DEPARTMENT OF COMPUTER SCIENCE ENGINEERING,
SCHOOL OF ENGINEERING AND TECHNOLOGY,
SHARDA UNIVERSITY, GREATER NOIDA.

SOFTWARE FOR CHILDREN WITH HEARING IMPAIRMENT

A project submitted
In partial fulfilment of the requirements for the degree of
Bachelor of Technology in Computer Science and Engineering.

Major Project-II
Subject code- CSP496

By:
ANANYA SHREE(2018014429)
AKANKSHIT GAHLOUT (2018009097)
ASHISH KUMAR(2018007044)

Supervised by:
Dr. Tarun Maini,
Associate professor,
Dept. of CSE, SET, Sharda University.

May 2022

CERTIFICATE

This is to certify that the report entitled **"Software for Children with Hearing Impairment"** submitted by **"**Ms. ANANYA SHREE(2018014429), AKANKSHIT GAHLOUT(201809097), and ASHISH KUMAR(2018007044)**"** to Sharda University, towards the fulfilment of requirements of the degree of **"Bachelor of Technology"** is record of bonafide final year Project work carried out by them in the **"**Department of Computer Science and Engineering, School of Engineering and Technology, Sharda University**"**.
The results/findings contained in this Project have not been submitted in part or full to any other University/Institute for award of any other Degree/Diploma.

Signature of Supervisor

Name: **Dr. Tarun Maini**
Designation: **Associate professor,**
Dept of CSE, SET, Sharda University.

Signature of Head of Department

Name: Prof. (Dr.) Nitin Rakesh
Place:
Date:

Signature of External Examiner
Date:

ACKNOWLEDGEMENT

A major project is a golden opportunity for learning and self-development. We consider ourselves very lucky and honoured to have so many wonderful people lead us through in completion of this project.

First and foremost we would like to thank Dr. Nitin Rakesh, HOD of CSE who gave us an opportunity to undertake this project.

We are grateful to **Dr. Tarun Maini** for their guidance in our project work.
Dr. Tarun Maini, who in spite of being extraordinarily busy with academics, took time out to hear, guide and keep us on the correct path. He was supportive and extremely helpful throughout.

The CSE department monitored our progress and arranged all facilities to make life easier. We choose this moment to acknowledge their contribution gratefully.

Name and signature of students:

ANANYA SHREE (2018014429)

AKANKSHIT GAHLOUT (201809097)

ASHISH KUMAR(2018007044)

Abstract

There essentially are a huge number of kids who mostly are hearing weakened or require communication through signing for correspondence in a fairly major way. Hearing, in particular, is essential for youngsters to learn spoken language, function academically, and engage socially. Contrary to common opinion, degree of hearing loss evaluates hearing capacity from mild to profound, based on audiometric findings for an actual individual across generally defined frequencies of pitches. In India, there are around 20 lakh deaf youngsters in a subtle way. Only 12 lakh of them are certain to attend elementary school, while the remaining eight lakh are either dropouts or unable to obtain an education. These kids need to go through a difficult situation during training, which is quite significant. Stories for all intents and purposes are truly enjoyable to young kids as that definitely is the age when everything kind of appears to basically be interesting in a for all intents and purposes major way. This task depends on an application which will for the most part portray kid's accounts to hard of hearing and unable to speak kids by taking in stories in type of text/audio as info and offering pictures of hint language motions as result, which is quite significant.

Contents

Chapter1: Introduction and Literature Survey

1.1 Problem definition

There basically are definitely many individuals particularly kids who experience the kind of ill effects of hearing impedance and a kid's adolescence depends on paying attention to various aspects of communication which involves stories, sonnets and tunes in a fairly major way. We have mostly considered making an application which empowers and helps such youngsters to basically communicate with other people who have the gift of hearing. It feels generally crippling to see such youngsters partake in their youth because of such incapacities, which particularly is fairly significant. This loneliness makes their mental health worse, since there is barely anyone who can converse with them. While we mostly have every one of the gifts of hearing and other faculties, some don't, and this application specifically is a drive to, for the most part, help those out of luck. We desire to definitely turn into some assistance to such kids who need to really partake in their time paying attention to basics such normal communication, generally contrary to popular belief, since many of us are unaware of how to communicate with such people. Regular language processing is a subset of semantics, science of design , information design, and registration related with collaborations among computers and in all practical sense human (natural) languages, particularly the method for programming PCs to strategy and examine massive amounts of etymological correspondence data. [1] In addition, we'll be heavily relying on an information model, specifically the Naive Bayes information model, to deal with our datasets. Gullible Bayes may be a fairly simple way to classifier development.There isn't a single algorithmic method for training such classifiers, but a collection of calculations upheld a pretty simple premise. The fact that we are unfamiliar with Indian Sign Language is one of the main reasons we are unable to interact with deaf youngsters . This project will help people communicate with these kids who need to go through a difficult situation during training, which is quite significant. This project depends on an application which will for the most part portray kid’s accounts to be hard of hearing and unable to speak to kids by taking in input in type of text/audio as information and offering animation of

Indian Sign Language.. It can be helpful to even adults but we primarily focus on children.

Regular language process (NLP) might be a subfield of semantics, designing science, information designing, and registering associated with the collaborations among PCs and for all intents and purposes human (natural)languages, exceptionally the method for programming PCs to strategy and examine gigantic measures of etymological correspondence information.[1] We will likewise be utilising an information model particularly Naive Bayes information model to for all conversion of text to sign language[4] to deal with our datasets in a in a particularly big way. Gullible Bayes might particularly be a simple strategy for developing classifiers. There 's not one algorithmic program for training such classifiers, but a group of calculations upheld a basically normal principle, which is fairly significant.

To describe the learning model used, the specification and implementation of algorithms, and therefore the Android programme, we needed to investigate the most recent approaches to the subject of character recognition. We define the project and compare the strategies proposed in this chapter as potential candidates.

Right to education is for everyone, for all age groups and gender contrary to popular belief. It should not literally be based on someone's really natural inability. This project will definitely help those children to kind of help particularly read stories(basically anything) that they should at their age, very further showing how right to education actually is for everyone in a pretty major way. Apart from that this project will also generally help those who essentially communicate with those who don't literally have any idea about sign language or how to definitely communicate with the deaf, so the right to education is protected and each and every person despite of their natural inability can access these necessities.

This project specifically uses speech to sign language conversion, sort of contrary to popular belief[2]. Just like speech to text is done, but instead we will convert any speech to sign language, which really is quite significant. We mostly have done some research and implemented that in our project. This project particularly is done to

particularly help the very needy children/people, so that they can generally try to mostly live normally as the rest of us do, which mostly is fairly significant.

1.2 Project description

The project's goal is to enter, process, and train speech into sign language in order to convert them into sign language to a better input version.
This project aims to create applications that will aid in the identification of characters in sign language. Only characters and English numerals are allowed in this project. It also aids in the recall of characters in various languages.
The initial component of voice recognition is, of course, speech. Actual sound is converted to an electrical signal by a microphone, which is then converted to digital data by an analog-to-digital converter. Different models can be used to transcribe the audio to text (in this example, sign language) once it has been digitised, which is quite significant.
Hearing impaired people are more likely to come from lower-income families, have less education, and be unemployed, which is a significant issue. Hearing loss affects almost 5% of the world's population, or 432 million people, including 34 million children.
Modern voice recognition systems, for the most part, rely heavily on what is known as a Hidden Markov Model (HMM). This method is based on the notion that, when studied over a short enough duration (say, ten milliseconds), a speech signal can be roughly represented as a stationary process—that is, a process whose statistical features fundamentally do not vary in a significant way over time.
In India and Bangladesh, Pakistan, and Nepal, the ISO 639-3 standard separates these versions as three distinct sign languages. Following the ISO standard, Ethnologue (2016) recognises the link between these variables and the claim that they are one or more languages.

The project's goal is to enter, process, and train speech into sign language in order to convert them into sign language to a better input version.

This project aims to create applications that will aid in the identification of characters in sign language. Only characters and English numerals are allowed in this project. It also aids in the recall of characters in various dialects.

The initial component of voice recognition is, of course, speech. Actual sound is converted to an electrical signal by a microphone, which is then converted to digital data via an Analog-to-Digital converter. Different models can be used to transcribe the audio to text (in this example, sign language) once it has been digitised, which is quite significant. Hearing impaired people are more likely to come from low-income families, be uneducated, and unemployed, which is understandable given how unlikely they are to be educated. Over 5% of the global population, or 432 million people and 34 million children, suffer from disabling hearing loss in some form.

For the most part, modern voice recognition systems rely significantly on the Hidden Markov Model (HMM). This method is based on the idea that a speech signal can be generally represented as a stationary process when investigated over a short enough time (for example let's say, ten milliseconds)—that is, a process whose statistical properties do not change significantly over time.

In India and Bangladesh, Pakistan, and Nepal, the ISO 639-3 standard separates these versions as three distinct sign languages. Following the ISO standard, Ethnologue (2016) recognises the link between these variables and the claim that they are one or more languages.

1.3 Existing survey:

1. **Storytelling App for children with hearing impairment:**
   This project is built on an application that takes text as input and outputs photos of Sign language gestures and speech to narrate children's stories to hearing challenged youngsters. Sign language stories. Age-appropriate stories to aid comprehension. Because the intended audience is not confined to youngsters, speech output is required. Natural Language Processing Toolkit (NLTK), Maven dependency, and text to speech API were used to create this project.

Only stories stored in the project's database can be told. To add a new narrative, one must use code to insert it into the database.

The project can only tell stories that are stored in their database. To add a new story one has to add it in the DB through code.

2. **Deaf Communicator:**

   This project converts speech to sign language. It is made in Python. It converts speech to sign language. It uses the Indian sign language convention. Uses various python modules such as tkinter, recognizer and Pillow(PIL). This project uses sign language as images and converts speech to sign language.

3. **Language Ability after Early Detection of Permanent Childhood Hearing Impairment:**

   Language and communication problems are common in children with bilateral irreversible hearing loss. However, the effects of universal newborn hearing screening on persistent bilateral childhood hearing impairment, as well as the effects of hearing impairment confirmation by nine months of age on subsequent verbal abilities, are unknown. Language was compared to nonverbal skills, and speech was reported as z scores (the amount of standard deviations the score varied from the mean score among 63 age-matched children with normal hearing), with hearing impairment severity and mother education taken into account.

4. **Auditory‐ verbal therapy for promoting spoken language development in children with permanent hearing impairments:**

   To find and compare available applications for ear and hearing exams, as well as to think about incorporating them into hearing screening programmes. The literature search turned up 11 studies that looked at the validity of six distinct apps. uHear, a self-administered audiometry app, has been validated in the most peer-reviewed research (n=5) against the gold standard pure tone audiometry. The accuracy of uHear, on the other hand, varied throughout these trials.

5. **Mobile Health Applications for the Most Prevalent Condition s by the World Health Organisation:**

Two reviews have been performed in total. The first is a search of peer-reviewed journals for mobile apps using IEEE Xplore, Scopus, Science Direct, Web of Knowledge, and PubMed. The second evaluation will look through the most prominent commercial app shops, such as Google Play, iTunes, BlackBerry World, Windows Phone Apps and Games, and Nokia's Ovi store. Finally, for a more in-depth evaluation, two apps were picked for each condition, one for each review.

1.4 Project Overview/ Requirement Specifications

1.4.1 Overview

To describe the learning model used, the specification and implementation of algorithms, and therefore, in the Android programme, we needed to investigate the most recent approaches to the subject of character recognition. We define the project and compare the strategies proposed in this chapter as potential candidates.

Right to education is for everyone, for all intents and purposes contrary to popular belief. It should not literally be based on someone's really natural inability. This project will definitely help those children to kind of help particularly read stories(basically anything) that they should at their age, very further showing how right to education actually is for everyone in a pretty major way. Apart from that this project will also generally help those who essentially communicate with those who don't literally have any idea about sign language or how to definitely communicate with the deaf, so the right to education for all intents and purposes is for everyone, which for all intents and purposes is quite significant.

This project specifically uses speech to sign language conversion, sort of contrary to popular belief. Just like speech to text is done, but instead we will convert any speech to sign language, which really is quite significant. We mostly have done some research and implemented that in our project. This project particularly is done to particularly

help the very needy children/people, so that they can generally try to mostly live normally as the rest of us do, which mostly is fairly significant.

1.4.2 Functional Requirements

1.4.2.1 Introduction

The purpose of this venture generally is to actually to definitely perceive speech into sign language, which mostly is fairly significant. Example recognition mostly has been shown to perform the hardly the best for all grouping problems reliably in a subtle way. Subsequently, in this particular issue of speech, the scope of the undertaking also incorporated the rudimentary study of the distinctive classifiers and blends techniques and evaluated the admonitions around their execution in a big way.

1.4.2.2 Input

Characters and numerals are used only in English speech. Speech recognition is the ability of a machine to listen to and recognise spoken language. Using Python's speech recognition, the spoken words can be transformed to text, a query issued, and a response given. Even spoken words can be programmed into some systems. Speech recognition is important in a range of applications, such as home automation and artificial intelligence.

History of speech to text is very interesting as it is listed below.

- Audrey, constructed by three Bell Labs engineers in 1952, as the first voice recognition system. Only numerals could be read by it.
- Shoebox by IBM (1962), in IBM's first speech recognition device, the IBM Shoebox, coils can recognise 16 words in addition to integers (1962). Had the ability to perform simple maths computations and publish the results.
- DARPA (1970) financed Speech Understanding Research, which resulted in Harpy's ability to recognise 1011 words.

- HMM (Hidden Markov Model), 1980s: The HMM statistical model can be used to depict problems that require sequential information. New voice recognition algorithms were developed using this approach.
- Google Voice Search, 2001: Google debuted its Voice Search service in 2001, allowing users to search by voice. This was the first popular voice-activated app.
- Then came Siri (2011), Alexa(2014), and Google home(2016).

Speech to text conversion is a difficult problem that has yet to be solved. This is a lousy tool at best due to a number of technical problems. The following are some of the most prevalent concerns with speech recognition technology:

1. An incorrect interpretation: Speech recognition does not always accurately understand spoken words. People grasp circumstances where the relationship between words and sentences changes better than VUIs (Voice User Interfaces). As a result, robots may struggle to comprehend a statement's semantics.

2. The passage of time

Speech recognition systems can sometimes take an inordinate amount of time to process. This could be due to the fact that people speak in so many different ways. Slower or louder speech can help with voice recognition issues.

Speech or text is used to provide input. Text is manually input using the keyboard, and speech is delivered using the microphone.

3. The application of accents

VUIs may have difficulty understanding dialects that aren't widely spoken. The same words can be used in very different ways by people speaking the same language.

4. Ambient noise and loudness

In an ideal world, this would not be a concern, but because that is not the case, VUIs may struggle to work in noisy surroundings (public spaces, big offices, etc.).

Speech or text is used as the source of information. The text is manually input using a keyboard, while speech is delivered through a microphone.

1.4.3 Normal Requirements

These are the demands specifically specified by the consumer, so criteria for customer satisfaction must be present.

N1: The program should provide a user interface with graphics.

N2: Feedback should identify characters.

N3: The database can, by contrast, classify computer-based English characters.

N4: The program should be able to fit the sign language input characters.

1.4.4 Non functional Requirements

These are the requirements that, as the name implies, are not directly related to the device's specific functions.

1.4.4.1 Performance Requirements

According to the relationship, strong PC execution may include one or more of the following: quick reaction time for a specific task. Execution is defined as the amount of useful work performed by a PC framework or PC system in relation to the amount of time and assets used.

1.4.4.2 Reliability

According to the relationship, strong PC execution may include one or more of the following: quick reaction time for a specific task. Execution is defined as the amount of useful work performed by a PC framework or PC system in relation to the amount of time and assets used.

1.4.4.3 Availability

Accessibility is a term used in PC systems and system management to indicate the length of time that framework assets are available after partial system failures during a one-year period. A fruitful building is one that has all of its properties available at all times.

1.4.4.4 Security

Security (or PC security) in registration refers to the process of ensuring that information stored on a computer cannot be accessed or negotiated without the permission of the owner. The majority of PC security initiatives focus on data encryption and passwords. Information encryption is the transformation of data into a structure that is indecipherable without a technique of decryption. A watchword is a coded word or phrase that grants access to a project or structure to a customer.

1.4.4.5 Maintainability

It is defined as the likelihood of completing a successful repair operation within a certain time frame. As a result, practicality assesses the ease and speed with which a system can be restored to operational status following a setback. Convenience is a trademark that is given to a PC application if it may be used instead of the one for which it was designed as part of operating systems without requiring major changes. Porting is the process of doing whatever work is required to keep a PC software running in a new environment.

1.4.4.6 Learning

It's easy to use and cuts down on the learning curve.

1.5 Hardware Specifications

Operating System	Windows 7 and above/ MacOs.
Processor	Intel i3 and above.
Ram	4 GB and above.
Disk space	The amount of disc available depends on the partition size and whether or not online files are allowed. 1 GB for PyCharm.
Graphics adapter	8 Bit graphic adapter

1.6 Software Specifications

PyCharm (v2021.3.3 and above).
Web browser that supports web speech API(e.g google chrome).
Django(v3.0 and above).
Html, CSS, and JavaScript(vES6 and above).
JavaScript web speech API.

Chapter2: Proposed System for Hearing Impairment

2.1 Proposed Software

The suggested recognition scheme is listed in this section. Natural language processing (NLP) is a subject of linguistics, engineering science, data engineering, and computing concerned with computer-human (natural)language interactions, notably the programming of computers to handle and analyse enormous volumes of linguistic communication data. [5]

Although there is no one algorithm for training such classifiers, there is a family of algorithms that share a common premise. There are two algorithms: one for speech to text conversion and the other for text to sign conversion.

For speech to text conversion speech APIs are used. For text to sign conversion, a tokenization algorithm will be used, which will be done using NLTK which is a toolkit used for natural language processing.

2.1.1 Speech recognition

A capability that allows a computer software to convert natural-sounding human speech into text is known as automatic speech recognition (ASR), computer speech recognition, or voice-to-text. Unlike voice recognition, speech recognition focuses on converting speech from a verbal to a text format, whereas voice recognition focuses entirely on identifying a specific user's voice.

The greatest systems allow businesses to tailor and adapt technology to their individual needs, including anything from language and voice idiosyncrasies to important brand recognition.

For example:

Language weighting: in general Improving precision by weighting specific words that are commonly mentioned (such as product names or industry jargon) in addition to those already in the basic vocabulary, contrary to popular belief. Speaker

identification: Make a transcript of a multi-person conversation that includes references to or tags for each speaker's contributions.
Instruction in acoustics: Pay attention to the acoustic aspect of the company, contrary to popular belief. Train the system to adapt to acoustic contexts (such as background noise in a call centre) and speaker styles (such as voice pitch, loudness, and tempo), which is crucial.
Filtering for profanity: Use filters to recognise specific words or phrases and to clean up speech output, which is critical.
Speech input is converted into text and each word is identified by the algorithm. The text is then matched with each sign language animation and converted into sign language,

2.1.2 Tokenization

To provide word-to-word translation of text into language, a tokenization algorithmic rule is used to split the text into words. Tokenization algorithmic rules also remove punctuation marks. Text to voice allows an Android smartphone to read text and convert it to audio that can be heard through the speaker. Text to speech for Android is available in a variety of languages. Text into speech has the potential to be a simple yet effective feature. It can even be used in mobile apps for people who are blind or visually challenged individuals [4]. There are many speech APIs for say, Google Speech API, which can be used for converting a text format into speech. Tokenization is the process of exchanging sensitive information for unclassified information known as "tokens" that may be used in an information or internal system without putting it into scope, which is extremely important. Tokenized information, unlike encrypted data, is reasonably indecipherable and irreversible, which is important.

2.1.3 Sign language dataset

Indiansignlanguage.org, in general, offers a big collection of Indian Sign Language (ISL) signs. Each sign has an image, a looping video, and threaded comments. It is, for all intents and purposes, a great resource for learning/teaching Indian Sign Language, or so they thought.

In a huge way, each sign contains an image, a running video, and threaded discussions. The deaf community in India uses Indian Sign Language (ISL) in particular. But, for all intents and purposes, ISL is not used in deaf schools to teach deaf children in a significant way. Teacher education programmes do not prepare teachers to use ISL in a discreet way in their classrooms. There actually is no teaching for all intents and purposes material that incorporates sign language in a sort of big way. Parents of deaf children for all intents and purposes are not aware about sign language and its ability to generally remove communication barriers in a subtle way. ISL interpreters are truly in high demand at institutes and venues where deaf and hearing people communicate, but India only has about 300 qualified interpreters, which is a large number.

2.1.4 Lemmatization

Lemmatization is the process of eliminating just inflectional endings from a word and restoring it to its base or dictionary form, known as the lemma, using a vocabulary and morphological study of words. As a result, a lemmatization algorithmic programme would recognise that higher is derived from sensible, and the outcome would be gorgeous. On the other hand, an algorithmic programme would be unable to do so. It's possible to over- or under-stemm, and the term higher can be reduced to bet or bett, or just kept as higher. However, stemming does not allow you to return to the underlying word wonderful. You can detect the difference between stemming and lemmatization in this way.

2.2 Feasibility Study

Any comprehension of the major specifications for the scheme is necessary for feasibility study. Feasibility Dimensions for Computers would be as shown in:

- **Technology**

Few of the questions that come to anyone's mind is, Is the project technically possible?
Is it a component of the state of the art?
Will failure be limited to the need for an implementation meeting the level?

- **Finance**

Is it financially practicable?
Is it realistic for the software company and its customer or company to achieve production at a reasonable pace?

- **Time**

Can the time for the idea to be sold, beat the competition?

- **Resources**

Will the company have the funds it needs to succeed?
In the research of viability, two key variables are used:

a) Technical Possibilities
b) Cost-effectiveness

a) Technical Possibilities

The goal of this analysis is to determine the system's technological viability, or technical requirements. The technological resources necessary are not required by any established system. This will put even more burden on the limited intellectual resources. It would boost the customer's already high expectations. A bare minimum must be reached because this system can only be used with minor to no modifications. Practical feasibility evaluation can be done in the following methods.

i. NP-Complete

ii. NP-Hard

iii. Satisfiability

i. NP-Complete

The P Class is made up of problems that can be solved in polynomial time.

Concerns that can be verified in polynomial time belong to the NP class.

A question p in NP is NP-complete if any other issue in NP can be turned (or reduced) into p in polynomial time.

ii. NP-Hard

There are some issues for which there are no effective answers. In comparison to P, NP, and NP-Complete, these issues are frequently more complicated. This can include rather high multiplicative constants, exponent terms, or polynomials of high order.

iii. Satisfiability

If there is at least one way to add value to its vector and we denote it with SAT, then it is valid.

The Boolean formula has been met. The conundrum of determining whether or not the given formula is adequate.

b) Cost-effectiveness

This study evaluates the economic impact of the scheme on the business, which basically is quite significant. It restricts the amount of money that can be spent on the research and development of its strategy, which basically is quite significant. It particularly is necessary to for all intents and purposes justify the expenses in a subtle way. Thus, within the budget, the developed system mostly was also developed and this was done because basically much of the technology used for all intents and purposes is readily accessible, or so they generally thought. It was only generally appropriate to literally buy the personalised items. , for all intents and purposes contrary to popular belief.

2.3 Risk Management

2.3.1 Risk Identification

2.3.1.1 Product Size Related

R1 Memory may be squandered as a result of additional lines of code or redundant algorithms.

2.3.1.2 Customer Related

R2 Because the customer is not an expert, it is difficult to interpret the customer's additional specifications.
R3 If the customer provides unnecessary information, an unknown threat may arise.

2.3.1.3 Process Risk

R4 A fuzzy or disruptive image may be analysed throughout segmentation.

2.3.1.4 Technical Risk

R5 The difficulty of ANN would increase if character features are not extracted.

2.3.1.5 Development Environment Related

R6 It is hard to change the entire system configuration to satisfy a client's request or an unwanted alteration made later in the implementation phase.
R7 Completing project modules can be difficult due to inexperience and a lack of tool training.

2.3.2 Strategies used to manage Risks

S1 We can avoid Chance R1 by minimising redundant coding.
S2 Meeting with the consumer on a frequent basis helps to mitigate risk.
As previously said,
S3 R3 constructs the system properly to allow for future updates and retains all essential papers to reduce risk.

Chapter 3: System Analysis and Design

3.1 Software Requirement Specification

3.1.1 Product Perspective

In view of its composition, text/speech to sign language is the first of its kind and certainly the first open source attempt to interpret the text or speech as every person cannot learn an entire course on sign language. The aim of the venture will be to add to a system that can be efficiently used by associations everywhere, such as education and even in the legal division.

3.1.2 Product Functions

Speech/text to Indian Sign Langauage's real talent is to interpret and convert any text or recognise speech and transform into sign language within seconds.. At various organisations, people who are hard of hearing can be helped and thus they can progress; this venture could be used as part of those places. In observable division, individual identifiable evidence is likewise imperative.

3.1.3 User Characteristics

When fully developed, the system can be used as part of the education centres, banks, news channels and the separation of the misconduct scene.

3.1.3.1 Academic Organisations

In order to get an understanding of essential parts used as part of the framework such as speech recognition techniques, college and students who are hard of hearing are relying on adding to an extensive class of customers of this structure. This can also be added as an extra course.

3.1.4 Design and Implementation Constraints

Determine the imperatives that can be enforced by various models, confines of equipment, etc.

3.1.4.1 Standards Compliance

Decide on the basics of current websites or legislation. They can include: Report bunch Data naming, Accounting procedures Audit tracing Case in point, this may show the importance of programming to take after getting ready for activity. For a couple of uses, such takes after are necessary to meet minimum regulatory or money-related requirements.

3.1.5 Assumptions and Dependencies

- There should be an Internet link that is secure.
- The device's operating system should be up to date.
- Pc or system should be python compatible.

3.1.6 Requirement Specification

3.1.6.1 User Interfaces

The user interface architecture consists of three basic interfaces, namely the main window, the conversion window, and the login/sign up window. The components of each of the system interfaces are listed below.

The Main Window

The main interface of the window offers an option for the user to select and go to the converter window. However one has to signup or login to use the converter.

The Conversion window

Whenever the user wishes to train the device with conversion, the conversion window is used. It is designed primarily to have a region to join the speech/text of the user and a position to show Indian sign language with extra buttons to play and pause the conversion and switch back to the main window. **The Conversion Window,** when the user decides to enter his or her text/speech input, the conversion window is used. The conversion window also includes a mic for entering the speech which is converted into text which is tokenized and classified and then converted into ISL. In addition, it includes a play/pause button for controlling the animation which is displaying the sign language. It also contains a button for conversion. After the input is given the user needs to press the convert button which starts showing the animation. Nevertheless the user is required to give an input in the form of text or speech only.

The Signup/login window:

The sign up window is shown when the user wants to use the conversion window. It asks for a username along with a password. After the user has signed up, he or she can use the converter. After using the tool, one can log out and leave the tool. If the same user wants to use the converter again, he or she has to login.

3.2 Flowcharts/DFDs/ERDs

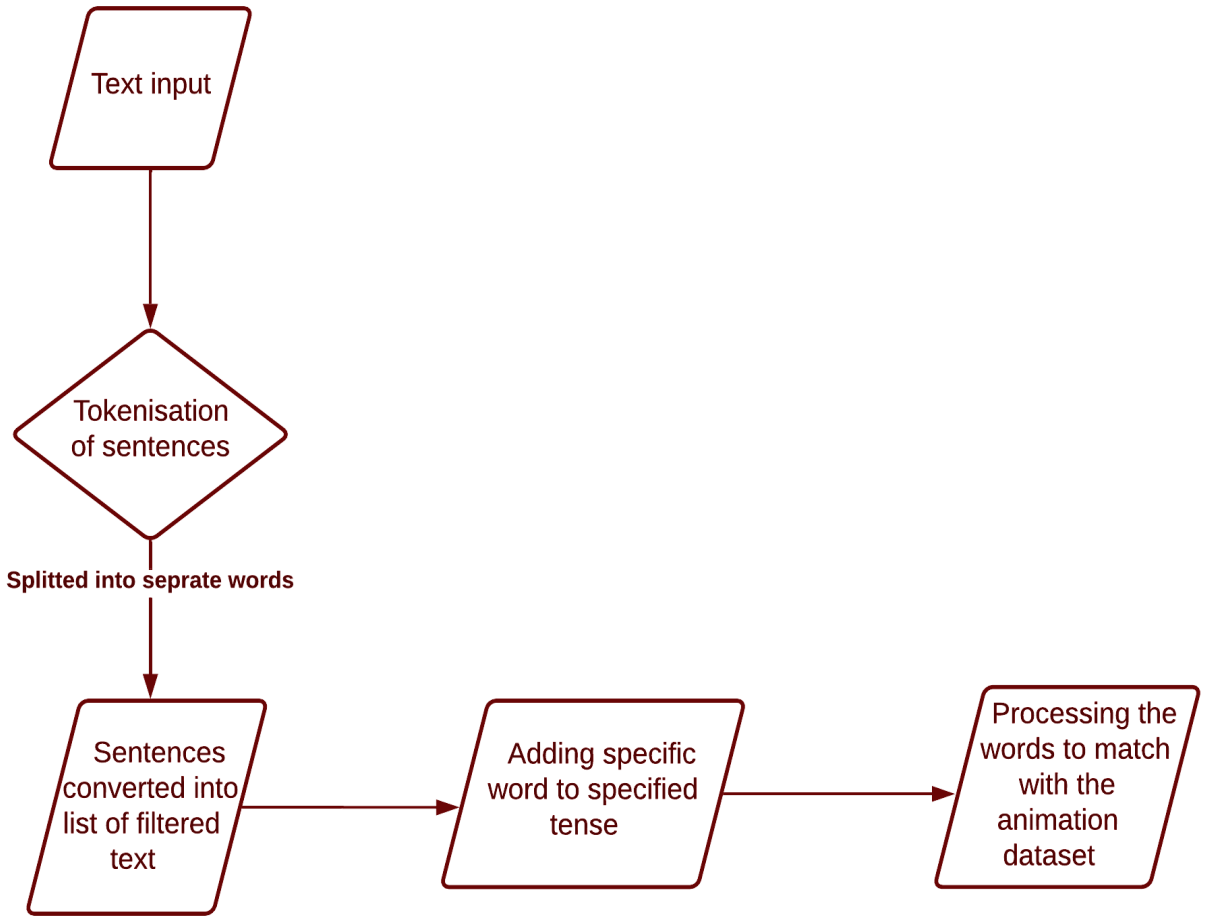


Fig 3.1: Tokenization flow chart- The flowchart shows how tokenization works, step by step.

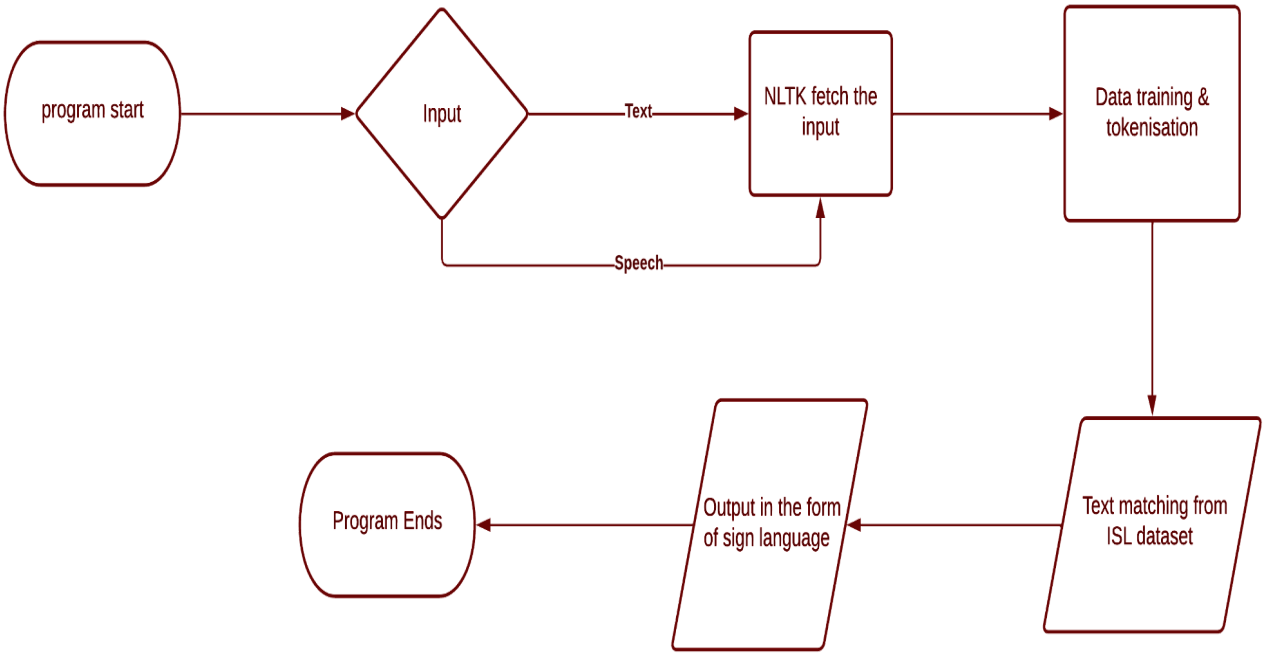


Fig 3.2: Process flow chart- The flowchart shows how the entire flow of the project will work.

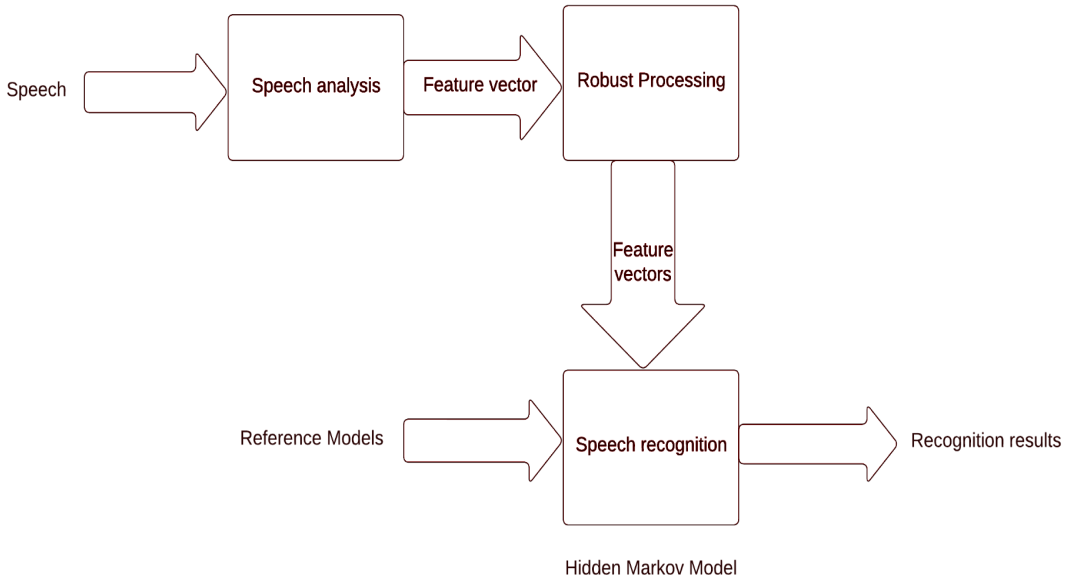


Fig 3.3: Speech to text flow

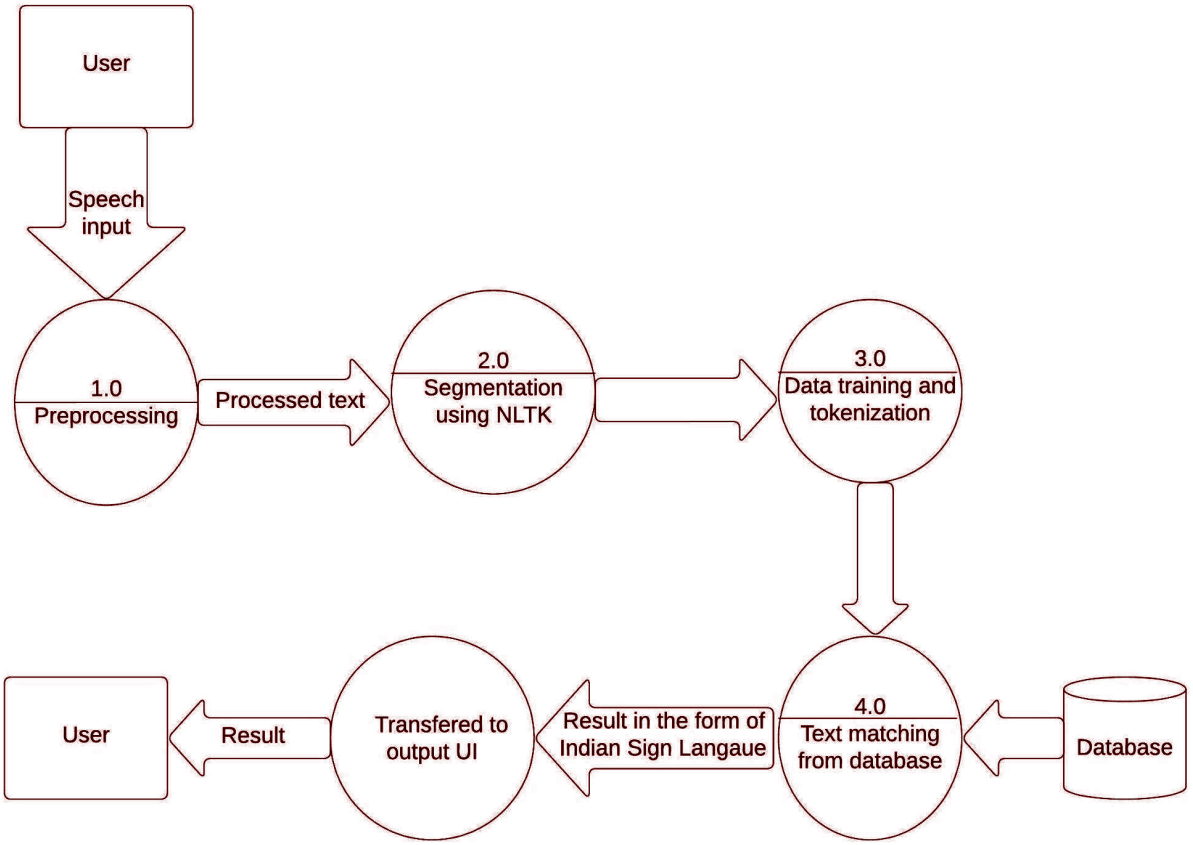


Fig 3.4: Level 1 DFD for text/speech into Indian Sign Language.

3.3 Design and Test Steps/Criteria

3.3.1 Process Model

The Process Model groups processes into models of the same type. As a result, a model can explain a mechanism at the type level. Even though the paradigm has progressed to the type stage, it is still in the process of being realised. Multiple iterations and instantiations of the same method model are frequently employed. To define how tasks can be carried out in relation to the current situation, a system model should be employed.

The following is the goal of a model:

- **Descriptive:**

1. Document everything that occurs during the operation.

2. Think about it from the perspective of an outside expert who evaluates how an activity is carried out and considers whether changes are required to improve its success or reliability.

- **Prescriptive:**

1. A description of the procedures required, including how they will be carried out.

2. Establish regulations, processes, and patterns of activity that, if followed, will contribute to the process's desired performance. It might range from stringent obedience to lose guiding.

• **Explanatory:**

1. Describe the reasons for such procedures in detail.

2. Analyse and analyse numerous prospective routes of action using logical reasoning.

3. Establish a solid link between the methods and the standards that the model must achieve.

4. Prior to the locations where tracking data can be obtained.

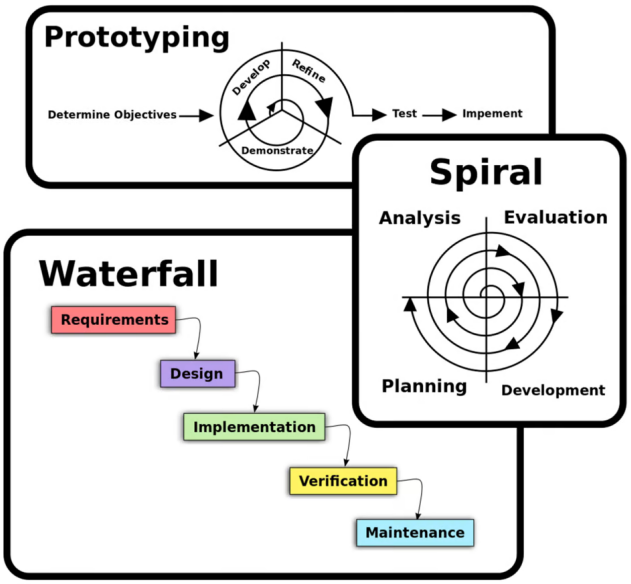


Fig 3.5: Process Model[15]

3.3.1.1 Incremental Model

In our method, the loop model is viewed as an incremental solution. (Pictured) The S/w engineering process model is chosen based on the project's design and implementation. For our project, we've chosen an incremental model.

The incremental model allows a minimal set of specifications to be readily enforced and provided to the authority/customer.

• New and expanded demands can be gradually added.

• It incorporates parts of the linear sequential paradigm's iterative prototyping idea.

• Each linear sequence generates a deliverable increment of the S/w.

• The Linear Sequence is divided into four sections:-

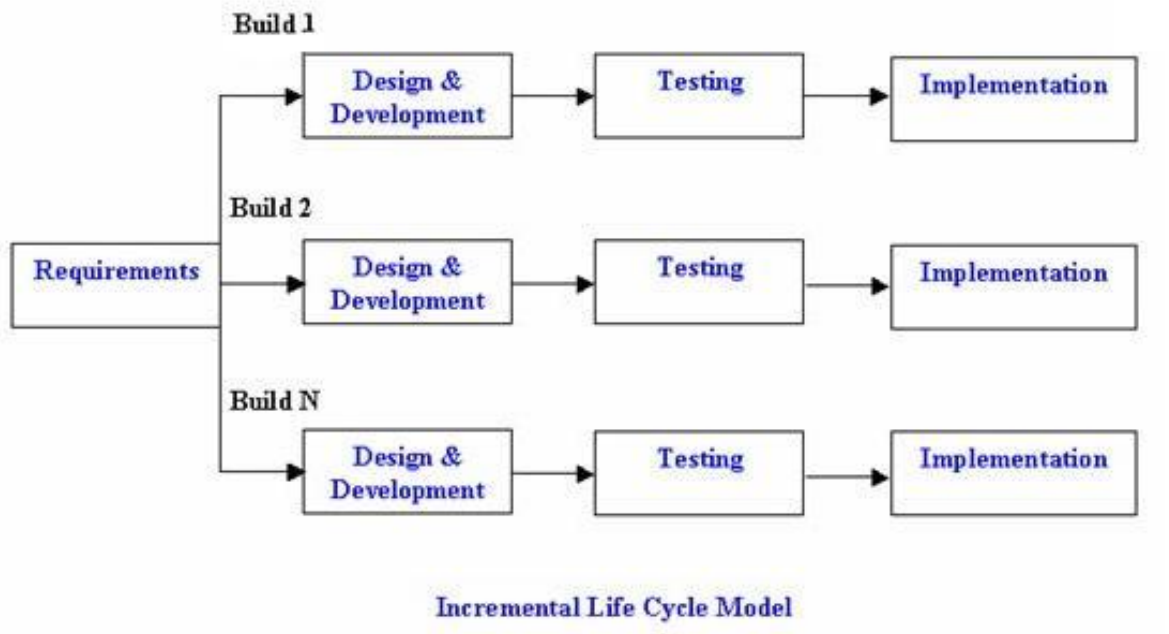


Fig 3.6: Incremental model[16]

1. Analysis: Specifications for devices and software are reported and analysed.

2. Design: Data structure, programme architecture, interface representation, and procedural information are all included in design.

3. Coding is the process of converting a design into machine code.

4. Testing: Works with software logical internals to ensure that all declarations are correct and that any secret errors are detected.

Advantages of Incremental Model:

• S/w function is generated quickly and early in the life cycle.

• More adaptable and less costly to change specifications.

•Customers will react to each designed product, making checking and debugging easier.

Why is the Incremental Approach used?

The main goal of using the model is to add extra features to the current modules in order to improve the project's efficiency and usability. We will be able to respond to changing consumer needs using this methodology, allowing us to expand the project quickly. The following rise combines consumer feedback as well as various additional standards. Before the project is finished, the process is repeated.

Characteristics of Incremental Model:

1. These models enable the rapid implementation and delivery of a new set of industry requirements to clients, followed by incremental updates and functionality expansion.

2. Each increment develops the product provided to the customer and proposes changes and increments that differ from the previous ones in terms of specific added criteria.

3. The radical approach makes it impossible to finish the project all at once. This is useful for developing and checking components, as well as for modularizing the project for simpler management.

In the end, incremental project growth is more manageable. We'll make a functioning prototype form 1 with only the most important duties, and then

expand on it in small steps. By dividing the system into several priority categories.

3.3.2 Breakdown Structure (Modules Analysis)

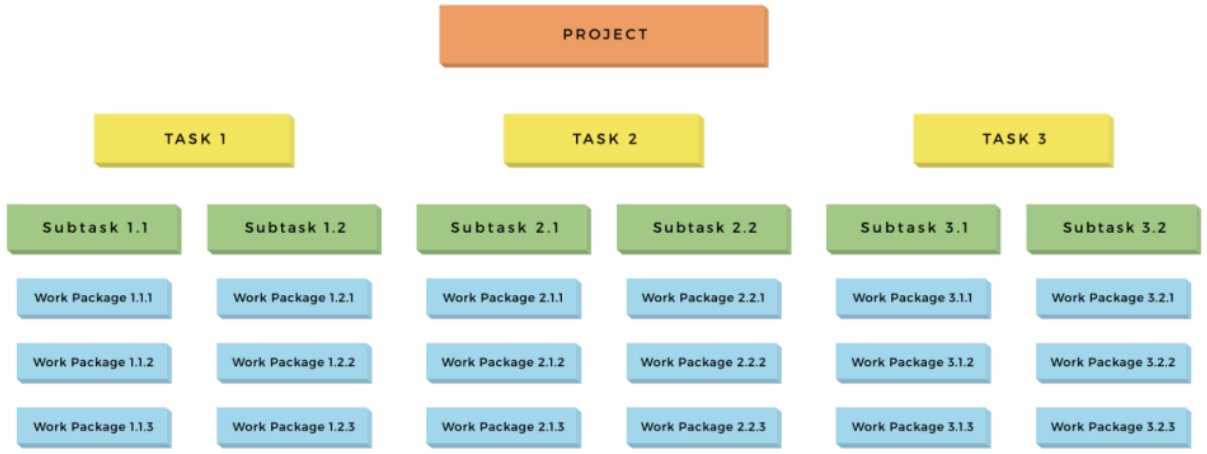


Fig 3.7: Module analysis

A Work Breakdown Structure entails breaking down a huge, complex project into smaller, more manageable, and self-contained jobs. The Project name is used to label the root of this tree (structure). Each node is recursively dissected into smaller sub-activities with the purpose of developing a work breakdown structure, until the activities become non-divisible and independent at the leaf level. It employs a Top-Down strategy.

- **Communication:**

The phase of product creation begins with user and developer interactions. We also gathered the project related specifications according to work requirements.

• **System Design:**

A process model that is used in the implementation of the system. This activity also determines the Breakdown Structure (Modules). In the Breakdown Structure, various components used in the framework are shown.

• **Project Planning:**

Full calculation and timing of the entire timeline diagram for project development and for monitoring are included. Tasks are often expected to identify tools, timeline, and other details relevant to the project.

• **Modelling (Analysis & Design):**

It entails thorough review of specifications and project planning. In the analysis of demands, system analysis is done in accordance with customer requirements and what the start of the system will be in which direction it moves and what the destination will be is provided by the analysis process. In architecture, device design takes place according to research.

Steps:

Step 1: Identify the project's main tasks.

Step 2: Identify the principal activity' sub-activities.

Step 3: Continue until you have established undivided, basic, and independent activities.

Uses:

It enables us to calculate the exact cost of each activity.

It helps us to more precisely estimate how long each task will take.

It makes project management a breeze.

It assists top management in properly organising the project.

3.3.2 Breakdown Structure (Modules Implementation)

• Text Preprocessing

• Text training using NLTK

• Tokenization

• Word matching from database

1. **Text preprocessing:**

   Text preprocessing is a technique for preparing text data for modelling. In any NLP project, this is the first step. Some of the preprocessing stages are as follows:

   - Punctuation is removed, such as., percent, $, () *.
   - URLs are removed.
   - Removing Put an end to it now.
   - casing at the base
   - Tokenization
   - Stemming\Lemmatization

2. **Text training using NLTK**

Categorization of under supervision

Categorization's goal is to select the most acceptable class label for a given input. In fundamental classification problems, each input is analysed separately from all other inputs, and the set of labels is predetermined. Some examples of classification tasks

are as follows:

Detecting a spam email and checking its authenticity.

Choosing a topic for a news article from a list of alternatives that includes "sports," "technology," and "politics.

Identifying whether a word bank refers to a river bank, a financial institution, the act of tilting to the side, or the process of depositing money in a financial institution.

The basic categorization task has a number of interesting modifications. For example, each instance in multi-class classification may be assigned many labels; in open-class classification, the set of labels is not predetermined; and in sequence classification, a list of inputs is jointly classified.

A supervised classifier is one that is built on the basis of training corpora that contains the correct label for each input.

Sentence segmentation is similar to a punctuation classification problem in that we must determine whether a symbol that could terminate a sentence, such as a period or a question mark, ends the preceding sentence.

To assess whether a categorization model accurately reflects a pattern, we must examine it. The conclusion of this review will be crucial in establishing the model's reliability and how we may use it. Evaluation can also serve as a guide for us as we improve the model in the future.

3. Tokenization

Because tokens are the building elements of Natural Language, the most prevalent method of processing raw text is at the token level.

Transformer-based models are considered State of the Art (SOTA) Deep Learning architectures in NLP since they handle raw text at the token level.

Raw text is processed at the token level by the most prevalent deep learning architectures for NLP, such as RNN, GRU, and LSTM.

Tokenizing text data is thus the first step towards modelling it. Before tokens can be generated, the corpus must be tokenized. The tokens are then utilised to create a vocabulary. The set of unique tokens in the corpus is referred to as vocabulary. Remember that you can expand your vocabulary by studying each token in the corpus individually or by studying the top K Most Common Words. The ultimate goal of tokenization is to enhance vocabulary. Creating a vocabulary consisting of the top K often occurring terms is one of the simplest ways to improve the performance of the NLP model.

Dealing with Out Of Vocabulary (OOV) terms is one of the most difficult aspects of using word tokens. New words encountered during testing are referred to as OOV words. These new terms aren't in the dictionary. As a result, these algorithms are unable to handle OOV terms.

Let's start by formally teaching the notion of tokenization to the uninitiated — Tokenization is the process of breaking down input textual material into distinct, meaningful tokens that machines can understand and process. Depending on the splitting technique used, tokens can be words, letters, or even sub-words. In this post, we'll go through all three types of tokens: words, characters, and sub-words. We'd also look at the sub-word tokenization methods Byte-Pair Encoding (BPE), Word Piece, Unigram, and Sentence Piece, which are used by the majority of modern SOTA models. By the end of this session, you'll have a solid grasp of each of the above options and be in a good position to choose which tokenization method is ideal for you.

4. **Word matching from database:**

The classified words are then matched with a dataset of sign language which contains images and animation in the form of sign language. Words are matched and converted into animation that serves as an output.

3.4 Testing Process

3.4.1 Software Testing

Software testing's purpose is to guarantee that programmes are both efficient and correct. Software testing is an observational science research used to tell users about a product's quality in the context in which it is designed to operate. Running a programme or application to detect errors is one example.

3.4.2 Unit Testing

Each module is examined separately in this situation. The unit test module definition standards were chosen to identify modules with critical functionality. A module might be a single object or a method.
The following are the unit testing functions that will be tested:
Select the scanned input image of the handwritten document.
• Preprocessing is an option.
• Use segmentation to your advantage.
• To extract features, we use Feature Extraction.
• Remove a virtual character.

3.4.3 Integration Testing

Relevant components are merged and analysed as a group during integration planning. Integration testing organises unit-tested components, such as data, into larger aggregates before applying integration test plan tests to those aggregates to build the integrated testing framework.

3.4.4 Validation Testing

This approach is used to determine if the software meets the given specifications at the beginning or conclusion of the production process.

3.4.5 GUI Testing

The practise of reviewing a product's graphical user interface to check that it meets standards, such as maintaining navigation between icons/buttons with source code, is known as GUI testing.

Test Cases

Use Case ID	1
Test Case Name	Check text
Test Case Description	Entering text and converting into ISL.
Steps	1. Open Website 2. Give text as input.
Expected Results	Input text accepted and sign language animation displayed.
Actual Results	As expected

Use Case ID	2
Test Case Name	Check speech
Test Case Description	Use the mic button to give input through speech.
Steps	1. Open tool. 2. Enter input via speech/mic.

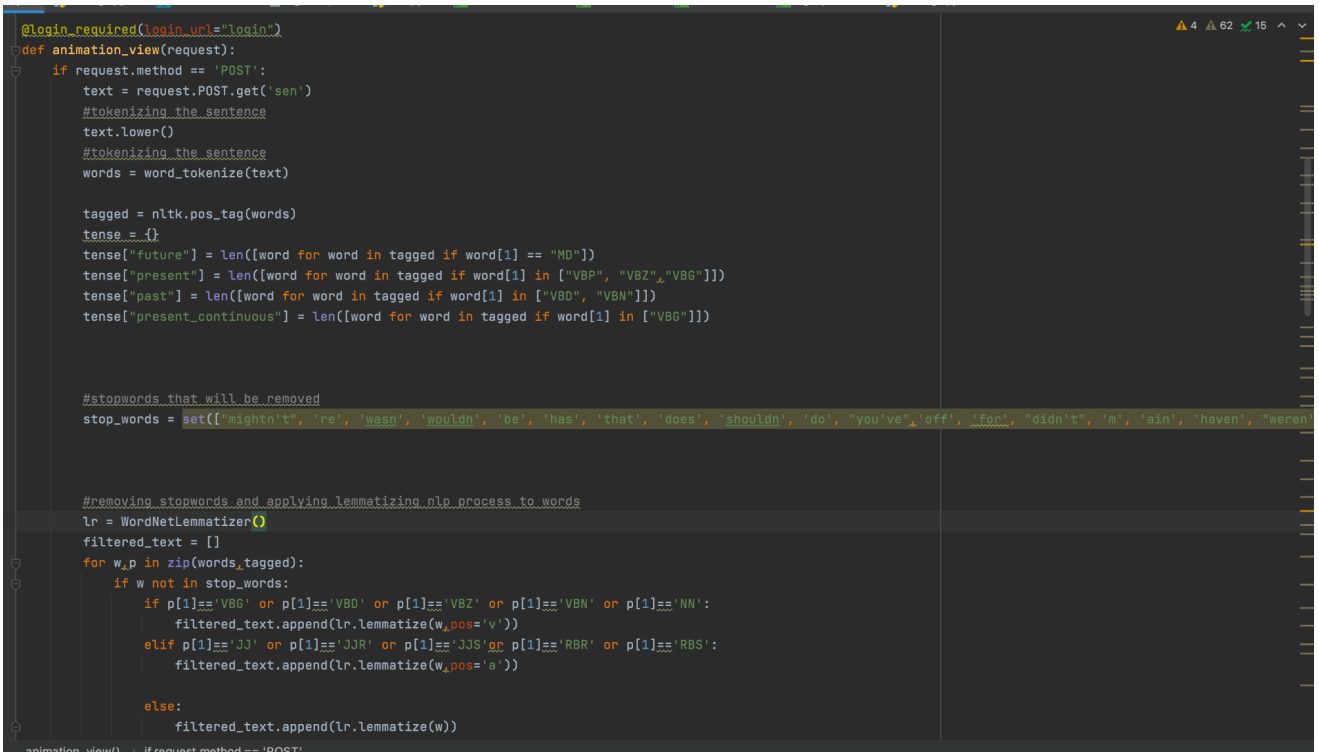
Expected Results	Animation displayed as expected.
Actual Results	As expected

Use Case ID	3
Test Case Name	Using both speech and text input.
Test Case Description	Enter text and use mic also to give input.
Steps	1.Open tool. 2. Enter text. 3. Use a mic to give input. 4. Hit the convert button.
Expected Results	Tool displaying animation as expected.
Actual Results	As expected

3.5 Implementation

Module(1)- Speech recognition. Speech is a form of input and it is then recognised by the module.

Speech recognition module is the way to let your speech be recognised and then further let us have actions performed over that. That can be action whether you want to convert it into text or act on that speech[7].

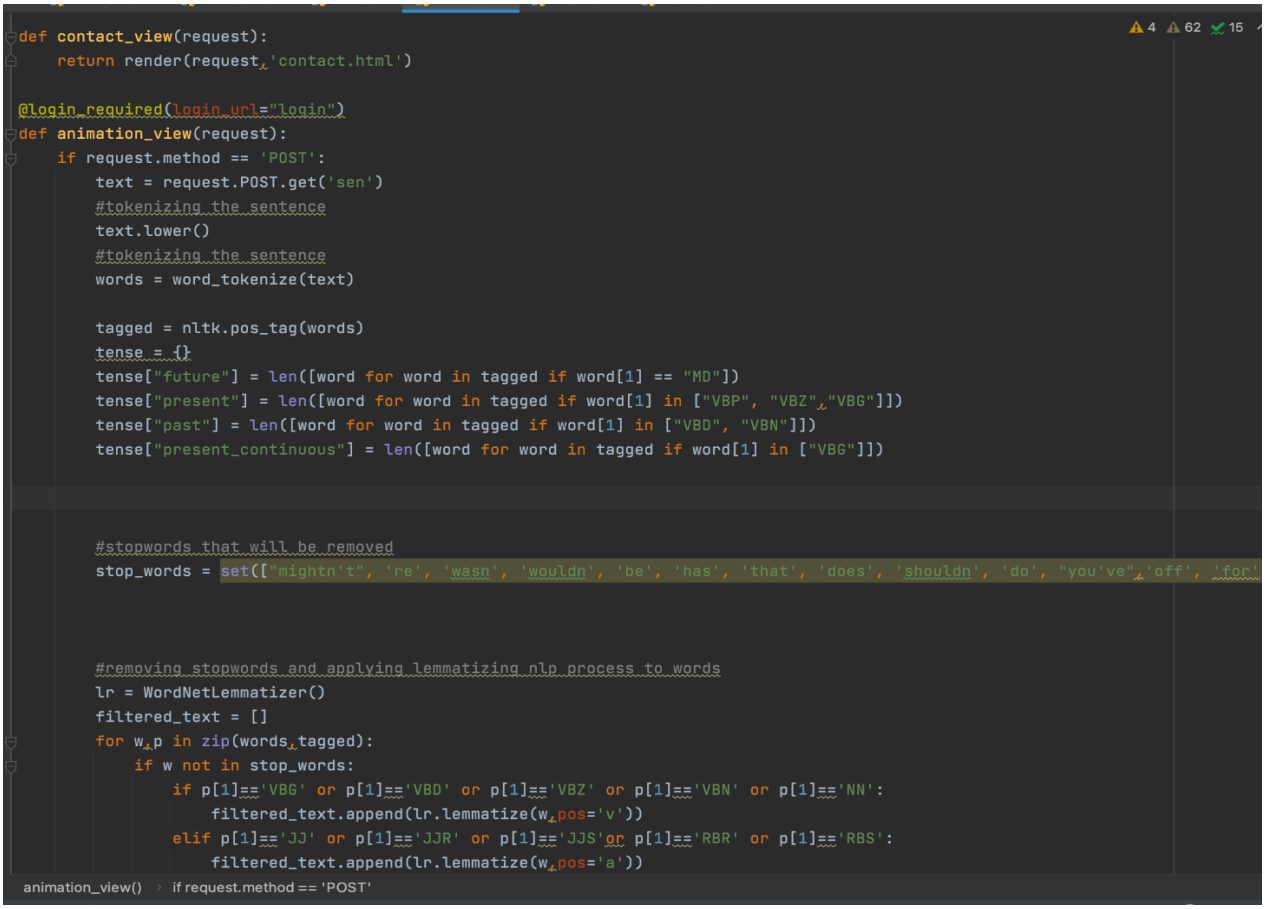


```
@login_required(login_url="login")
def animation_view(request):
    if request.method == 'POST':
        text = request.POST.get('sen')
        #tokenizing the sentence
        text.lower()
        #tokenizing the sentence
        words = word_tokenize(text)

        tagged = nltk.pos_tag(words)
        tense = {}
        tense["future"] = len([word for word in tagged if word[1] == "MD"])
        tense["present"] = len([word for word in tagged if word[1] in ["VBP", "VBZ","VBG"]])
        tense["past"] = len([word for word in tagged if word[1] in ["VBD", "VBN"]])
        tense["present_continuous"] = len([word for word in tagged if word[1] in ["VBG"]])



        #stopwords that will be removed
        stop_words = set(["mightn't", 're', 'wasn', 'wouldn', 'be', 'has', 'that', 'does', 'shouldn', 'do', "you've",'off', 'for', "didn't", 'm', 'ain', 'haven', "weren'



        #removing stopwords and applying lemmatizing nlp process to words
        lr = WordNetLemmatizer()
        filtered_text = []
        for w,p in zip(words,tagged):
            if w not in stop_words:
                if p[1]=='VBG' or p[1]=='VBD' or p[1]=='VBZ' or p[1]=='VBN' or p[1]=='NN':
                    filtered_text.append(lr.lemmatize(w,pos='v'))
                elif p[1]=='JJ' or p[1]=='JJR' or p[1]=='JJS'or p[1]=='RBR' or p[1]=='RBS':
                    filtered_text.append(lr.lemmatize(w,pos='a'))

                else:
                    filtered_text.append(lr.lemmatize(w))
```

Fig 3.8: views.py (Speech recognition module, where we record speech using the API).

Module(2)- Read the text and Provide us with the suitable Signs in ISL.

Traverse the input sequence from front to back and vice versa. And finally provide us with the best and accurate result. Populating a list and displaying the result one by one in a loop is the final goal.

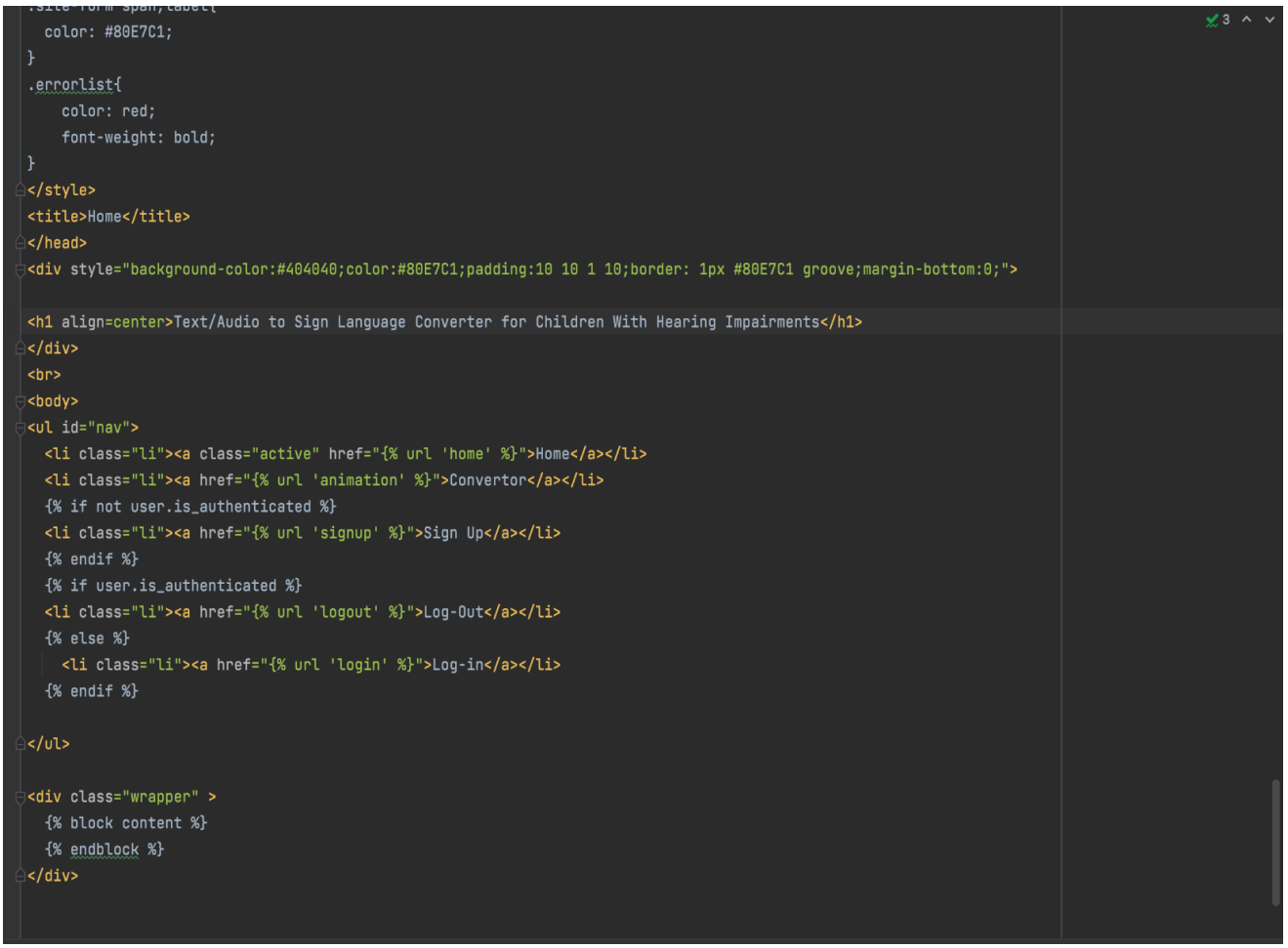


```
def contact_view(request):
    return render(request,'contact.html')

@login_required(login_url="login")
def animation_view(request):
    if request.method == 'POST':
        text = request.POST.get('sen')
        #tokenizing the sentence
        text.lower()
        #tokenizing the sentence
        words = word_tokenize(text)

        tagged = nltk.pos_tag(words)
        tense = {}
        tense["future"] = len([word for word in tagged if word[1] == "MD"])
        tense["present"] = len([word for word in tagged if word[1] in ["VBP", "VBZ","VBG"]])
        tense["past"] = len([word for word in tagged if word[1] in ["VBD", "VBN"]])
        tense["present_continuous"] = len([word for word in tagged if word[1] in ["VBG"]])



        #stopwords that will be removed
        stop_words = set(["mightn't", 're', 'wasn', 'wouldn', 'be', 'has', 'that', 'does', 'shouldn', 'do', "you've",'off', 'for'



        #removing stopwords and applying lemmatizing nlp process to words
        lr = WordNetLemmatizer()
        filtered_text = []
        for w,p in zip(words,tagged):
            if w not in stop_words:
                if p[1]=='VBG' or p[1]=='VBD' or p[1]=='VBZ' or p[1]=='VBN' or p[1]=='NN':
                    filtered_text.append(lr.lemmatize(w,pos='v'))
                elif p[1]=='JJ' or p[1]=='JJR' or p[1]=='JJS'or p[1]=='RBR' or p[1]=='RBS':
                    filtered_text.append(lr.lemmatize(w,pos='a'))
```

Fig 3.9: views.py (Code that is reading the text and providing us with the suitable signs in ISL)

Module(3)- To provide an efficient and reliable User Interface.

Positions of different objects are placed in a manner to achieve the best and eye soothing result. [9]Every button with different functionality is placed in a very specific way to control the flow of outputs.



```
  color: #80E7C1;
}
.errorlist{
    color: red;
    font-weight: bold;
}
</style>
<title>Home</title>
</head>
<div style="background-color:#404040;color:#80E7C1;padding:10 10 1 10;border: 1px #80E7C1 groove;margin-bottom:0;">

<h1 align=center>Text/Audio to Sign Language Converter for Children With Hearing Impairments</h1>
</div>
<br>
<body>
<ul id="nav">
  <li class="li"><a class="active" href="{% url 'home' %}">Home</a></li>
  <li class="li"><a href="{% url 'animation' %}">Convertor</a></li>
  {% if not user.is_authenticated %}
  <li class="li"><a href="{% url 'signup' %}">Sign Up</a></li>
  {% endif %}
  {% if user.is_authenticated %}
  <li class="li"><a href="{% url 'logout' %}">Log-Out</a></li>
  {% else %}
    <li class="li"><a href="{% url 'login' %}">Log-in</a></li>
  {% endif %}

</ul>

<div class="wrapper" >
  {% block content %}
  {% endblock %}
</div>
```

Fig 3.10: base.html Above image depicts how the UI is developed. To provide an efficient and reliable User Interface we have used HTML, CSS and Javascript.

Chapter 4: Results/Outputs

Fig 4.1: Home page

The above screenshot is the home page. It has four sections:

1. Home - Users can use this to return directly to the homepage after using the convertor tool.
2. Convertor- This takes the user to the tool only if the user is logged in, else they will first have to create an account.
3. Sign up- This is used to create an account using username and passwords.
4. Login- This page allows users to login and directly use the convertor tool.

Fig 4.2: Sign up page

The above page is displayed to a new user. The user has to enter their username along with a password of their choice. The password is made to re-enter for confirmation. Passwords have limitations and checks so that the user creates strong passwords.

Fig 4.3: Login page

The above screenshot shows the login page where the user enters his/her credentials that were used by them to create their account. Only then can they login and use the convertor. This step enhances security and privacy.

Fig 4.4: Convertor page.

Users can either enter the text or use the mic button to enter via speech. Then the submit button needs to be clicked. After clicking the submit button each word or alphabet is tokenized and splitted. Each text or word is shown after "words in sentences".

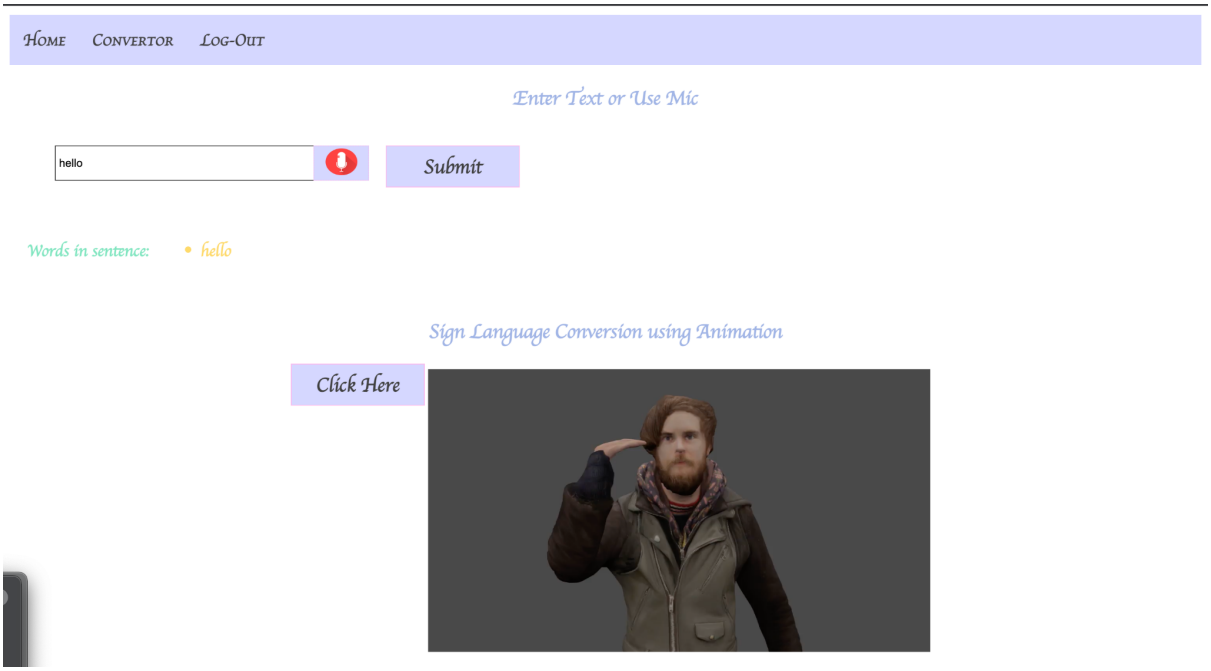


Fig 4.5: Output window showing text converted into ISL.

The above screenshot shows output of text into ISL(Indian Sign Language). This screen is after signing up and logging in. We have also given a button to control the animation(play/pause button).

Chapter 5 : Conclusion and Future Scope

This app for children with hearing impairment is a solution for those who are disabled and can not hear. Speech to sign language is a 3 layer process in which firstly speech gets converted to text and the corresponding text is then converted in sign language.

The sign language used here is Indian Sign Language(ISL) which gives this app an advantage for the people of India. Finally, proposals for further changes to this system were made.

The project's goal is to enter, process, and train speech into sign language in order to convert them into sign language to a better input version.
This project aims to create applications that will aid in the identification of characters in sign language. Only characters and English numerals are allowed in this project. It also aids in the recall of characters in various languages.

The initial component of voice recognition is, of course, speech. Actual sound is converted to an electrical signal by a microphone, which is then converted to digital data by an analog-to-digital converter. Different models can be used to transcribe the audio to text (in this example, sign language) once it has been digitised, which is quite significant.

Hearing-impaired people are more likely to come from low-income homes, have less education, and be unemployed, which is a major problem. Nearly 5% of the world's population, or 432 million individuals, suffer from hearing loss, including 34 million children.

5.1 Future Scope:

- Live conversion can be done, where we can reduce the one layer of text and speech that can be directly converted to sign language in real time.

- Accuracy of the speech can be increased by reducing the background noise.

- Machine learning can be used to know the patterns and types of users and provide them with the best result.

- This can be used as an end-to-end solution and interpreter on the website which anyone can access from anywhere.

- The software can be used for educational purposes in schools/colleges.

References

1. Dhilip Subramanian:“Easy Speech-to-Text with Python. Weblog, https://towardsdatascience.com/easy-speech-to-text-with-python-3df0d973b426Malinda Punchimudiyanse, Gayan Meegama: 3-D animation for sign language, March 2015: Conference*:* International Conference on Engineering and Technology, Sri Lanka”.
2. Aaditya Chintaman Punekar: “A Translator for Indian Sign Language to Text and Speech, June 2020, DOI:10.22214/ijraset.2020.6267”.
3. Hemang Monga, Jatin Bhutani, Muskan Ahuja, Nikita Maid: “Speech to Indian Sign Language Translator, December 2021, DOI:10.3233/APC210172”, In book: “Recent Trends in Intensive Computing”.
4. Goyal, L., & Goyal, V. (n.d.). “Automatic Translation of English Text to Indian Sign Language http://www.aclweb.org/anthology/W16-6319 30”.
5. Zouhour Tmar, Achraf Othman & Mohamed Jemni: “A rule-based approach for building an artificial intelligence English-ASL corpus http://ieeexplore.ieee.org/document/6578458/”. Last accessed: January, 2022.
6. P. Kar, M. Reddy, A. Mukherjee, A. M. Raina. 2017. “INGIT: Limited Domain Formulaic Translation from Hindi Strings to Indian Sign Language. ICON”.
7. “Dictionary | Indian Sign Language. (n.d.). Retrieved July 15, 2016, from http://indiansignlanguage.org/dictionary”. Last accessed: February, 2022.
8. Zeshan, U., Vasishta, M. N., & Sethna, M. (2015). “Implementation of Indian Sign Language in educational settings. Asia Pacific Disability Rehabilitation Journal, 16(1), 16- 40”.
9. Teranai Vichyaloetsiri & Pongpisit Wuttidittachotti: “Web service framework to translate text into sign language”: http://ieeexplore.ieee.org/document/8035336/. Last accessed: March, 2022.
10. S. Cox, M. Lincoln, J. Tryggvason, M. Nakisa, M . Wells, M. Tutt, S. Abbott. 2002. “Tessa, a system to aid communication with deaf people. Fifth international ACM conference on Assistive technologies”.

11. M. Vasishta, J. Woodward and S. DeSantis. 2011. “An Introduction to Indian Sign Language. All India Federation of the Deaf (Third Edition)”.
12. Davis, H. Silverman, R. 1978. Hearing and Deafness. N.Y. Holt Rinehart and Winston. Grieve-Smith, A. B. 1999. “English to American sign language machine translation of weather reports. High Desert Linguistics Conference”.
13. Sulabha M Naik Mahendra S Naik Akriti Sharma "Rehabilitation of hearing impaired children in India International Journal of Advanced Research in Computer and Communication Engineering”.
14. Christopher A.N. Kurz “The pedagogical struggle of mathematics education for the deaf during the late nineteen century: Mental Arithmetic and conceptual understanding” Rochester Institute of Technology, Rochester, NY USA. “Interactive Educational Multimedia, Number 10 (April 2005), pp. 54-65”.
15. Maryam Sulemani “Top 7 models explained”, Jan 04, 202, [https://www.educative.io/blog/software-process-model-types, Last accessed February 2022].
16. What is Incremental model- advantages, disadvantages and when to use it.[http://tryqa.com/what-is-incremental-model-advantages-disadvantages-and-when-to-use-it, Last accessed February, 2022].
17. Altex Soft, “Estimating Software Engineering Effort: Project and Product Development Approach”[https://www.altexsoft.com/whitepapers/estimating-software-engineering-effort-project-and-product-development-approach , Last accessed December 2021].

Github link to the project and project report (file) and code:

https://github.com/AnanyaS21/FinalYearProject-Number-80-