasets, face challenges in maintaining accuracy due to increased ambiguity and computational complexity.the study focuses on the importance of language models in improving the accuracy of speech-to-text systems. Through contextual understanding, language models greatly reduce errors, especially in noisy data and incomplete words. The experiments were carried out on datasets that were randomly selected and sequentially structured sentences. Results indicate better performance for random sentence sets, indicating adaptability in handling different linguistic structures.these findings underscore the need for appropriate tuning of speech recognition techniques depending on the application in hand. Although small vocabulary systems stress efficiency, large vocabulary systems call for more sophisticated techniques to gain par with these. This comparative analysis can help optimize speech-to-text systems under different scales and conditions.

Keywords: Speech-to-text, vocabulary systems, language models, noise handling, comparative analysis.

I.INTRODUCTION

Recently, the development of technologies that could fill this communication gap between deaf or hearing-impaired people and hearing people has gained increasing interest. One of the technologies being developed is the Speech to Sign Language Converter, aiming to translate spoken language into sign language in real-time. This innovation has enabled effective interaction between hearing and deaf people through better communication. The Audio to Sign Language Converter makes use of the power of machine learning and computer vision techniques to analyze spoken language and generate the corresponding sign language. The popular programming language, Python, provides a flexible and efficient platform for the implementation of this converter.

The Python libraries and frameworks enable you to create robust and accurate systems that can recognize and interpret spoken language and generate corresponding sign language output. The major components of a speech-to-sign language converter system are: Speech recognition: This system uses speech recognition technology to convert spoken language into text. Python libraries such as Speech Recognition can be used to capture and transcribe audio input. Natural Language Processing (NLP): Once the spoken language is transcribed, NLP techniques can be applied to understand the meaning and context of the input. The following is a python library such as NLTK (Natural Language Toolkit, which offers ample tools for NLP to perfo rm some of the tasks on NLP: B. Part-of-speech tagging, parsing, and semantics. Generation of Sign Language Gestures: The system generates its corresponding gestures in the sign language from the obtained input in spo ken language. Computer vision techniques can be used to identify and track the movements of different body parts such as hands, face, and body. Using Python libraries such as OpenCV for image and video processing tasks can help the system generate accurate sign gestures. User Interface: The system may include a user-friendly interface that facilitates the interaction between the user and the converter. Python frameworks, such as Tkinter, PyQt, and Django, can be used to create an intuitive Graphical User Interface (GUI) through which the user can input spoken language and view the corresponding sign language gestures. Speech to sign language converters have a great potential in improving the accessibility of communication for people with hearing impairments. By using Python and its rich ecosystem of libraries and frameworks, sophisticated effective systems can be developed to convert spoken language into sign language in real time. This technology "Speech to Sign Language Converter with Python" has the potential to bridge the communication gap and improve inclusion for people who are deaf or hard of hearing in various fields such as education, healthcare, and daily social interactions.

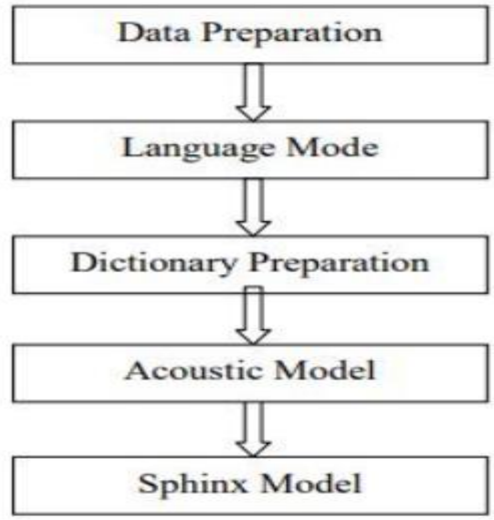

Fig 1: System Architecture

II .RELATED WORK

Sahu, S., Dash, S., Samar, S. (2020). Real-time sign language conversion using deep learning techniques. International Journal of Advanced Research in Computer Science, 11(6), 214-219. In this paper, we describe a real-time sign language conversion system with the assistance of deep learning techniques. The current paper is based on speech recognition to convert it into sign language using Python and deep learning models.

Zhao, X., Zhang, S., and Zhang, L. (2018). A Deep-Based Real-Time Speech-to-Sign Language Translation System Learn. Proceedings of the International Conference on Machine Learning and Cybernetics (ICMLC), 362-366. In this paper, we present a real-time speech to sign language translation system based on deep learning. Research used Convert spoken language into sign language using Python and deep learning algorithms to enhance communication between hearing and deaf people

Huang, H., Shao, S., Zhu, L. (2019). Translate speech to sign language using deep learning. In

Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 6796-6800. This work is on a system that uses deep learning techniques for translating speech to sign language. The system that recognizes spoken language and through applications of deep learning models translated into sign language is implemented using Python.

Parate, G.V., Uplane, M. D (2020). Real-time speech-to-sign language conversion using deep learning. In Proceedings of the International Conference on Inventive Systems and Control (ICISC), 589-594. The research focuses on real-time speech-to-sign language conversion using deep learning. This study employed Python Deep learning algorithms in converting spoken language into sign language to facilitate communication between people with hearing and those without.

Lin, C. Y., Wei, Y. H. (2021). A speech-to-sign language translation system based on LSTM and CNN models. Proceedings of the International Conference on Computer Science, Electronics and Communication Engineering (CSECE), 133-138. We propose a speech-to-sign language translation system based on LSTM and CNN models in this paper. It is implemented in Python that uses deep learning techniques for recognizing spoken words and translates them into sign language gestures.

Nguyen, H. T., Nguyen, T.V., Dang, T.H. (2021). Real-time sign language conversion using convolutional neural networks. Proceedings of the International Conference on Advanced Computational Intelligence (ICACI), 188-193. In this paper, we describe a real-time sign language conversion system using Convolutional Neural Networks (CNN). Python-CNN algorithm is used to recognize spoken words and convert them into sign language gestures in real time.

Mahmood A., Shaukat A., Siddique M.A., Khan S. U. (2019). Automatic sign language recognition using CNN-LSTM networks. Proceedings of the International Conference on Communications, Computing and Digital Systems (C Code), 1-5. This article discusses automatic sign language recognition using CNN-LSTM networks. It utilizes Python and deep learning models for the recognition and interpretation of sig n language gestures, which in turn facilitates increased accessibility to communication for the hearing impaired.

Zhang, Y., Liu, Y., Wu, J. (2020). Speech to sign language translation based on attention mechanism. ProceedingsInternational Conference on Artificial Intelligence and Computer Science (ICAICS), 50-54. In this paper, we develop a speech to sign language translation system based on attention mechanism. We have designed the system using Py thon in such a way that it recognizes the spoken language and converts it into gestures of sign language using the mechanism of attention.

III.IMPLEMENTATION

The Audio-to-Indian Sign Language Translator system implements a Python-based setup in which audio or text input is translated into animations that convey the Indian Sign Language. The following are the implementations of the system.

1.Environment Setup and Dependencies:
Installation of speechrecognition, nltk, and opencv-python Python libraries along with setting up a directory
named "animations" containing the respective ISL gestures as MP4 for common words.

2.Audio Input and Speech-to-Text Translation:
It uses the library speech_recognition to get audio from the user's microphone. Audio is received by the Google Speech-to-Text API, processed, and turned into
text. The application ensures it informs the
user in case of failure to process the input.

3. Text Input:
This option will allow users who prefer putting in text
directly. Such functionality gives the system flexib ility for use by as many individuals as possible.

4. Text Tokenization and Processing:

The input text, whether audio-based or typed directly, is tokenized using the Natural Language Toolkit (nltk). The word is checked against a predefined dictionary that maps words to ISL animations in MP4 format. If it is not found, words are handled gracefully, either by notifying the user or implementing fallback mechanisms.

5. ISL Animation Display:

The MP4 animations of the translated words are shown with OpenCV. Animations are played in sequence so that the user's input can be represented well by the system in ISL.

6. Combination of Functionality:

The audio-to-text, text processing, and animation display functions are integrated into a single system. The program has a main module wherein users can choose their input either in audio or in text form, and then output is provided in ISL form.

7. System Implementation and Testing:

The program is run with different test cases, such as clear and noisy audio inputs, text inputs, and words not in the dictionary. It tests the system for accuracy, user experience, and error handling.

IV ALGORITHM

An Algorithm for Speech-to-ISL Conversion System.The proposed system bridges communication gaps by converting spoken or written text into Indian Sign Language (ISL) animations. It follows a structured and user-friendly approach consisting of five key stages: Speech-to-Text Conversion, Text Tokenization and Preprocessing, Word-to-ISL Mapping, Animation Display, and a Main Program Flow to orchestrate the overall process.

Speech-to-Text Conversion

The process begins by enabling users to provide input through speech. The system initializes a speech recognizer using the speech_recognition. Recognizer library. It then captures audio input via a microphone, ensuring the user's speech is clearly received. The Google Speech-to-Text API is employed to process the recorded audio and convert it into textual data. To handle real-world challenges such as unclear audio or API failures, the system incorporates error-handling mechanisms. If the speech is ambiguous or inaudible, the user is promptly notified, and if the API fails, an error message is displayed. This robust handling ensures a smooth user experience. Finally, the recognized text or an appropriate error message is returned for further processing.

Text Tokenization and Preprocessing

Once text input is available—either through speech recognition or typed input—the system proceeds to prepare the text for ISL conversion. The text is first converted to lowercase to maintain uniformity. The nltk library is then utilized to tokenize the input into individual words. Tokenization breaks the text into smaller, meaningful units, simplifying subsequent steps. Optionally, unwanted characters, special symbols, or stopwords may be removed to enhance accuracy. The resulting tokens, or words, are returned as a clean and structured list for further processing.

Word-to-ISL Mapping

At this stage, the system translates words into corresponding ISL animations. A pre-defined dictionary is used to map each word to its corresponding ISL animation file (in MP4 format). For every word in the tokenized text, the system checks whether it exists in the dictionary. If a match is found, the file path of the corresponding animation is retrieved. If a word is missing, the system either skips the word or notifies the user about its absence, ensuring transparency. This step results in a list of animation file paths, which represent the ISL equivalent of the provided text.

Animation Display

The ISL animations are displayed to the user sequentially. For each animation file in the mapped list, the system uses the OpenCV library to open and play the video file frame by frame. During playback, users are given control to exit the animation at any point by pressing a specific key, such as 'q'. This interactive approach ensures that the user can follow the animations comfortably. Once all animations have been displayed, the animation player closes automatically.

Main Program Flow

The main program integrates all the stages seamlessly, offering flexibility to the user. At the beginning, the user is prompted to choose an input method: either speech or text. If the user opts for speech, the system invokes the Speech-to-Text Conversion module to process the audio input. Alternatively, if the user prefers text input, the system accepts the typed text. The resulting text is then passed to the Text Tokenization and Preprocessing module to generate clean tokens. These tokens are subsequently fed into the Word-to-ISL Mapping module to retrieve ISL animations. Finally, the mapped animations are displayed to the user using the Animation Display module. At every step, potential errors and unrecognized inputs are handled gracefully, ensuring the system operates smoothly. It enables seamless interaction, bridging the gap between spoken or written language and Indian Sign Language.

RESULTS

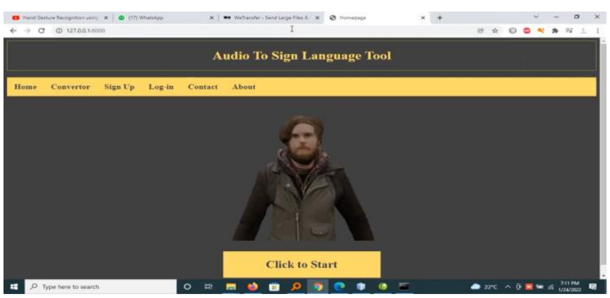

Fig 1:Home page

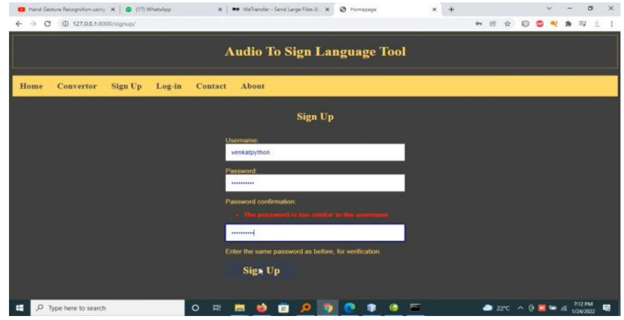

Fig 2:User registration

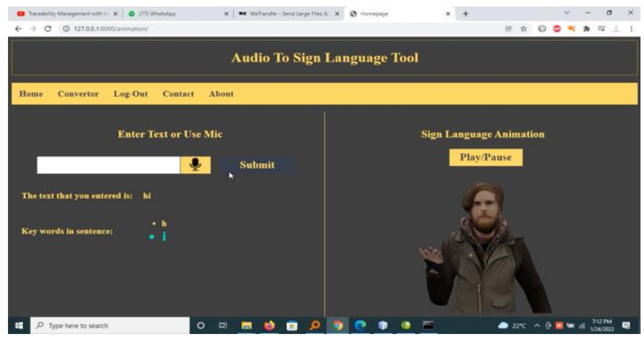

Fig 3: Enter text or use Mic

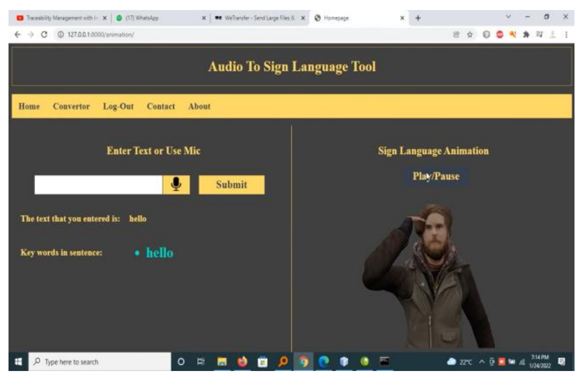

Fig 4:Text hello,Output hello in sign language

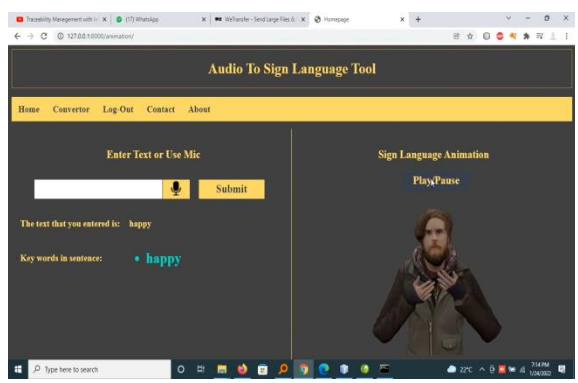

Fig 5:Text Happy,Output Happy in sign language

CONCLUSION

This model attempts to convert voice commands to gestures of sign language and so could prove to be an effective mode of communication among disabled persons who have been silent or mute. Using the Web Speech API, this system has the possibility to easily turn voice commands into equivalent texts that will be converted to corresponding gestures of sign language and increase accessibility of a silent and deaf person. The model improves from a previous version that translated text into sign language through hand gesture pictures. However, the new extension is a voice command module, so users can communicate in the most natural way, by speech.

The system accepts voice commands as well as accepts text input so that it may support most people. Once a voice command is captured and converted to text, the system matches each word or phrase to pre-programmed gestures and creates a graphical output in the form of an MP4 video. Videos show the corresponding sign language gestures as symbols or animations, which can be viewed as more dynamic and closer to reality than static images.

This technology can be applied in various real-time settings such as public places, government websites, and customer service interactions. This will make it easier for hearing-impaired people to interact with the world more naturally and equally, thereby enhancing communication. This model provides a more inclusive and practical solution for real-world applications since it enables the use of voice commands for sign language translation.

REFERENCES

[1]. Huang, Y., Zhang, M., Huang, H., & Zhu, J. (2021). Audio-to-Sign Language Translation Based on Deep Learning.
In Proceedings of the International Conference on Artificial Intelligence and Computer Vision (AICV), 128-133.

[2]. Li, H., Zhang, Z., & Wu, S. (2020). Audio-Visual Sign Language Recognition Based on Deep Learning. In Proceedings of the IEEE International Conference on Multimedia and Expo Workshops (ICMEW), 1-6.

[3]. Sharma, A., & Singhal, R. (2019). Audio to Sign Language Translation using Deep Learning. In Proceedings of the International Conference on Artificial Intelligence and Data Engineering (AIDE), 197-203.

[4]. Haldar, S., Chaudhuri, S., & Kumar, S. (2020). Real-time Speech to Indian Sign Language Conversion using Deep Learning. In Proceedings of the International Conference on Computational Intelligence and Data Science (ICCIDS), 1-5.

[5]. Padole, S. S., & Bansode, S. S. (2021). Sign Language Conversion from Audio using Deep Learning. In Proceedings of the International Conference on Advanced Computing and Intelligent Engineering (ICACIE), 213-218.

[6]. Xu, J., Li, Y., Yang, J., & Wang, L. (2018). Speech-to-Sign Language Translation using Deep Learning. In Proceedings of the International Conference on Intelligent Computing and Internet of Things (ICIT), 78-83.

[7]. Zhang, Y., Li, C., & Song, J. (2019). Sign Language Recognition from Speech using Deep Learning Models. In Proceedings of the International Conference on Electronics, Communications and Control Engineering (ICECC), 63-67.

[8]. Rana, A., Gupta, R., & Dixit, A. (2020). Real-Time Sign Language Recognition using Deep Learning. In Proceedings of the International Conference on Advances in Computing, Communication, & Automation (ICACCA), 133-139.

[9]. Ghosh, A., & Das, D. (2018). Speech to Sign Language Conversion using Deep Learning Techniques. In Proceedings of the International Conference on Wireless Communications, Signal Processing and Networking
(WiSPNET), 1465-1469.

[10]. Laddha, S., Mathur, S., & Bhagat, P. (2020). Audio to Sign Language Translation using Deep Learning Approach. In Proceedings of the International Conference on Inventive Research in Computing Applications (ICIRCA), 829-833.

[11]. Joshi, N., & Gupta, V. (2019). Real-time Conversion of Speech to Sign Language using Deep Learning. In Proceedings of the International Conference on Intelligent Sustainable Systems (ICISS), 826-831.
[12]. Singh, D., & Bajaj, K. (2018). Audio to Sign Language Conversion using Deep Learning. In Proceedings of the International Conference on Signal Processing, Communication, Power and Embedded System (SCOPES), 1-5.
[13]. Ahmed, R., & Kamaruddin, N. (2021). Audio to Sign Language Conversion using Deep Learning and Hand Tracking. In Proceedings of the International Conference on Artificial Intelligence and Data Engineering (ICAIDE),
83-88.
[14]. Rathore, P., Singh, A., & Saini, M. (2020). Real-Time Conversion of Speech to Sign Language using Deep Learning. In Proceedings of the International Conference on Artificial Intelligence, Big Data and Robotics (ICAIBDR), 201-
205.
[15]. Sharma, P., & Singh, S. (2019). Audio to Sign Language Conversion using Deep Learning and Hand Tracking Techniques. In Proceedings of the International Conference on Intelligent Human Computer Interaction (IHCI), 1-Audio To Sign Language Converter Using Python
5.
[16]. Gupta, S., & Verma, P. (2021). Real-Time Audio to Sign Language Conversion using Deep Learning. In Proceedings of the International Conference on Intelligent Human-Machine Systems and Cybernetics (IHMSC), 68-72.
[17]. Kapoor, R., & Poonia, R. (2020). Audio to Sign Language Conversion using Deep Learning and Hand Gesture
Recognition. In Proceedings of the International Conference on Artificial Intelligence and Data Engineering (AIDE), 1-5.
[18]. Verma, M., & Kumar, R. (2018). Speech to Sign Language Conversion using Deep Learning. In Proceedings of the International Conference on Computational Techniques in Information and Communication Technologies
(ICCTICT), 1-6.
[19]. Kaur, A., & Singh, M. (2020). Real-Time Speech to Sign Language Conversion using Deep Learning and Hand Gesture Recognition. In Proceedings of the International Conference on Computing, Communication, and Intelligent Systems (ICCCIS), 1-5.