A TOOL TO CONVERT AUDIO/TEXT TO SIGN LANGUAGE USING PYTHON LIBRARIES

DR M MADHUBALA¹, N V KRISHNA RAO², DR. Y MOHANA ROOPA³, D KALYANI⁴, M PRANAY⁵, CH KRISHNAM RAJU⁶

¹, ², ³, ⁴, ⁵, ⁶Department of Computer Science and Engineering, Institute of Aeronautical Engineering, Dundigal, Hyderabad-500043.

ABSTRACT:

Deaf people's mother tongue is sign language, which is a visual language. Unlike acoustically transmitted sound patterns, sign language employs body language and manual communication to express a person's thoughts. It is done by integrating hand forms, hand orientation and movement, and facial expressions all at the same time. It can be used by people who have difficulty in listening, people who can hear but cannot speak, and regular people to communicate with people who are deaf or hard of hearing. It is important for a deaf person's psychological, mental, and linguistic advancement to have access to a sign language. Deaf people's first language should be accepted, and their schooling should be conducted bilingually in sign language and written or spoken language. Deaf and hard-of-hearing people use Indian Sign Language to communicate by making various body gestures. There are various groups of deaf people all over the world, and their languages will be different as well. American Sign Language (ASL) is used in the United States; British Sign Language (BSL) is used in the United Kingdom; and Indian Sign Language (ISL) is used in India for transmitting feelings and communicating. Manual speech and body language (non-manual communication) are used to express emotions, concepts, and feelings in "Indian Sign Language (ISL)." One handed, two handed, and non-manual ISL signals can both be grouped into three groups. Manual signs, such as one-handed and two-handed signs, are made with the signer's hands to communicate information. By altering body posture and facial expressions, non-manual signs are produced. The website we are going to createmainly focuses on Manual signs and converts English text into Sign language, enabling hearing-impaired people in India to communicate with others.

Keywords: Sign Language, Natural Language Processing, Django, Asynchronous server gateway interface (asgi), Web server gateway interface (wsgi).

I. INTRODUCTION

To communicate with each other and with other deaf people, deaf people require sign language. Furthermore, some ethnic groups with very distinct phonologies (such as Plain Indians Sign Language and Plateau Sign Language) have used sign languages to communicate with other ethnic groups. The study of physical sounds in human speech is alluded as phonology. Sign language's phonology can be established. Phonemes are separate signs in a row of hand signs, rather than sounds. The following variables are taken into consideration:

1. **Hand shape** when rendering the sign is the first configuration.

2. **Hand Orientation:** This determines the direction in which the palm of the hand faces.

3. **Placement:** This refers to the location where the sign is over.

4. **Hand motion** when making the symbol.

5. **Line of interaction:** The part of the hand that creates contact with the body.

6. **Plane:** The sign is determined by the distance between the body and the object.
The fundamentals of sign languages have been presented in a concise manner. The Phonology section contains the core linguistic features that the scheme uses. The conditions that are taken into consideration are as follows:

1. **Location**: This is the location where the sign appears.

2. **Movement**: When making a symbol, the hand moves (straight, swaying, circularly).

3. **Plane**: The distance seen between body.

The parameters applied to these characteristics in relation of position, motion, and plane have been taken from the ASL dictionary's various simple signs.

Deafness and deaf people are as old as mankind, but the contact and education of deaf people was first recorded in the 16th century. In Spain, deaf children from wealthy families were put in the care of a monk who taught them how to write. It was important to speak in order to achieve money. The argument about oral vs. sign language, which has raged for decades, started with this case. The oral approach requires instructing or communicating with Deaf people using spoken words. In Germany, this method was refined to the level that it became known as the "German Method." The "French Method" gets its name from the fact that sign language was developed and used extensively in French deaf schools. In 1880, an effort was made to scrub Sign Language from the face of the earth. Hearing teachers and educationalists attended a conference in Milan (Italy) that issued a statement prohibiting further use of Sign Language in Deaf schools. Sign Language has developed into a hidden language. Outside of the school, Deaf students used Sign Language, but it became a living and universal language.

Four critical things need to be stated in the creation of a symbol. They are modes of hands, position, action and guidance. In Sign Language, finger spelling refers to how the alphabet's 26 letters are formed on the fingertips. Finger spelling is used to spell people's names, locations, ideas, and terms for which there are no signs or for which one has forgotten. Finger spelling is not the same as Sign Language. It's a code switch strategy in which you represent the written English word in space at the time you finger spell a term in English. Finger Spelling is restricted to individuals (deaf or hearing) who have been introduced to written English or some other spoken language.

**Software Details:**
1. Programming Language: Python3, SQL
2. Packages Used: NLP, python libraries (like Django)
3. Tools: Blender

**Hardware Details:**
1. Device (Mobile/Laptop)
2. Mic
3. Monitor to view output
4. Internet for voice recognition
5. Web browser

The Existing solutions: Humans are making clever inventions every year to assist themselves and others who are affected by any disability, as technology advances at a breakneck pace. We want to make it easier for deaf people to communicate, so we've created a sign interpreter that converts audio to sign language automatically. Sign language is the only means of communication for the deaf. Physically handicapped people use sign language to communicate their feelings to others. Because common people struggle to learn the particular sign language, communication becomes impossible. Because sign language consists of a variety of hand movements and gestures, achieving the right precision at a low cost has become a mammoth task. We now have physical devices and software that can transform audio to sign language, so we're improving the tool with Natural Language Processing. The word library can be enlarged to include the vast majority of frequently used English words. Using various NLP algorithms, speech to text conversion can be improved and text processing can be optimized.

Our objective is to comprehend the challenges that specially abled people face on a daily basis and to devise a solution that is a. cost-effective, b. adaptable by people, and c. simple to implement. Understanding the needs of the disabled community and finding a solution to them is crucial to making a difference. To increase the physical and mental health of people with disabilities, as well as their overall quality of life.

II. RELATED WORK

Deploying with ASGI and WSGI

Asynchronous Server Gateway Interface is an acronym for ASGI. It enhances the functionality of WSGI (Web Server Gateway Interface), which is a standardized method of communication between the server and web applications in most Python frameworks, including Django. Both ASGI and WSGI are specifications for providing a standard interface between Python web servers, applications, and frameworks. The only concern designers had when developing WSGI was to create a protocol that provides a common ground for web
development so that users can easily switch between multiple web frameworks without worrying about the specifics of how the new framework communicates with the server. And while WSGI did a good job of dealing with these concerns, when the relatively new protocols other than HTTP (HyperText Transfer Protocol), particularly WebSocket, began to gain popularity among web developers, WSGI failed to provide a method of creating applications that could handle these protocols.

Because WSGI applications can only accept requests from the server and return replies to the client/server, it is intrinsically suited to only handle HTTP protocol. WSGI applications are single, synchronous callable that take a request as input and return a response, allowing you to:

- Connections are short-lived, which is ideal for HTTP but not for long-polling HTTP or WebSocket, which have long-lived connections.
- Because requests in applications only have one route to follow, protocols with multiple incoming events (such as receiving WebSocket frames) cannot be handled by a WSGI application.

ASGI is separated into two sections, each with its own set of responsibilities:

1. A protocol server that suspends sockets and maps them into connections and event messages for each connection.
2. An application that runs within a protocol server is created only once per connection and treats event messages as they occur.

The server, like WSGI, hosts and runs the application within it, as well as passing incoming requests to it in a standardized format. Applications, unlike WSGI, are objects that accept events instead of just simple callable and must run as coroutines capable of handling asynchronous I/O operations. In contrast to WSGI, a relationship consists of two parts:

1. A connection scope is a representation of a protocol connection with a user that lasts for the period of the connection.
2. When anything occurs on that connection, events are sent to the application.

Applications are started by passing a connection scope, and then they run in an event loop, handling events and sending data back to the client in the form of events. A single incoming socket/connection is mapped to an application instance that lasts for the period of the connection, or maybe a little longer if any cleanup is required. A single asynchronous callable defines an ASGI application. It approves scope, which contained data about the user requests, send, which enables you to send events to the client, and receive, which allows you to receive events from the client.

The ASGI application can now accept both incoming and outgoing events as a matter of the reordering of the design of the model, removing the restriction of the WSGI application having a single path for incoming requests. Not only that, but the ASGI application can also run a background coroutine, allowing it to do more than just handle requests in the background.

In an ASGI application, every event you send or receive is a Python dict with a predefined format. ASGI applications can be easily swapped between different web servers thanks to these predefined event formats. These events still have a type root level key, which can be helpful in determining the structure of the event.
Where application is a WSGI application that takes two parameters: environ, which holds information about incoming requests, and start_response, which is a callable that returns the HTTP header. To make ASGI backward compatible with WSGI applications, we need to allow for it to run WSGI synchronous applications within an asynchronous coroutine. Additionally, ASGI receives incoming request information via scope, while WSGI accepts environment. As a result, we must map environ to scope.

III. PROCEDURE AND METHODOLOGY
A. **A2SL FOLDER:**

1. **ASGI.py and WSGI.py:**
   
   To make ASGI backward compatible with WSGI applications, we need to allow for it to run WSGI synchronous applications within an asynchronous coroutine. Additionally, ASGI receives incoming request information via scope, while WSGI accepts environment. As a result, we must map environment to scope. Asynchronous Server Gateway Interface is an acronym for ASGI. It enhances the functionality of WSGI (Web Server Gateway Interface), which is a standardized method of communication between the server and a web application in most Python frameworks, including Django. Both ASGI and WSGI are specifications for providing a standard interface between Python web servers, applications, and frameworks.

2. **Homepage view.py, URL’s and Settings:**
   
   The website view consists of “Home”, “about”, “Login”, “signup”, “contact” and “convertor”. To make these views possible, all the html files related to those terms are programmed in view.py and when you click on “login” in website it should open login page and settings are done accordingly.

B. **ASSETS FOLDER:**

   All the recorded and created sign language animations are stored in this folder. If the input given by the user matches with the name of the sign language animation that output is displayed to the user.

C. **TEMPLATES FOLDER:**

   Website Designing is done here. Every html file which are added in view.py has its own description. For example, home.html consists information related to homepage of the website and about.html consists of information related to the people who created the website. Similarly, Login.html consists of information related to login page.
D. **SQLite3 file:**
   The data related to username and password in the login page stored in a database file format sqlite3.

E. **Main.py:**
   This is the main program and we have to import `execute_from_command_line` from `django.core.management` and run this `main.py` executable python file in anaconda prompt. Prompt gives server information through URL as an output. Paste that URL in a browser and website is shown.

**After website creation is done, process goes in this way:**

![Diagram](image)

*Fig. 7: Schematic diagram for sign language conversion*

The methodology follows as input is given to the tool in the form of text/Audio. If input is given as audio, it gets converted into text using speech recognition tools in python based on Natural language processing.

A. **Audio to Text Conversion:**
   Audio input is taken using python PyAudiomodule.

   Conversion of audio to text using microphone

   Dependency parser is used for analyzing grammar of the sentence and obtaining relationship between words.

![Image](image)

*Fig. 8: Audio to text conversion*

B. **Splitting:**
   If the word is found send video to the user as output otherwise break the sentence into words and fetch the videos of that words and combine them into a single video and show the video to the user as output.
C. Searching:
Create animations for datasets using blender tool. The word library has been expanded to include most of the commonly used word in English Language as well as technical words and others which are available. Speech to text conversions can be made more accurate and text processing can be optimized using various NLP algorithms which are available. Search for the entered text/Audio in the database created. Create another database file through SQLite to store the usernames who gets signup/login into the website. Client and server communicates through Django modules (ASGI & WSGI)

D. Fetching Sign Language Animation:
After clicking on submit button, sign language animation is displayed.

IV. IMPLEMENTATION AND RESULTS

Output generation
The equivalent sign language representation of a given English text is used to generate output. The system's output would be a clip of ISL vocabulary. Each and every distinct word will have a video in the predefined database, and the output video will be a fused video of those words.
It is not easy for a machine to learn our language, but it is possible with the assistance of NLP. Here's how it works in practice:

The machine receives audio as feedback. That audio input is recorded by the machine. The audio is then converted to text and displayed on the screen by the machine. The NLP system breaks down the text into components, determines the context of the conversation and the person's purpose, and then chooses which command to execute based on the NLP results. Actually, NLP is the process of developing an algorithm that converts text into words and labels them based on their position and function in sentences.

When you speak “hello” and “Thank You” as input into the microphone, the following output pops up:
V. CONCLUSION AND FUTURE SCOPE

A sign language translator comes in handy in a variety of situations. Anyone should use this system to learn and communicate in sign language in schools, colleges, hospitals, universities, airports, and courts. It facilitates contact between those who have normal hearing and those who have difficulty hearing. The future work is to modify the Website UI which can be improved and new functionalities can be added. Various front-end options are available such as .net or android app, that can be used to make the system cross-platform and increase the availability of the system. Although it is well recognized that facial expressions communicate an essential part of sign language, this project did not focus on them. We are excited to continue the project by incorporating facial expressions into the system.

REFERENCES

2. Ankita Harikade#1, Sarika Namade#2, Shefali Patil#3, Anita Morey #4, Department of Information Technology, Usha Mittal Institute of Technology, Audio to Sign Language Translation for Deaf People, (IJET) Volume 9, Issue 10, April 2020